Locally Linear Back-propagation Based Contribution for Nonlinear Process Fault Diagnosis

Jinchuan Qian, Li Jiang, and Zhihuan Song

Abstract — **This paper proposes a novel locally linear back-propagation based contribution (LLBBC) for nonlinear process fault diagnosis. As a method based on the deep learning model of auto-encoder (AE), LLBBC can deal with the fault diagnosis problem through extracting nonlinear features. When the on-line fault diagnosis task is in progress, a locally linear model is firstly built at the current fault sample. According to the basic idea of reconstruction based contribution (RBC), the propagation of fault information is described by using back-propagation (BP) algorithm. Then, a contribution index is established to measure the correlation between the variable and the fault, and the final diagnosis result is obtained by searching variables with large contributions. The smearing effect, which is an important factor affecting the performance of fault diagnosis, can be suppressed as well, and the theoretical analysis reveals that the correct diagnosis can be guaranteed by LLBBC. Finally, the feasibility and effectiveness of the proposed method are verified through a nonlinear numerical example and the Tennessee Eastman benchmark process.**

Index Terms—**Auto-encoder (AE), deep learning, fault diagnosis, locally linear model, nonlinear process, reconstruction based contribution (RBC).**

I. Introduction

In order to keep modern industrial plants to work in normal operation and improve product qualities, process monitoring technique has been widely developed in recent decades. With the advanced computer and networked control system techniques, a large number of the process data have been recorded and stored in industrial databases in recent years. Meanwhile, data-driven multivariable statistical process monitoring (MSPM) has received great attention as extracting useful information from process data can be more convenient and flexible than traditional mechanism-based methods [1]–[3]. Most of the primitive MSPM methods are based on linear models, such as principal component analysis (PCA) and partial least square (PLS), which assume that the correlation between different process variables are linear. However, in actual industry processes, nonlinear correlations are widespread, which will seriously affect the monitoring performance of those methods. Thus, several extensions of MSPM methods have been proposed to handle the nonlinear problem with the kernel PCA [4], the support vector data description (SVDD) [5] and neural network based methods like auto-associative neural networks [6] and principal curve based nonlinear PCA [7].

As a crucial part of the process monitoring, fault diagnosis aims to find the faulty part or component of the process, which can help engineers to locate the root causes of the faults and fix the responsible part in the entire process after the fault detection. And one way to do it is to find the critical variables to the detected fault, also known as fault identification. Contribution plots and reconstruction based contribution (RBC) are two traditional data-based methods for fault diagnosis [8]. Contribution plots can find the faulty variables by calculating the contribution of each process variable to the fault detection index [9], Tan *et al.* [10], [11] improve the performance of the contribution plot by combining this method with different monitoring models. However, because of the existence of the smearing effect, the contribution plot may not be able to give the correct diagnosis results, and RBC is proposed to solve this problem. Compared to the contribution plots method, RBC considers the fault information propagation in the model, which is able to suppress the smearing effect and has been proven to have better diagnosis performance [12]. The traditional RBC method is established based on the linear PCA model, which is not suitable for the nonlinear process. To address the nonlinear issue in the process, several improvements on RBC have been proposed. Alcala *et al*. [13] have extended RBC to the kernel PCA (KPCA) model as a nonlinear version (KPCA-RBC). However, since the dimension of the kernel matrix equals the number of samples, the calculation would be severely time-consuming when dealing with large-scale datasets, which makes KPCA-RBC hard to be implemented in practice. Ge *et al*. [14] approximate the nonlinear feature space with several linear subspaces to build linear RBC in each of them and combine the results by Bayesian inference PCA (BSPCA). It is effective and easy to construct, nevertheless, it may not be able to capture strong nonlinear features of processes. According to the recent works, variable selection methods are used to locate the critical variables to the fault. Yan *et al*. propose a least absolute shrinkage and selection operator (LASSO) based method to identify the faulty variable [15] and further combine LASSO with PLS

Manuscript received January 1, 2020; revised February 21, 2020; accepted March 6, 2020. This work was supported by the Key Project of National Natural Science Foundation of China (61933013) and Ningbo 13th Five-year Marine Economic Innovation and Development Demonstration Project (NBHY-2017-Z1). Recommended by Associate Editor Jun Fu. *(Corresponding author: Zhihuan Song.)*

Citation: J. C. Qian, L. Jiang, and Z. H. Song, "Locally linear back-propagation based contribution for nonlinear process fault diagnosis," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 3, pp. 764–775, May 2020.

J. C. Qian and Z. H. Song are with the State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: qianjinchuan@zju.edu.cn; songzhihuan@zju.edu.cn).

L. Jiang is with Shanghai Research Institute of Huawei Technology Co., Ltd, Shanghai 200127, China (e-mail: jiangli1989@zju.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2020.1003147

and discriminant analysis to improve the performance [16]. Yu *et al.* [17] build a fault relevance based on the kernel canonical correlation analysis (KCCA) to describe the correlation between variables and faults. Compared to the contribution plot and RBC, these methods have better performance when dealing with multi-variate fault, however, the variable selecting process is time-consuming and some of the works can only be performed when a fault dataset has been built, which means that these methods cannot give the result in time when used online.

Deep learning has become a hot research topic in the fields of artificial intelligent and machine learning in the recent years. It includes a series of powerful feature extract models, such as auto-encoder (AE), restricted Boltzman machine (RBM), deep brief network (DBN), etc. These models can learn the representation of the data and extract complex nonlinear features [18], [19]. Several works have been conducted to handle the practical industry problem such as quality predicting and process monitoring with the help of deep learning models. Yuan *et al*. [20]–[22] developed several novel nonlinear feature extraction methods based on variable-wise weighted stacked AE and long short-term memory network (LSTM). Yan *et al*. [23] proposed a variant AE method to solve the nonlinear fault detection problem. Jiang *et al*. [24] further improved the monitoring performance of AE with a denoising criterion. Besides, Zhao [25] proposed a new monitoring model by combining AE and PCA. For fault diagnosis, most work based on deep learning models treated it as a classification task. Shao *et al*. [26] proposed a tracking deep wavelet auto-encoder method for fault diagnosis of electric locomotive bearings, Tamilselvan *et al*. [27] applied DBN for health diagnosis. Wang *et al*. [28] proposed a novel extended DBN model to perform fault diagnosis task in chemical process. However, few methods can be found for analyzing the critical variables of process faults with deep learning models.

The motivation of the paper is to develop a fault diagnosis method based on the deep learning model for industrial processes with strong nonlinear relationships between variables. A novel method called locally linear back-propagation based contribution (LLBBC) is proposed for fault diagnosis in this paper. In LLBBC, an AE model is firstly trained offline. When a fault sample is detected, a local linear model at the fault sample is built to approximate the whole AE model. Then the basic idea of RBC is utilized to calculate the contribution of each variable. Due to the similarity of the propagation of the fault information and the training error, back-propagation (BP) algorithm is used to describe the propagation of the fault in calculating the contribution. Theoretically, the nonlinear features extracted by AE make the method suit for the fault diagnosis of nonlinear process, and the local linear model of LLBBC can prevent diagnosis results from the smearing effect. Two case studies presented in the paper will demonstrate the superiority of the proposed method.

The organization of the remaining paper is given as follows: Section II presents a brief review about AE and stacked auto-encoder (SAE) and the training strategy of denoising criterion. In Section III, two fault diagnosis methods in AE, including back-propagation based contribution (BBC) and the proposed LLBBC are introduced, theoretical proof of the validity of LLBBC and the relationship between LLBBC and RBC are also given in this section. Section IV provides two case studies including a numerical example and the Tennessee Eastman benchmark process. Finally, conclusions are made.

II. Preliminaries

A. Auto-encoder (AE) Model

The auto-encoder model is an unsupervised feed-forward neural network, which is widely used for feature extraction [19]. The architecture of the simplest AE model consists of three layers: an input layer, a hidden layer and an output layer (as shown in Fig. 1).

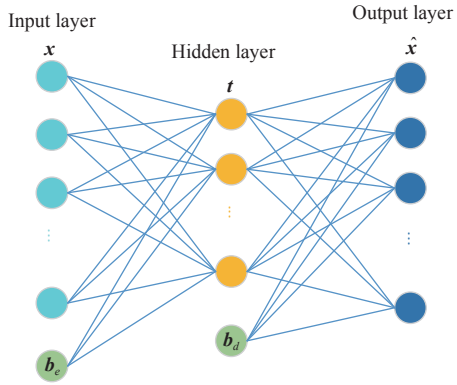

Fig. 1. Architecture of the basic AE.

The total mapping function of AE contains two parts, an encoder function, which maps the input to the feature space, and a decoder function, which is used to reconstruct the input [29]. And the parameters of AE are trained by minimizing the reconstruction error.

Assume that the nth input sample is denoted by $\boldsymbol{x}^{(n)} \in \mathbb{R}^m$, where m is the number of variables. Firstly, the samples are mapped to the feature space (hidden layer) by encoder function as follows:

$$t(\boldsymbol{x}^{(n)}) = \sigma(\boldsymbol{W}_e^T \boldsymbol{x}^{(n)} + \boldsymbol{b}_e) \tag{1}$$

where $\sigma(*)$ is the sigmoid function and calculated as $\sigma(*) = 1/(1+e^{-*})$. $\boldsymbol{W}_e \in \mathbb{R}^{m\times d}$ and $\boldsymbol{b}_e \in \mathbb{R}^d$ denote the weights and the bias of the encoder function, respectively, where d is the number of the node in the hidden layer, i.e., the dimension of the feature space.

Then the feature expression of the hidden layer is reconstructed to input space by the decoder function, given as follows:

$$\hat{\boldsymbol{x}}^{(n)} = \boldsymbol{W}_d^T t(\boldsymbol{x}^{(n)}) + \boldsymbol{b}_d \tag{2}$$

where $\boldsymbol{W}_d \in \mathbb{R}^{d\times m}$ and $\boldsymbol{b}_d \in \mathbb{R}^m$ denote the weights and the bias of the decoder function, respectively. The total mapping function from the input layer to the output layer is shown as follows:

$$r(\boldsymbol{x}^{(n)}) = \boldsymbol{W}_d^T \sigma(\boldsymbol{W}_e^T \boldsymbol{x}^{(n)} + \boldsymbol{b}_e) + \boldsymbol{b}_d. \tag{3}$$

Finally, the BP algorithm can be utilized to optimize all the parameters $\boldsymbol{W}=\{\boldsymbol{W}_e,\boldsymbol{W}_d,\boldsymbol{b}_e,\boldsymbol{b}_d\}$ of the AE model by minimizing the reconstruction error as follows:

$$\boldsymbol{W}=\arg\min_{\boldsymbol{W}}\frac{1}{N}\sum_{n=1}^{N}\left\|\hat{\boldsymbol{x}}^{(n)}-\boldsymbol{x}^{(n)}\right\|^2 \tag{4}$$

where N is the number of the training samples.

B. Denoising Criterion

The denoising criterion is a training strategy that can help AE to extract more robust feature and structure in the input distribution, then AE can obtain better representation [24]. The AE model trained by denoising criterion is also called denoising auto-encoder (DAE) [30].

The key point of the denoising criterion is adding some noise to the input data before training the whole AE model, and then use the corrupted data to reconstruct the original data. The loss-function of DAE can be described by the following equation:

$$\boldsymbol{W}=\arg\min_{\boldsymbol{W}}\frac{1}{N}\sum_{n=1}^{N}\left\|r(\boldsymbol{x}^{(n)}+\boldsymbol{\varepsilon})-\boldsymbol{x}^{(n)}\right\|^2 \tag{5}$$

where $\boldsymbol{\varepsilon}\sim N(0,\sigma^2\boldsymbol{I})$ is the random noise, and the same as the AE model, parameters of DAE can be obtained by minimizing the loss-function with BP algorithm.

When the training of the DAE model is completed, the original uncorrupt data is used as input to map these data to the hidden layer to get the feature representation.

C. Stacked Auto-encoder (SAE) Model

The stacked auto-encoder model can be used to extract features that are more complex, and a common method to train SAE is the greedy layer-wise approach, which means training each layer in turn [31]. Use the original data to train the first AE, and then use the features obtained by the first AE, i.e., the output of the first AE's hidden layer to train the second AE. In a similar fashion, the ith AE can be trained in the same way. After the training of the ith AE is completed, connect them together as the architecture shown in Fig. 2.

As an example, consider the two layers SAE that is stacked by two AEs. The total mapping function of the SAE is expressed as follows:

$$r_{\mathrm{SAE}}(\boldsymbol{x}^{(n)})=\boldsymbol{W}_{1d}^T(\boldsymbol{W}_{2d}^T\sigma(\boldsymbol{W}_{2e}^T\sigma(\boldsymbol{W}_{1e}^T\boldsymbol{x}^{(n)}+\boldsymbol{b}_{1e})+\boldsymbol{b}_{2e})+\boldsymbol{b}_{2d})+\boldsymbol{b}_{1d} \tag{6}$$

where $\boldsymbol{W}_{1e},\boldsymbol{W}_{1d},\boldsymbol{b}_{1e},\boldsymbol{b}_{1d}$ are the parameters of the first AE, and $\boldsymbol{W}_{2e},\boldsymbol{W}_{2d},\boldsymbol{b}_{2e},\boldsymbol{b}_{2d}$ are the parameters of the second AE.

And it is easy to extend it to the multilayer SAE that is stacked by n AEs, as shown in (7), where $\boldsymbol{W}_{ne},\boldsymbol{W}_{nd},\boldsymbol{b}_{ne},\boldsymbol{b}_{nd}$ denote the parameters of the nth AE.

$$\begin{aligned}r_{\mathrm{SAE}}(\boldsymbol{x}^{(n)})=\boldsymbol{W}_{1d}^T(\cdots(\boldsymbol{W}_{(n-1)d}^T(\boldsymbol{W}_{nd}^T\sigma(\boldsymbol{W}_{ne}^T\cdots\\ \sigma(\boldsymbol{W}_{2e}^T\sigma(\boldsymbol{W}_{1e}^T\boldsymbol{x}^{(n)}+\boldsymbol{b}_{1e})\\ +\boldsymbol{b}_{2e})\cdots+\boldsymbol{b}_{ne})+\boldsymbol{b}_{nd})+\boldsymbol{b}_{(n-1)d})\cdots)+\boldsymbol{b}_{1d}.\end{aligned} \tag{7}$$

Denoising criterion can also be used in the training of every single AE, which helps us to build a stacked denoising auto-encoder (SDAE) to extract the more robust and complex nonlinear features.

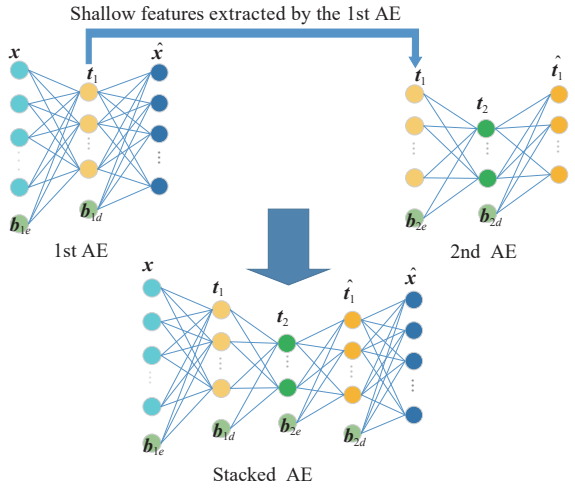

Fig. 2. Training procedure of stacked auto-encoder model.

III. Fault Diagnosis Method Based on Auto-encoder

A. Back-propagation Based Contribution (BBC)

Back-propagation based contribution is a fault diagnosis method based on the AE model. In BBC, nonlinear features extracted by the AE model can be used to obtain better performance in fault diagnosis task. Moreover, when a fault happens, the fault information will propagate around the whole AE model, so all the output variables will contain the fault information, which will lead to the smearing effect and seriously affect the diagnosis result. Therefore, if we just calculate the contribution plot using the input and the output of the AE model, we may not get the correct result. With BBC, the smearing effect can be suppressed by considering the propagation of the fault using the BP algorithm.

Similar to the RBC, the basic idea of the BBC contains two parts. First of all, find an f_i to adjust the ith variable of the online sample $\boldsymbol{x}$ (denoted by x_i) such that the corresponding fault detection index is minimized [12]. Secondly, build an index to measure the magnitude of f_i, and the final diagnosis result is obtained based on the magnitude of the index.

When the fault detection task is performed in the AE model that is trained by denoising criterion, the squared prediction error (SPE) is chosen as the fault detection index, which is calculated by the following equation [12]:

$$SPE_{\mathrm{AE}}=(\hat{\boldsymbol{x}}-\boldsymbol{x})^T(\hat{\boldsymbol{x}}-\boldsymbol{x}) \tag{8}$$

where $\hat{\boldsymbol{x}}$ is the output of the AE model.

In order to calculate the f_i, Back-propagation (BP) algorithm is used to describe the propagation of the fault information. BP is a traditional algorithm for neural network training, and the most important part of the BP algorithm is to calculate the partial derivative of the predictive error with respect to the weight, then the value of the derivative can be used to update the weights until the loss function has approached its minimum. When we are performing the RBC, the first step is to calculate the magnitude f_i, so as to minimize fault detection index of $x+\xi_i f_i$, where ξ_i is the ith column of

the identity matrix, and this is similar to the purpose of the BP algorithm.

In Fig. 3, a structure of AE is illustrated and the formulas below the structure show the procedure of the error back propagation, where a_i^l denotes the output of the ith nodes in the lth layer and z_i^l denotes the input of the ith nodes in the lth layer, different formulas have different colors, and each of them corresponds to a path of the same color in the AE structure.

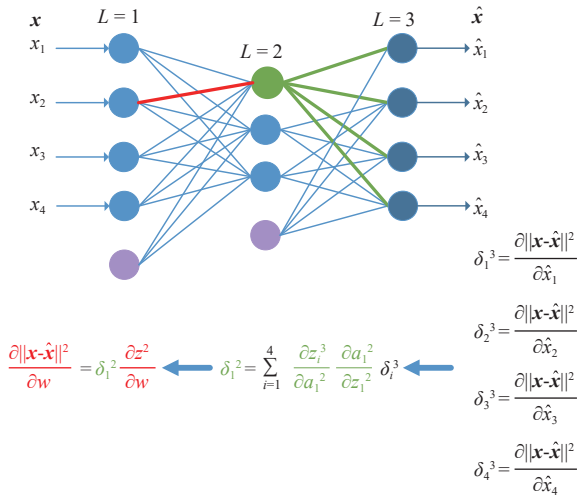

Fig. 3. Error propagation path in BP algorithm.

As shown in Fig. 3, when the BP algorithm calculates the gradient of the weight, the error propagates in an established path at the same time. Due to the similarity of the propagation of the fault information and the error, the BP algorithm can be regarded as a method to describe the propagation of the fault and help to calculate the magnitude of the fault reconstruction. Thus, when a fault sample is obtained, calculate the error $\boldsymbol{E}$ between the input and output of the AE model first, and then calculate the differential of $\boldsymbol{E}$ with respect to x_i by BP algorithm as shown in (9) and (10).

$$\boldsymbol{E} = \boldsymbol{x} - \hat{\boldsymbol{x}} \tag{9}$$

$$\frac{\partial \boldsymbol{E}}{\partial x_i} = \frac{\partial(\boldsymbol{x} - \hat{\boldsymbol{x}})}{\partial x_i} \tag{10}$$

Assume the fault magnitude of the variable x_i is f_i and do the integral in the both sides of (10), the f_i can be calculated by (11).

$$\boldsymbol{E} = \int_{x_i}^{x_i+f_i} \frac{\partial \boldsymbol{E}}{\partial x_i} dx_i$$

$$f_i = \boldsymbol{E}^T \frac{\partial x_i}{\partial \boldsymbol{E}}. \tag{11}$$

The fault detection index of variable x_i is built as the following equation:

$$BBC_i^{\text{index}} = \left\| \boldsymbol{\xi}_i f_i - \hat{\boldsymbol{\xi}}_i f_i \right\|^2 \tag{12}$$

where ξ_i is the ith column of the identity matrix and $\hat{\boldsymbol{\xi}}_i$ denotes the reconstruction result of ξ_i by the trained AE model.

Finally, the fault diagnosis task can be completed by finding the variables with significantly large contributions.

B. Locally Linear Back-propagation Based Contribution (LLBBC)

Although BBC has considered the propagation of the fault, it cannot prevent the smearing effect from affecting the right diagnosis result thoroughly. In this section, a locally linear back-propagation based contribution (LLBBC) method is proposed to solve this problem and get the correct diagnosis with the smearing effect.

It can be seen that the nonlinear part exists in the contribution index of BBC, which represents the mapping function of neural network and is hard to interpret. Thus, the contribution index of BBC is difficult to be described theoretically, and the contribution of the most relevant variables may not have the largest magnitude. However, we need the nonlinear mapping to make use of the nonlinear features extracted by AE. In this situation, a number of local linear models can be used to approximate the whole nonlinear AE model (as shown in the Fig. 4), which can help us to build a contribution index in a linear model. The nonlinear AE model can be specifically described and the nonlinear features extracted by AE can be effectively utilized at the same time.

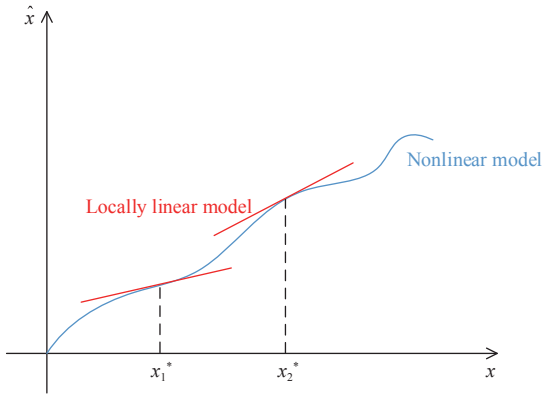

Fig. 4. Description of a locally linear model.

The basic steps of LLBBC are the same as the BBC's, however, the construction of the contribution index needs to be changed. First of all, AE is trained by the dataset from the normal operating state offline using the denoising criterion. When a fault occurs, the model is linearized at the fault sample $\boldsymbol{x}^*$. Since the use of the linear decoder, the nonlinear part of the original mapping function only exists in the encoder function. Then we only need to linearize the encoder function, which can be expressed as

$$\begin{aligned} t(\boldsymbol{x}) &= \left(\frac{\partial t(\boldsymbol{x})}{\partial \boldsymbol{x}}\bigg|_{\boldsymbol{x}^*} \right)^T (\boldsymbol{x} - \boldsymbol{x}^*) + \sigma(\boldsymbol{W}_e^T \boldsymbol{x}^* + \boldsymbol{b}_e) \\ &= (\boldsymbol{W}_e - \boldsymbol{K}_e)^T (\boldsymbol{x} - \boldsymbol{x}^*) + \sigma(\boldsymbol{W}_e^T \boldsymbol{x}^* + \boldsymbol{b}_e). \end{aligned} \tag{13}$$

The total mapping function of the local linear model becomes

$$\begin{aligned} \hat{x}(\boldsymbol{x}) &= \boldsymbol{W}_d^T \boldsymbol{K}_e \boldsymbol{W}_e^T \boldsymbol{x} - \boldsymbol{W}_d^T \boldsymbol{K}_e \boldsymbol{W}_e^T \boldsymbol{x}^* + \boldsymbol{W}_d^T \sigma(\boldsymbol{W}_e^T \boldsymbol{x}^* + \boldsymbol{b}_e) + \boldsymbol{b}_d \\ &= \boldsymbol{K}_{de} \boldsymbol{x} + \boldsymbol{B}_{de}. \end{aligned} \tag{14}$$

The parameters $\boldsymbol{K}_{de}$ and $\boldsymbol{B}_{de}$ in (14) can be calculated by the following equations:

$$\boldsymbol{K}_e = \begin{pmatrix} t^1(\boldsymbol{x})\left(1-t^1(\boldsymbol{x})\right) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & t^d(\boldsymbol{x})\left(1-t^d(\boldsymbol{x})\right) \end{pmatrix} \in \mathbb{R}^{d\times d}$$

$$\boldsymbol{K}_{de} = \boldsymbol{W}_d^T \boldsymbol{K}_e \boldsymbol{W}_e^T$$

$$\boldsymbol{B}_{de} = -\boldsymbol{W}_d^T \boldsymbol{K}_e \boldsymbol{W}_e^T \boldsymbol{x}^* + \boldsymbol{W}_d^T \sigma(\boldsymbol{W}_e^T \boldsymbol{x}^* + \boldsymbol{b}_e) + \boldsymbol{b}_d \tag{15}$$

where $t^i(\boldsymbol{x})$ is the output of the ith node of hidden layer. The calculation method of $\boldsymbol{K}_e$ shown here is based on sigmoid function. If other activation functions are used, we only need to change diagonal elements to corresponding derivatives value.

Then the value of f_i can be obtained by minimizing the square predictive error.

$$f_i = \arg\min_{f_i} \|\boldsymbol{K}_{de}(\boldsymbol{x}-\boldsymbol{\xi}_i f_i) + \boldsymbol{B}_{de} - (\boldsymbol{x}-\boldsymbol{\xi}_i f_i)\|^2. \tag{16}$$

Assuming that $J = \|\boldsymbol{K}_{de}(\boldsymbol{x}-\boldsymbol{\xi}_i f_i) + \boldsymbol{B}_{de} - (\boldsymbol{x}-\boldsymbol{\xi}_i f_i)\|^2$, the minimization can be completed by taking the first derivative of J and equating it to zero.

$$\frac{\partial J}{\partial f_i} = -[(\boldsymbol{K}_{de}-\boldsymbol{I})\boldsymbol{\xi}_i]^T [\boldsymbol{K}_{de}(\boldsymbol{x}-\boldsymbol{\xi}_i f_i) + \boldsymbol{B}_{de} - (\boldsymbol{x}-\boldsymbol{\xi}_i f_i)] = 0. \tag{17}$$

It can be further changed to the following form:

$$\begin{aligned} &[(\boldsymbol{I}-\boldsymbol{K}_{de})\boldsymbol{\xi}_i]^T \boldsymbol{B}_{de} = [(\boldsymbol{I}-\boldsymbol{K}_{de})\boldsymbol{\xi}_i]^T [(\boldsymbol{K}_{de}-\boldsymbol{I})(\boldsymbol{x}^*-\boldsymbol{\xi}_i f_i)] \\ &[K\boldsymbol{\xi}_i]^T \boldsymbol{B}_{de} = [\boldsymbol{K}\boldsymbol{\xi}_i]^T [\boldsymbol{K}(\boldsymbol{x}^*-\boldsymbol{\xi}_i f_i)] \\ &\boldsymbol{\xi}_i^T \boldsymbol{K}^T \boldsymbol{K}\boldsymbol{\xi}_i f_i = [\boldsymbol{K}\boldsymbol{\xi}_i]^T [\boldsymbol{K}\boldsymbol{x}^* - \boldsymbol{B}_{de}]. \end{aligned} \tag{18}$$

Then the solution of f_i can be expressed as

$$f_i = \left[\boldsymbol{\xi}_i^T \boldsymbol{K}^T \boldsymbol{K}\boldsymbol{\xi}_i\right]^{-1} [\boldsymbol{K}\boldsymbol{\xi}_i]^T [\boldsymbol{K}\boldsymbol{x}^* - \boldsymbol{B}_{de}] \tag{19}$$

where $\boldsymbol{K} = \boldsymbol{I} - \boldsymbol{K}_{de}$. Since parameters of the AE are always full-rank, f_i can be calculated as

$$f_i = \left[\boldsymbol{\xi}_i^T \boldsymbol{K}^T \boldsymbol{K}\boldsymbol{\xi}_i\right]^{-1} \left[\boldsymbol{\xi}_i^T \boldsymbol{K}^T \boldsymbol{K}\right]\left[\boldsymbol{x}^* - \boldsymbol{K}^{-1}\boldsymbol{B}_{de}\right]. \tag{20}$$

The next step is to build a contribution index. Follow the idea of building the contribution of the RBC, where a positive semi-definite matrix $\boldsymbol{M}$ is formed to give the correct diagnosis with the smearing effect. We can see from (20) that the matrix $\boldsymbol{M}$ can be calculated by $\boldsymbol{M} = \boldsymbol{K}^T\boldsymbol{K}$, because the matrix $\boldsymbol{K}$ can be used to map the original data to the feature space, which is similar to the loading matrix in the PCA model. However, differently from the RBC, the matrix $\boldsymbol{M}$ in LLBBC changes with the sample. Thus, the contribution of the LLBBC of the variable x_i can be built as follows:

$$\begin{aligned} LLBBC_i &= (\boldsymbol{x}^*)^T \left[\boldsymbol{K}^T \boldsymbol{K}\boldsymbol{\xi}_i\right]\left[\boldsymbol{\xi}_i^T \boldsymbol{K}^T \boldsymbol{K}\boldsymbol{\xi}_i\right]^{-1}\left[\boldsymbol{\xi}_i^T \boldsymbol{K}^T \boldsymbol{K}\right](\boldsymbol{x}^*) \\ &= (\boldsymbol{x}^*)^T [\boldsymbol{M}\boldsymbol{\xi}_i]\left[\boldsymbol{\xi}_i^T \boldsymbol{M}\boldsymbol{\xi}_i\right]^{-1}\left[\boldsymbol{\xi}_i^T \boldsymbol{M}\right](\boldsymbol{x}^*) \\ &= \frac{\left[\left(\boldsymbol{\xi}_i^T \boldsymbol{M}\right)\boldsymbol{x}^*\right]^2}{m_{ii}} \end{aligned} \tag{21}$$

where $\boldsymbol{M} = \boldsymbol{K}^T\boldsymbol{K}$ is symmetrical and positive semi-definite. Equation (21) shows that LLBBC is calculated by reconstruction along each variable, however, like RBC, the diagnostic problem that LLBBC can deal with is not limited to single variable fault. With the help of the nonlinear features, the faulty variables will have significantly larger LLBBC values than those irrelevant variables.

The steps of the LLBBC can be summarized as follows and showed in Fig. 5:

1) Use the normal state data to train AE offline with denoising criterion.

2) When the fault sample $\boldsymbol{x}^*$ has been obtained, calculate the parameters of the locally linearized AE model at the fault sample $\boldsymbol{x}^*$ by (15).

3) Calculate the contribution index of variable x_i by (21).

4) Complete the diagnosis task by finding the variables with large values of contributions.

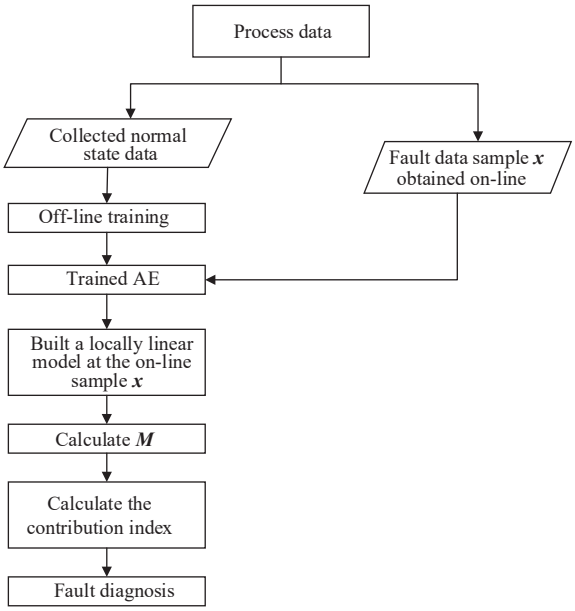

Fig. 5. Flowchart of LLBBC.

C. LLBBC in SAE

In order to extract deeper features, SAE can be used to substitute AE. Then, LLBBC can be easily constructed in SAE with a similar structure as AE, where only a few changes are needed in the calculation of parameters.

Because SAE contains several encoders and decoders, the parameters of each AE should be calculated first, and then, the parameter $\boldsymbol{K}_{de}$ can be obtained as shown in (22).

$$\boldsymbol{K}_{e_i}^i = \begin{pmatrix} t_1^i(\boldsymbol{x})\left(1-t_1^i(\boldsymbol{x})\right) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & t_{d_i}^i(\boldsymbol{x})\left(1-t_{d_i}^i(\boldsymbol{x})\right) \end{pmatrix} \in \mathbb{R}^{d_i\times d_i}$$

$$\boldsymbol{K}_e = \prod_{i=1}^{n_l} \boldsymbol{K}_{e_i}^i \boldsymbol{W}_{e_i}^T$$

$$\boldsymbol{K}_d = \prod_{i=1}^{n_l} \boldsymbol{W}_{d_i}^T$$

$$\boldsymbol{K}_{de} = \boldsymbol{K}_d \boldsymbol{K}_e \tag{22}$$

where d_i is the number of hidden layer nodes of the ith AE, $t_j^i(\boldsymbol{x})$ denotes the output of jth hidden layer node in the ith AE, n_l represents the number of AE, and $\boldsymbol{W}_{e_i}, \boldsymbol{W}_{d_i}$ denote the weights of the ith AE.

$$\boldsymbol{M} = (\boldsymbol{I} - \boldsymbol{K}_{de})^T (\boldsymbol{I} - \boldsymbol{K}_{de}) \tag{23}$$

After $\boldsymbol{K}_{de}$ has been calculated, $\mathbf{M}$ can be calculated as shown in (23) and the index of the contribution can be given by the same way as LLBBC.

D. Fault Smearing in LLBBC

In this section, it is proved that the smearing effect will not affect the diagnosis result in LLBBC. As shown in (21), after using the local linear model, the calculation of the contribution index is the same as RBC. Thus, the same procedure can be used in the proof here [12].

Assume that the fault sample $\boldsymbol{x}^*$ is exactly in the jth direction, that is, $\boldsymbol{x}^* = \boldsymbol{\xi}_j f$. Then the index can be calculated by the following equations:

$$\begin{aligned} LLBBC_i &= \frac{\left[\boldsymbol{\xi}_i^T \boldsymbol{M} \boldsymbol{\xi}_j f\right]^2}{m_{ii}} \\ &= \begin{cases} m_{ii}^{-1} m_{ij}^2 f^2, & i \neq j \\ m_{jj} f^2, & i = j. \end{cases} \end{aligned} \tag{24}$$

According to the property of the semi-definite matrix, we have

$$\begin{bmatrix} \boldsymbol{\xi}_i & \boldsymbol{\xi}_j \end{bmatrix}^T \boldsymbol{M} \begin{bmatrix} \boldsymbol{\xi}_i & \boldsymbol{\xi}_j \end{bmatrix} = \begin{pmatrix} m_{ii} & m_{ij} \\ m_{ij} & m_{jj} \end{pmatrix} \geq 0 \tag{25}$$

which implies

$$\begin{aligned} &\det \begin{bmatrix} m_{ii} & m_{ij} \\ m_{ij} & m_{jj} \end{bmatrix} = m_{ii} m_{jj} - m_{ij} m_{ij} \geq 0 \\ &m_{jj} \geq m_{ii}^{-1} m_{ij}^2. \end{aligned} \tag{26}$$

It shows that $LLBBC_j \geq LLBBC_i$, which guarantees that LLBBC can give the correct diagnosis.

E. The Relationship Between LLBBC and RBC

If LLBBC is used in the linear PCA model, then the parameters of LLBBC can be given as follows:

$$\begin{aligned} &\boldsymbol{M} = (\boldsymbol{I} - \boldsymbol{P})^T (\boldsymbol{I} - \boldsymbol{P}) \\ &\boldsymbol{B}_{de} = \boldsymbol{0} \end{aligned} \tag{27}$$

where $\boldsymbol{P}$ denotes the loading matrix of the PCA model.

Then, put $\boldsymbol{P}$ into (20), and the value of f_i^{LLBBC} can be obtained by

$$f_i^{\text{LLBBC}} = \frac{\boldsymbol{\xi}_i^T \left(\boldsymbol{I} - \boldsymbol{P}^T \boldsymbol{P}\right) \boldsymbol{x}^*}{\boldsymbol{\xi}_i^T \left(\boldsymbol{I} - \boldsymbol{P}^T \boldsymbol{P}\right) \boldsymbol{\xi}_i}. \tag{28}$$

Finally, the contribution index of LLBBC can be described by the following equations:

$$LLBBC_i = \frac{\left[\boldsymbol{\xi}_i^T \left(\boldsymbol{I} - \boldsymbol{P}^T \boldsymbol{P}\right) \boldsymbol{x}\right]^2}{m_{ii}}. \tag{29}$$

From the above two equations, it can be seen that both the value of f_i and the index are the same as those of RBC. Hence, it can be concluded that LLBBC equals RBC in the PCA model.

Fig. 6 shows the relationship between RBC, BBC and LLBBC. The three methods are based on different models, but have similar ideas. PCA and AE have similar model structures, while locally linearized AE is built on the basis of AE by performing local linearization at the online sample. And LLBBC can adapt to nonlinearity better.

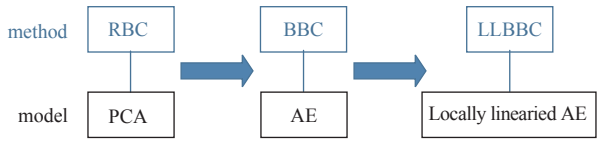

Fig. 6. Relationship diagram between the three method.

IV. Case Study

In this section, the proposed fault diagnosis method LLBBC is applied to a nonlinear numerical example and the simulated Tennessee Eastman process, and the performances are compared with RBC (SPE index), BSPCA, KPCA-RBC and BBC. The number of the principle components in the PCA model is determined by the cumulative variance of 90%. The Gaussian kernel $k(x,y) = \exp(-\|x-y\|^2/c)$ is used in the KPCA model here and the parameter c is selected by the empirical formula $c = 10m$, where m is the dimension of the input space [4]. The maximum number of iteration is 20 in KPCA-RBC, and the convergence condition is set to $|f(k-2) - f(k)| < 0.0001$, where $f(k)$ denotes the value of the contribution in KPCA-RBC after the kth iteration. All the simulations are performed in environment of Core i7-6700 CPU.

A. A Nonlinear Numerical Example

The data of the nonlinear numerical example are generated by (30).

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} -0.2 & -0.08 & -0.3 \\ -0.3 & 0.7 & -0.2 \\ -0.2 & -0.3 & -0.5 \\ -0.4 & -0.3 & -0.4 \\ -0.6 & 0.3 & 0.2 \\ -0.5 & -0.4 & 0.6 \end{bmatrix} \begin{bmatrix} \sin(t_1^2) \\ t_2^3 \\ \exp(t_3) \end{bmatrix} + \varepsilon \tag{30}$$

where t_1, t_2, t_3 are the latent variables subject to the zero-mean Gaussian distribution with variance 0.6, and ε denotes the noises, which follow the zero-mean Gaussian distribution with variance 0.02.

The faults are set by the form $\boldsymbol{x}_{\text{fault}} = \boldsymbol{x}^* + \boldsymbol{\xi}_i f$, where $\boldsymbol{x}^*$ is the normal state data generated by (30), $\boldsymbol{\xi}_i$ is the fault direction, which is out of the six possible variable directions.

The training dataset includes 1000 normal state samples. The number of the hidden nodes in AE is 3, and the whole model is trained with the denoising criterion. The noise chosen to corrupt the original normal state data follows $N(0, 0.09)$. The testing dataset includes 1000 samples and 6 types of faults are set in the latter 600 samples. The details of the 6 fault modes are listed in Table I.

Fig. 7 illustrates the detailed fault diagnosis results given by

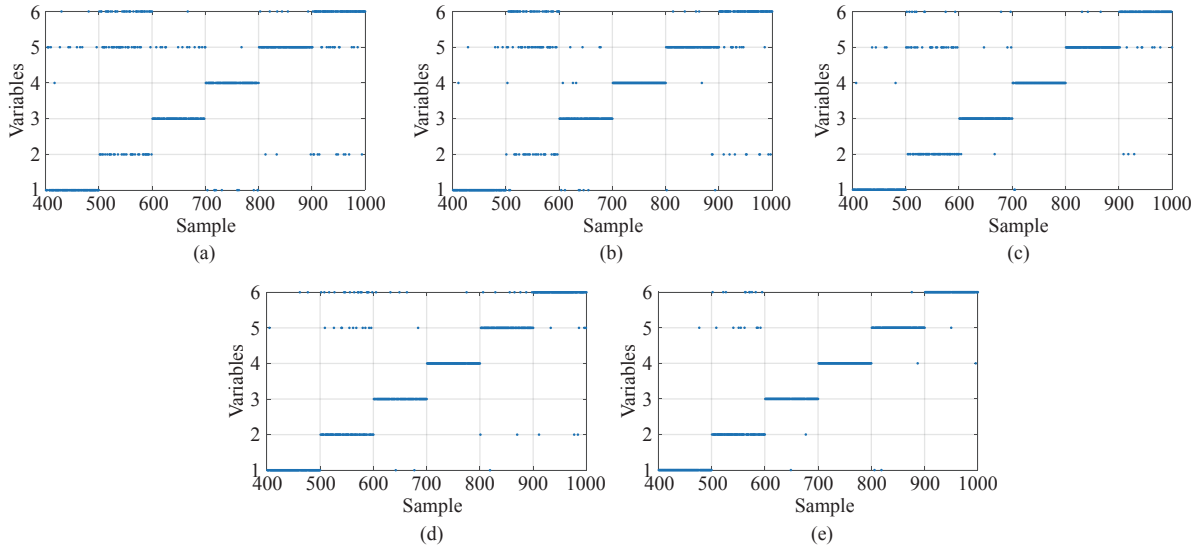

Fig. 7. Fault diagnosis results for the numerical example (a) RBC; (b) BSPCA; (c) KPCA_RBC; (d) BBC; (e) LLBBC.

TABLE I
FAULT DESCRIPTIONS IN THE NUMERICAL EXAMPLE

Fault	Variable	Sample	Type	Magnitude
1	x_1	401–500	Random noise	$N(0,1)$
2	x_2	501–600	Step	0.5
3	x_3	601–700	Random noise	$N(0,1.2)$
4	x_4	701–800	Step	0.8
5	x_5	801–900	Random noise	$N(0,1.5)$
6	x_6	901–1000	Step	1

RBC, BSPCA, KPCA-RBC, BBC and LLBBC. The blue points in the Fig. 7 indicate the variable that owns the biggest contribution from sample 401 to sample 1000, i.e., the diagnosis result. And Table II shows the average fault diagnosis accuracy of different methods after testing 10 times.

According to the Fig. 7 and Table II, we can see that both BBC and LLBBC have better performance than other methods in the most faults, especially in the faults 1, 2, 3 and 6. Since BBC and LLBBC can utilize the nonlinear features extracted by AE, the diagnostic performance was improved enormously. Moreover, on the basis of BBC, LLBBC improves the construction of the contribution index to have better suppression of the smearing effect. Thus, from those results, it can be seen that LLBBC has higher accuracy than BBC in all faults, especially in fault 5, where BBC has the worst performance among all the methods, but LLBBC still has the highest accuracy.

B. Tennessee Eastman Process

The Tennessee Eastman (TE) process is a chemical testing experimental platform that developed from a realistic chemical joint reaction process. It has been widely used for the evaluation and comparison of the performance of various process monitoring methods in recent years [32]. The TE benchmark process contains five major operating units: reactor, condenser, compressor, separator, and stripper. The schematic diagram of the TE process is illustrated in Fig. 8.

TABLE II
AVERAGE FAULT DIAGNOSIS ACCURACY OF DIFFERENT METHODS (%)

Fault	RBC	BSPCA	KPCA-RBC	BBC	LLBBC
1	88.3	94.1	94.2	94.6	97
2	25.7	25.3	50	76.7	85.3
3	84.9	84.4	92.5	94.7	96.2
4	94.5	99.6	98.1	99.9	99.9
5	92.7	91.1	93.4	90.1	94.5
6	80.5	80.9	87.3	91.3	98.2

Among all the 52 process variables, 33 measurement variables are selected for fault diagnosis in the TE process and the descriptions are illustrated in Table III. Besides, 5 faults are chosen and listed in Table IV for the comparison of fault diagnosis performance. The training dataset includes 500 samples that are acquired from the normal operating condition, and five testing datasets are set with 100 fault samples in each fault mode.

The performances of RBC, BSPCA, KPCA-RBC, BBC, LLBBC and LLBBC in SAE are compared in this subsection. AE is also trained with denoising criterion, the noise for corruption follows $N(0,0.09)$, and the number of the hidden layer nodes in AE is 28. The structure of SAE in this simulation is 33-60-28-60-33, which means that the input layer has 33 nodes, the first hidden layer has 60 nodes, the number of the second hidden layer nodes is 28.

Figs. 9–13 illustrate the detailed fault diagnosis results for the five faults and the length of each bar represents the contribution of each variable in different methods. And the contributions shown in these figures are the averaged result in the corresponding testing dataset. Table V collects the average time required for KPCA-RBC and LLBBC to diagnose one sample after testing 10 times. Note that the diagnosis time of LLBBC listed in the Table V is the sum of the time for establishing the local linear model and the time for computing the contribution.

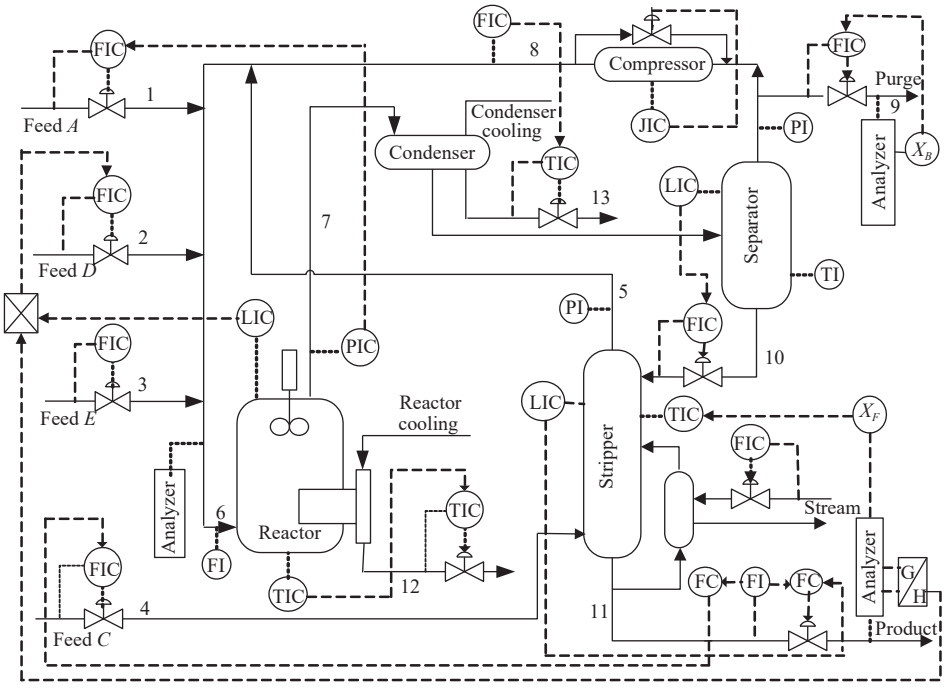

Fig. 8. Schematic diagram of the TE process.

TABLE III
DESCRIPTION OF THE VARIABLES IN TE PROCESS

No.	Measured variable	No.	Measured variable	No.	Measured variable
1	feed *A*	12	product separator level	23	feed *D* flow valve
2	feed *D*	13	product separator pressure	24	feed *E* flow valve
3	feed *E*	14	product separator underflow	25	feed *A* flow valve
4	total feed	15	stripper level	26	total feed flow valve
5	recycle flow	16	stripper pressure	27	compressor recycle valve
6	reactor feed rate	17	stripper underflow	28	purge valve
7	reactor pressure	18	stripper temperature	29	separator pot liquid flow valve
8	reactor level	19	stripper steam flow	30	stripper liquidproduct flow valve
9	reactor temperature	20	compressor work	31	stripper steam valve
10	purge rate	21	reactor cooling water outlet temperature	32	reactor cooling water flow
11	product separator temperature	22	separator cooling water outlet temperature	33	Condenser cooling water flow

TABLE IV
FAULT DESCRIPTIONS IN THE TE PROCESS

Fault	Description	Type
1	feed *A/C* ratio, *B* composition constant (stream 4)	step
4	reactor cooling water inlet temperature	step
5	condenser cooling water inlet temperature (stream 2)	step
10	feed *C* temperature (stream 4)	random variation
14	reactor cooling water valve	sticking

TABLE V
DIAGNOSTIC TIME FOR KPCA-RBC AND LLBBC (S)

Fault	Methods	
	KPCA_RBC	LLBBC
Fault 1	1.18814	0.00034
Fault 4	1.97411	0.00051
Fault 5	2.35078	0.00090
Fault 10	2.40957	0.00150
Fault 14	1.18813	0.00043

Figs. 9 and 10 illustrate the diagnosis results of fault 1 and fault 4. According to the fault description listed in Table IV, the most relevant variables to fault 1 are variable 1 and variable 25, and the most relevant variable to fault 4 is variable 32. It can be seen from the Fig. 9 that the nonlinear methods, KPCA-RBC, BBC, LLBBC, and LLBBC in SAE can all give the correct diagnosis, while the results given by RBC and BSPCA are not correct. This is because RBC, as a linear diagnosis method, cannot handle the nonlinear problem, and BSPCA has limited ability to extract nonlinear features.

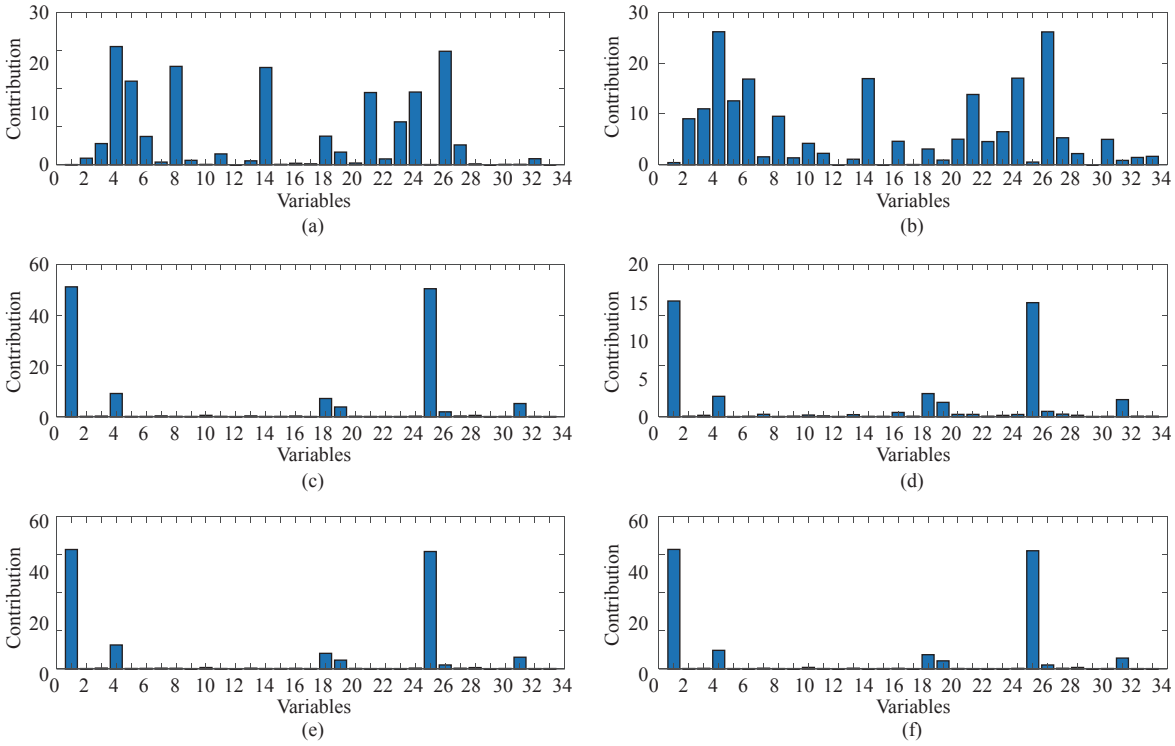

Fig. 9. Fault diagnosis of fault 1 (a) RBC; (b) BSPCA; (c) KPCA_RBC; (d) BBC; (e) LLBBC; (f) LLBBC in SAE.

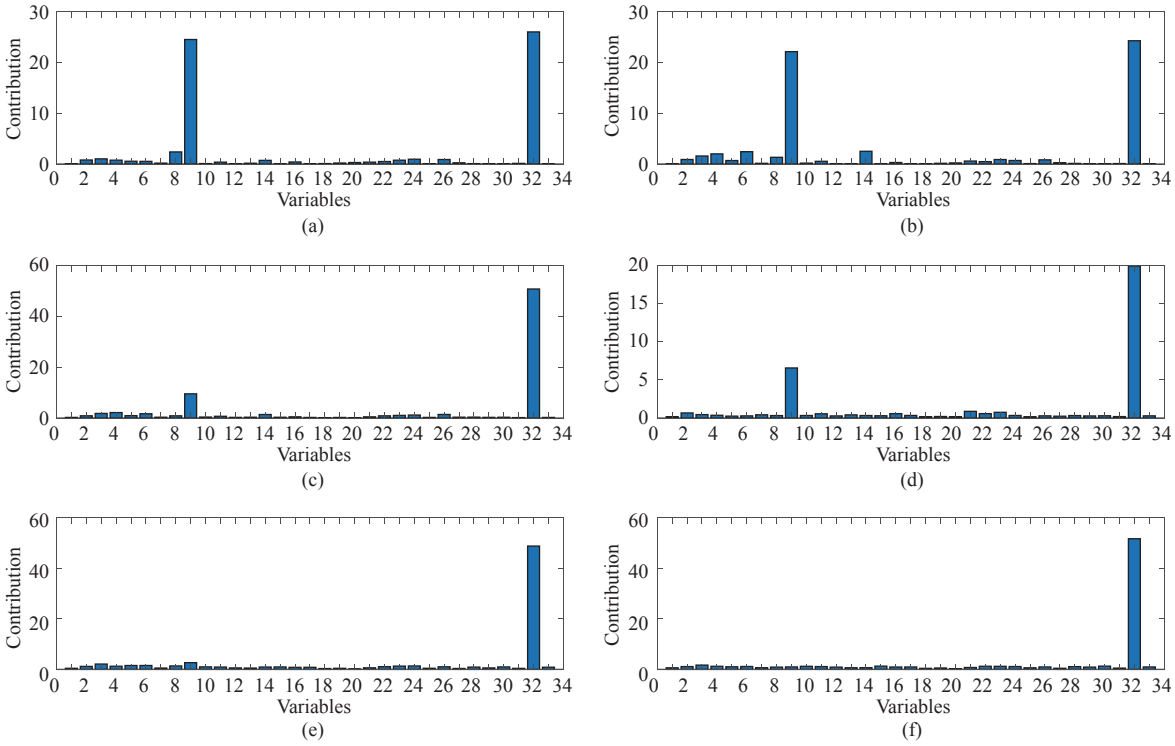

Fig. 10. Fault diagnosis of fault 4 (a) RBC; (b) BSPCA; (c) KPCA_RBC; (d) BBC; (e) LLBBC; (f) LLBBC in SAE.

Fig. 10 shows that the results given by RBC and BSPCA are confusing for the contributions of the variable 9 have large magnitudes. LLBBC can utilize the nonlinear features extracted by AE and have better suppression of the smearing effect. Therefore, LLBBC can make the contributions of the irrelevant variables lower than other methods.

Fig. 11 shows the diagnosis results of fault 5. And the most relevant variable of the fault 5 is variable 33. And according to Fig. 11, we can find that RBC and BSPCA give a high contribution to the variable 17, which is irrelevant to the fault 5. Comparison among Figs. 11 (c)–(f) shows that LLBBC outperforms BBC for the variable 17 in LLBBC has a lower contribution, and we can see from Fig. 11 (f) that when LLBBC is used in SAE model, the contribution of the key

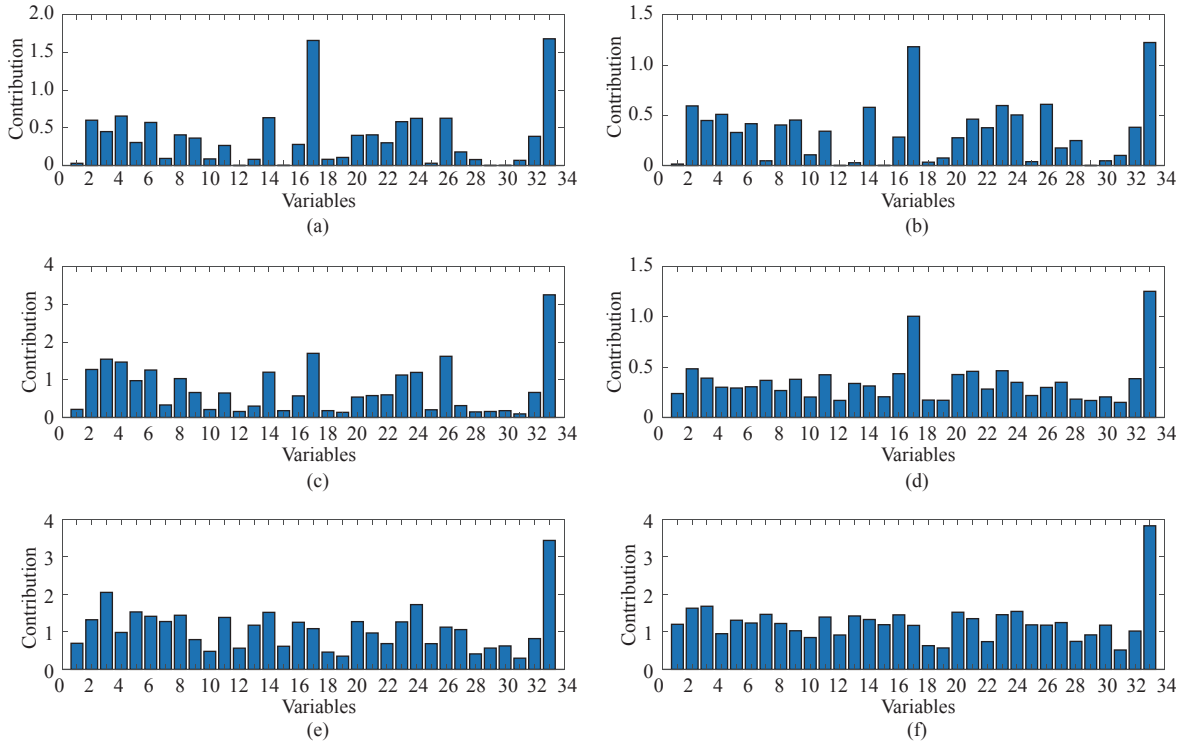

Fig. 11. Fault diagnosis of fault 5 (a) RBC; (b) BSPCA; (c) KPCA_RBC; (d) BBC; (e) LLBBC; (f) LLBBC in SAE.

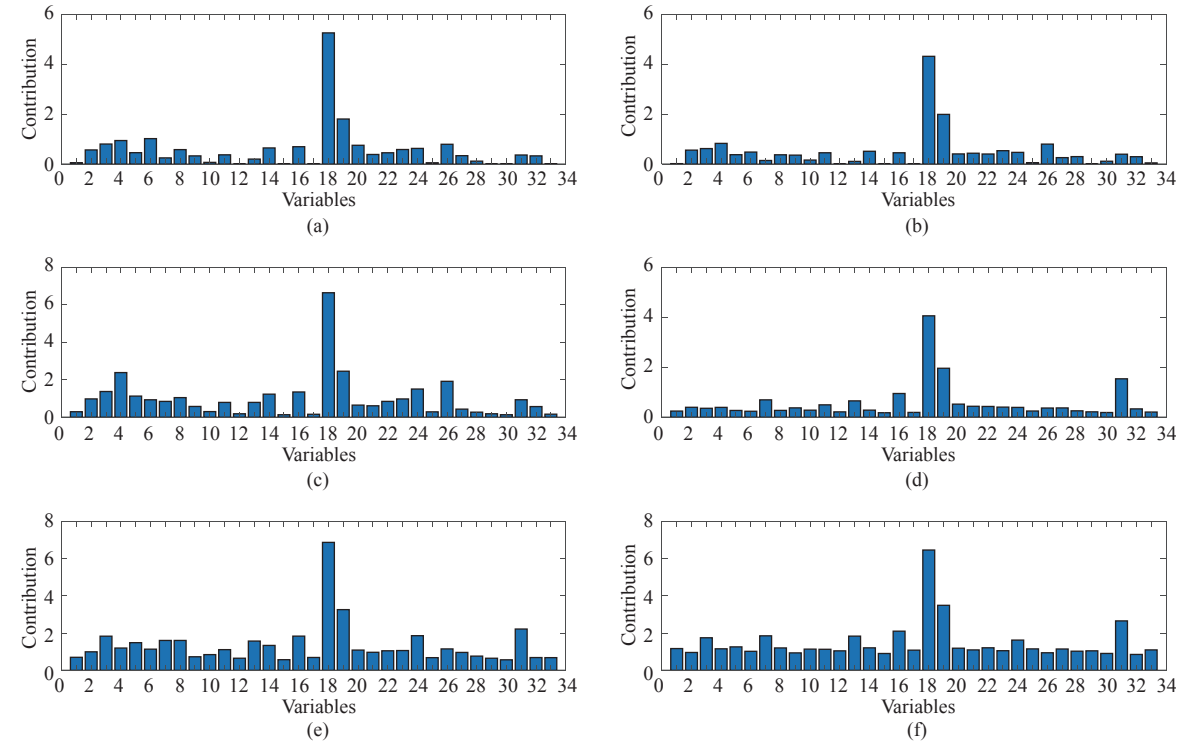

Fig. 12. Fault diagnosis of fault 10 (a) RBC; (b) BSPCA; (c) KPCA_RBC; (d) BBC; (e) LLBBC; (f) LLBBC in SAE.

variable is more obvious, which means the deeper features extracted by SAE help to provide a more convincing diagnosis result.

As for feed *C* temperature fault (fault 10), the most relevant variable is variable 18, because material *C* is sent directly to the stripper, the temperature of material *C* can be reflected in the temperature of the stripper. As shown in Fig. 12, all the 6 methods can give the correct diagnosis. And Fig. 13 illustrates the diagnosis result of the fault 14, and the key variables of fault 14 are variable 9, variable 21, and variable 32. And according to the results shown in Fig. 13, we can see that RBC and BSPCA can only identify the variable 21. All the other four nonlinear methods can give the correct diagnosis. However, it should be emphasized that LLBBC in contrast to

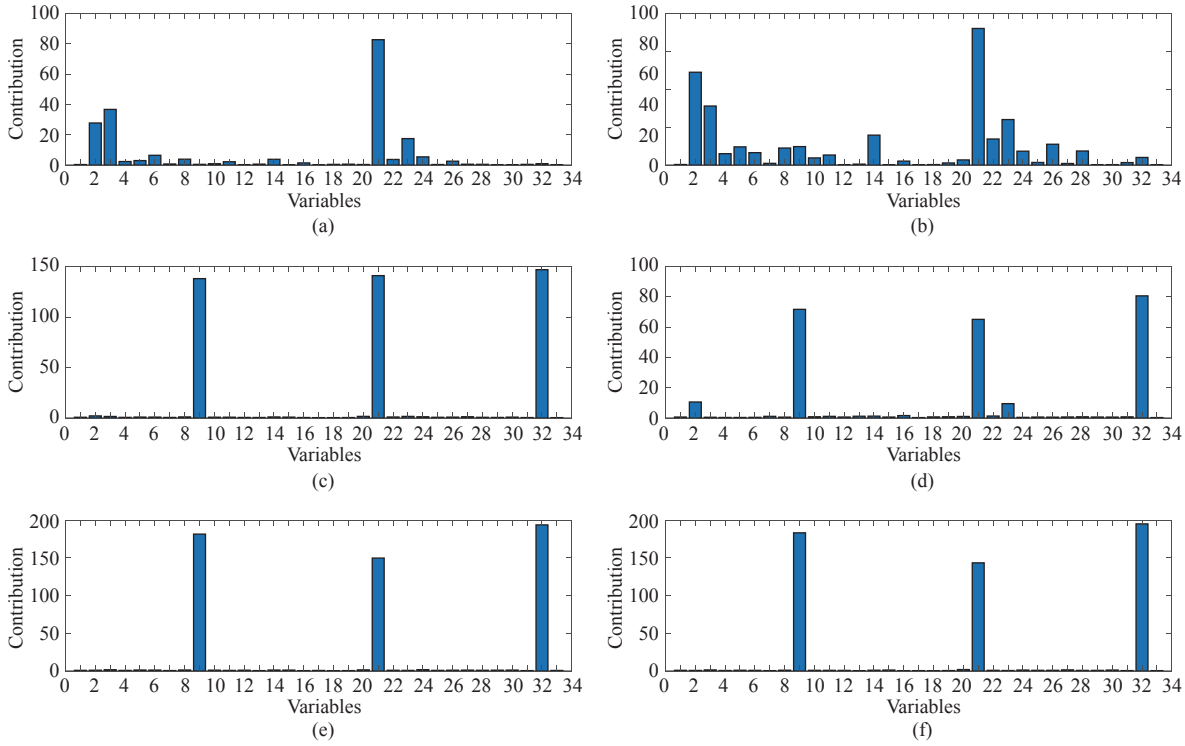

Fig. 13. Fault diagnosis of fault 14 (a) RBC; (b) BSPCA; (c) KPCA_RBC; (d) BBC; (e) LLBBC; (f) LLBBC in SAE.

BBC can perform better, for all the irrelevant variables in LLBBC have very low contribution.

The results illustrated in Figs.9–13 show that the performance of the KPCA-RBC is as good as that of LLBBC, however, after comparing the diagnostic times listed in the Table V, we can see that the time required for KPCA-RBC to diagnose one sample is about 2500 times as long as that of LLBBC. When RBC is performed in KPCA model, it needs an iterative process to get the final contribution, and in each iteration, a kernel vector needs to be calculated, so it will take a long time to get the final diagnosis results, while LLBBC can get diagnosis result immediately, once the sample is obtained. Thus, LLBBC is more suitable for the online fault diagnosis than KPCA-RBC.

In summary, it can be seen from the above fault diagnosis results that with the help of the nonlinear features extracted by AE, both BBC and LLBBC have stronger fault diagnosis capabilities than RBC and BSPCA. Besides, compared to BBC, LLBBC can give a more accurate diagnosis result, giving those irrelevant variables a lower contribution. Also, the suppression of the smearing effect can be seen in the results of LLBBC. Moreover, compared to KPCA-RBC, LLBBC has much faster diagnostic speed. Since few deeper features exist in the simulation case, the improvement given by SAE is not so obvious. However, after comparing the performance of LLBBC in AE and LLBBC in SAE, it can be seen that the performance of LLBBC in SAE is slightly better in some of the diagnosis results, for the contributions of the key variables are more obvious among the whole variables.

V. Conclusion

In this paper, a novel locally linear back-propagation based contribution is proposed for industrial process fault diagnosis. The basic idea behind this method is similar to that of the traditional RBC method. However, compared to the traditional RBC method, the LLBBC method is based on the AE model, which can use the nonlinear features extracted by the AE model to improve the fault diagnosis accuracy. Besides, instead of using the trained AE model directly, LLBBC uses a locally linear model at the current fault sample to calculate the contribution, which can always give the correct diagnosis with the smearing effect because of the special structure of the contribution. Furthermore, LLBBC can be easily used in the SAE model, which is able to extract more complex nonlinear features. The results of two case studies, including one nonlinear numerical process and the TE process show that the LLBBC owns a better diagnosis performance.

The local linearization skill used in LLBBC can also be considered as a solution to connect the RBC and nonlinear model, which can help RBC to be more suitable for the nonlinear fault diagnosis task and suppress the smearing effect at the same time.

References

[1] S. J. Qin, "Survey on data-driven industrial process monitoring and diagnosis," *Annual Reviews in Control*, vol. 36, no. 2, pp. 220–234, 2012.

[2] Z. Q. Ge, Z. H. Song, and F. R. Gao, "Review of recent research on data-based process monitoring," *Industrial & Engineering Chemistry Research*, vol. 52, no. 10, pp. 3543–3562, 2013.

[3] Z. Q. Ge, Z. H. Song, S. X. Ding and H. Biao, "Data mining and analytics in the process industry: the role of machine learning," *IEEE Access*, vol. 5, pp. 20590–20616, 2017.

[4] J.-M. Lee, C. Yoo, S. W. Choi, P. A. Vanrolleghem, and I.-B. Lee, "Nonlinear process monitoring using kernel principal component analysis," *Chemical Engineering Science*, vol. 59, no. 1, pp. 223–234,

2004.

[5] Z. Q. Ge and Z. H. Song, "Bagging support vector data description model for batch process monitoring," *J. Process Control*, vol. 23, no. 8, pp. 1090–1096, 2013.

[6] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE Journal*, vol. 37, no. 2, pp. 233–243, 1991.

[7] D. Dong and T. J. McAvoy, "Nonlinear principal component analysis-based on principal curves and neural networks," *Computer & Chemical Engineering*, vol. 20, no. 1, pp. 65–78, 1996.

[8] K. Wang, J. H. Chen, and Z. H. Song, "Fault diagnosis for processes with feedback control loops by shifted output sampling approach," *J. Franklin Institute*, vol. 355, no. 7, pp. 3249–3273, 2018.

[9] J. A. Westerhuis, S. P. Gurden, and A. K. Smilde, "Generalized contribution plots in multivariate statistical process monitoring," *Chemometrics and Intelligent Laboratory Systems*, vol. 51, no. 1, pp. 95–114, 2000.

[10] R. M. Tan and Y. Cao, "Multi-layer contribution propagation analysis for fault diagnosis," *Int. J. Autom. and Computing*, vol. 16, no. 1, pp. 40–51, 2019.

[11] R. M. Tan and Y. Cao, "Deviation contribution plots of multivariate statistics," *IEEE Trans. Industrial Informatics*, vol. 15, no. 2, pp. 833–841, 2018.

[12] C. F. Alcala and S. J. Qin, "Reconstruction-based contribution for process monitoring," *Automatica*, vol. 45, no. 7, pp. 1593–1600, 2009.

[13] C. F. Alcala and S. J. Qin, "Reconstruction-based contribution for process monitoring with kernel principal component analysis," *Industrial & Engineering Chemistry Research*, vol. 49, no. 17, pp. 7849–7857, 2010.

[14] Z. Q. Ge, M. G. Zhang, and Z. H. Song, "Nonlinear process monitoring based on linear subspace and bayesian inference," *J. Process Control*, vol. 20, no. 5, pp. 676–688, 2010.

[15] Z. B. Yan and Y. Yao, "Variable selection method for fault isolation using least absolute shrinkage and selection operator (LASSO)," *Chemometrics and Intelligent Laboratory Systems*, vol. 146, pp. 136–146, 2015.

[16] Z. B. Yan, T.-H Kuang, and Y. Yao, "Multivariate fault isolation of batch processes via variable selection in partial least squares discriminant analysis," *ISA Trans.*, vol. 70, pp. 389–399, 2017.

[17] J. X. Yu, K. Wang, L. J. Ye, and Z. H. Song, "Accelerated kernel canonical correlation analysis with fault relevance for nonlinear process fault isolation," *Industrial & Engineering Chemistry Research*, vol. 58, no. 39, pp. 18280–18291, 2019.

[18] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[19] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[20] X. F. Yuan, B. Huang, Y. L. Wang, C. H. Yang, and W. H. Gui, "Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE," *IEEE Trans. Industrial Information*, vol. 14, no. 7, pp. 3235–3243, 2018.

[21] X. F. Yuan, C. Ou, Y. L. Wang, C. H. Yang, and W. H. Gui, "Deep quality-related feature extraction for soft sensing modeling: a deep learning approach with hybrid VM-SAE," *Neurocomputing*, Apr. 2019. DOI: 10.1016/j.neucom.2018.11.107.

[22] X. F. Yuan, L. Li, and Y. L. Wang, "Nonlinear dynamic soft sensor modeling with supervised long short-term memory network," *IEEE Trans. Industrial Information*, Feb. 2019. DOI: 10.1109/TII.2019.2902129.

[23] W. W. Yan, P. J. Guo, L. Gong, and Z. K. Li, "Nonlinear and robust statistical process monitoring based on variant autoencoders," *Chemometrics and Intelligent Laboratory Systems*, vol. 158, pp. 31–40, 2016.

[24] L. Jiang, Z. H. Song, Z. Q. Ge, and J. H. Chen, "Robust self-supervised model and its application for fault detection," *Industrial & Engineering Chemistry Research*, vol. 56, no. 26, pp. 7503–7515, 2017.

[25] H. T. Zhao, "Neural component analysis for fault detection," *Chemometrics and Intelligent Laboratory Systems*, vol. 176, pp. 11–21, 2018.

[26] H. D. Shao, H. K. Jiang, K. Zhao, D. D. Wei, and X. Q. Li, "A novel tracking deep wavelet auto-encoder method for intelligent fault diagnosis of electric locomotive bearings," *Mechanical Systems and Signal Processing*, vol. 110, pp. 193–209, 2018.

[27] P. Tamilselvan and P. F. Wang, "Failure diagnosis using deep belief learning based health state classification," *Reliability Engineering & System Safety*, vol. 115, pp. 124–135, 2013.

[28] Y. L. Wang, Z. F. Pan, X. F. Yuan, C. H. Yang, and W. H. Gui, "A novel deep learning based fault diagnosis approach for chemical process with extended deep belief network," *ISA Trans.*, 2019. DOI: 10.1016/j.isatra.2019.07.001.

[29] G. Alain and Y. Bengio, "What regularized auto-encoders learn from the data-generating distribution," *The J. Machine Learning Research*, vol. 15, no. 1, pp. 3563–3593, 2014.

[30] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion," *J. Machine Learning Research*, vol. 11, no. 12, pp. 3371–3408, Dec. 2010.

[31] L. Jiang, Z. Q. Ge, and Z. H. Song, "Semi-supervised fault classification based on dynamic sparse stacked auto-encoders model," *Chemometrics and Intelligent Laboratory Systems*, vol. 168, pp. 72–83, 2017.

[32] P. R. Lyman and C. Georgakis, "Plant-wide control of the tennessee eastman problem," *Computer & Chemical Engineering*, vol. 19, no. 3, pp. 321–331, 1995.

Jinchuan Qian received the B.Eng. degree in automation from Hefei University of Technology, in 2017. He is currently working towards the Ph.D. degree at the College of Control Science and Engineering, Zhejiang University. His research interests include deep learning, process monitoring, and fault diagnosis.

Li Jiang received the B.Eng. and Ph.D. degrees from the College of Control Science and Engineering, Zhejiang University, in 2011 and 2018, respectively. He currently works for Huawei Technology Co., Ltd. His research interests include deep learning, fault detection, and fault diagnosis.

Zhihuan Song received the B.Eng. and M.Eng. degrees in industrial automation from Hefei University of Technology, in 1983 and 1986, respectively, and the Ph.D. degree in industrial automation from Zhejiang University, in 1997. Since 1997, he has been with the College of Control Science and Engineering, Zhejiang University, where he was at first a Postdoctoral Research Fellow, then an Associate Professor, and currently a Professor. His research interests include modeling and fault diagnosis of industrial processes, embedded control systems, and advanced process control technologies. He has published more than 200 papers in journals and conference proceedings.