Data-Driven Two-Dimensional Deep Correlated Representation Learning for Nonlinear Batch Process Monitoring

Qingchao Jiang, *Member, IEEE*, Shifu Yan, Xuefeng Yan, *Member, IEEE*, Hui Yi, and Furong Gao

***Abstract*—Dynamics and nonlinearity may exist in the time and batch directions for batch processes, thereby complicating the monitoring of these processes. In this article, we propose a two-dimensional deep correlated representation learning (2D-DCRL) method to achieve the efficient fault detection and isolation of the nonlinear batch processes. Three-way historical data are first unfolded as two-way time-slice data. Second, a stacked autoencoder based deep neural network is constructed to characterize the correlation among the process variables. Considering that the time and batch directions may be dynamic, for each time-slice measurement, a constructed 2-D measurement containing samples from the previous time instants and batches is then obtained. Subsequently, DCRL is performed between the current running-batch measurements and the constructed 2-D measurements to characterize the 2-D dynamics and nonlinearity. The 2D-DCRL-based monitoring examines the status of a sample by considering the 2-D nonlinear and dynamic information, providing improved monitoring performance. Applications on two typical batch processes demonstrate the effectiveness of the proposed 2D-DCRL monitoring scheme.**

***Index Terms*—Batch process monitoring, data-driven fault detection, deep correlated representation learning (DCRL), two-dimensional (2-D) modeling.**

Manuscript received June 27, 2019; revised August 16, 2019; accepted October 26, 2019. Date of publication November 11, 2019; date of current version January 17, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61603138 and Grant 61973119, in part by Shanghai Pujiang Program under Grant 17PJD009, in part by the Fundamental Research Funds for the Central Universities under Grant 222201917006, in part by the Programme of Introducing Talents of Discipline to Universities (the 111 Project) under Grant B17017, and in part by the Open Research Project of the State Key Laboratory of Industrial Control Technology, Zhejiang University, under Grant ICT1900331. Paper no. TII-19-2751.
(Corresponding author: Qingchao Jiang.)

Q. Jiang, S. Yan, and X. Yan are with the Key Laboratory of Advanced Control and Optimization for Chemical Processes of Ministry of Education, East China University of Science and Technology, Shanghai 200237, China (e-mail: qchjiang@ecust.edu.cn; yanshifu@mail.ecust.edu.cn; xfyan@ecust.edu.cn).

H. Yi is with the College of Electronic Engineering and Control Science, Nanjing University of Technology, Nanjing 211816, China (e-mail: jsyihui@126.com).

F. Gao is with the Department of Chemical and Biomolecular Engineering, The Hong Kong University of Science and Technology, Clear Water Bay 999077, Hong Kong (e-mail: kefgao@ust.hk).

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

Digital Object Identifier 10.1109/TII.2019.2952931

I. Introduction

Batch processes account for a large portion of the manufacturing industry, and the modeling and monitoring of batch processes are important [1]–[3]. Timely fault detection and isolation can facilitate the maintenance of a favorable process condition and are important for ensuring product consistency. A batch process generally comprises various physical and chemical changes. Establishing the mechanism model of such complex batch processes is generally difficult, thus complicating the usage of the model-based fault detection methods [4]–[6]. Meanwhile, the measuring and sensing techniques rapidly progress, and a modern batch process is generally equipped with various sensors. Consequently, a large amount of the process data is generally available. The exploration of process operation information from the data and the identification of the process status have gained considerable interest. The data-driven batch process monitoring techniques are gaining increasing attention [7]–[9].

Multivariate analysis (MVA) feature extraction-based methods are the most widely used data-driven monitoring methods [10]–[12]. The classical MVA methods are widely used for monitoring continuous processes. Differently, the historical batch process data are generally stored as a three-way dataset. Three dimensions, namely, the variable, time, and batch dimensions, are involved. The data-driven process monitoring is conducted by first unfolding the data and then establishing the MVA-based monitoring model. For example, a multiway principal component analysis (PCA) first unfolds the three-way data and then performs PCA to establish the monitoring model [13]. Similarly, multiway partial least squares (PLS), which first unfolds the three-way data and then performs PLS between the process data and quality variables, were developed for quality-related monitoring [13]. A time-slice canonical correlation analysis (CCA) method has been recently proposed for key unit monitoring. The time-slice CCA first unfolds the three-way historical data into the two-way time-slice data and then establishes the CCA-based monitoring model in each time instant [14]. Although good performance is obtained, the following limitations exist. First, a batch process may be characterized by process dynamics, and these methods do not consider the process dynamic information. Second, a batch process may involve remarkably nonlinearity, and these methods do not consider the nonlinearity.

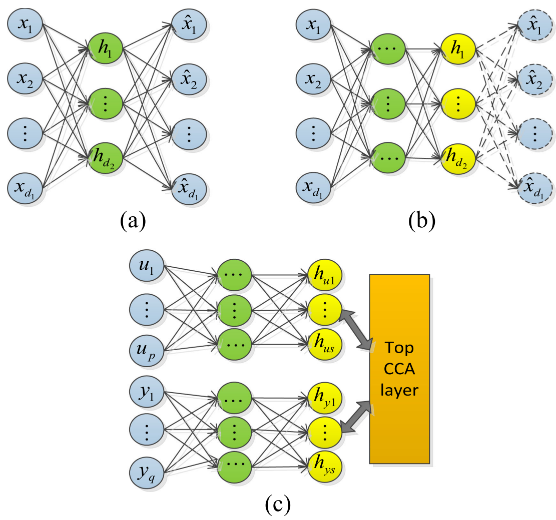


Fig. 1. Structures of (a) basic AE, (b) SAE, and (c) DCCA.

Multiway dynamic monitoring methods, including multiway dynamic PCA and multiway dynamic PLS, have been proposed to deal with the process dynamics [15]. These methods consider the correlation from the time dimension, which is called the one-dimensional (1-D) dynamic monitoring model. However, given that the batch processes are generally operated repeatedly, correlations may exist in the time and batch directions, that is, the 2-D dynamic problem. The 2-D batch process monitoring models have been developed to consider the 2-D dynamics [16]–[18]. A multiobjective 2D-CCA (M2D-CCA) method, which examines the status of a sample by considering information from the previous instants and batches, was recently proposed [19]. These methods address the dynamic issue in the batch process monitoring but are limited only in the linear correlation cases.

For monitoring nonlinear processes, the typical data-driven methods include the kernel learning and artificial neural network (ANN) methods [20], [21]. Multiway kernel learning methods have been developed for nonlinear processes [22]. However, the kernel learning methods are unsuitable for industrial big data situations because of the following reasons. First, in the kernel learning-based monitoring methods, each sample should perform kernel operation with every reference sample, which is generally computationally complex. Second, the representing ability is limited by a certain form of kernel function. Third, determining the parameters in the kernel function is difficult. The ANN-based methods are another form of the nonlinear process monitoring methods [23]. However, conventional ANNs generally have shallow structures and suffer problems in training and their representing ability.

A deep neural network (DNN) is gaining considerable attention due to its successful usage in various fields [24], [25]. Given their superior capability in extracting highly abstract representations from the industrial big data, DNNs have been introduced for process data analytics, such as process modeling, fault detection, and fault diagnosis [26]–[30]. However, learning deep representations for batch process monitoring, especially when dynamics and nonlinearity exist, has not been discussed. The key contribution of this article is the proposal of a 2-D deep correlated representation learning (2D-DCRL) method for monitoring successive batch processes considering possible 2-D dynamics. The status of a sample is examined by considering the correlation information from the previous instant and batch. The optimal time and batch lag numbers are determined by a representation correlation-based grid search method. The feasibility of the 2D-DCRL monitoring is verified on a simulated fed-batch penicillin production (FBPP) process and an industrial inject molding (IIM) process.

The rest of this article is organized as follows. In Section II, related existing techniques are reviewed, and the 2-D nonlinear dynamic monitoring problem is formulated. In Section III, the proposed 2D-DCRL batch process monitoring method is introduced. In Section IV, two experimental studies are provided. Finally, conclusions and discussions are presented in Section V.

II. Preliminaries and Motivations

A. Stacked Autoencoders (SAE) and Deep CCA

SAE is one of the basic DNNs that has been extensively used in photo, video, and audio processing fields. An SAE is constructed by several autoencoders (AEs), with each AE being a three-layer ANN, as shown in Fig. 1(a). The working mechanism of an AE comprises the encoding and decoding processes as follows [24]:

$$\boldsymbol{h} = f_e\left(\boldsymbol{W}^{(1)}\boldsymbol{x} + \boldsymbol{b}^{(1)}\right) \tag{1}$$

$$\hat{\boldsymbol{x}} = f_d\left(\boldsymbol{W}^{(2)}\boldsymbol{h} + \boldsymbol{b}^{(2)}\right) \tag{2}$$

where $\boldsymbol{x} \in \Re^{d_1}$, $\boldsymbol{h} \in \Re^{d_2}$, and $\hat{\boldsymbol{x}} \in \Re^{d_1}$ are the input, representation, and output vectors, respectively; $\boldsymbol{W}^{(1)} \in \Re^{d_2\times d_1}$, $\boldsymbol{b}^{(1)} \in \Re^{d_2\times 1}$, $\boldsymbol{W}^{(2)} \in \Re^{d_1\times d_2}$, and $\boldsymbol{b}^{(2)} \in \Re^{d_1\times 1}$ are the weight and bias parameters under the optimization during training. The optimization objective of an AE is to minimize the difference between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ as follows:

$$\{\boldsymbol{W}^*, \boldsymbol{b}^*\} = \underset{\boldsymbol{W},\boldsymbol{b}}{\operatorname{argmin}} \left\|\boldsymbol{x} - \hat{\boldsymbol{x}}\right\|^2. \tag{3}$$

An SAE treats the hidden layer of the $(i-1)$th AE as the input layer of the subsequent ith AE, as shown in Fig. 1(b). Mathematically, an SAE is constructed as follows:

$$\begin{cases} \boldsymbol{x}_{[i]} = \boldsymbol{h}_{[i-1]} \\ \boldsymbol{h}_{[i]} = f_e\left(\boldsymbol{W}_{[i]}^{(1)}\boldsymbol{x}_{[i]} + \boldsymbol{b}_{[i]}^{(1)}\right), \end{cases} \quad (i \geq 2) \tag{4}$$

With the layer-by-layer training, the top layer representations of an SAE, which is detonated as $f_x(\boldsymbol{x})$, are highly abstract and contain the most important information for reconstructing the original data. An SAE is efficient in extracting representations from a set of data; however, it does not consider the correlation between two sets of data. Deep CCA (DCCA), which is a generalization of CCA to the nonlinear form, has been developed to explore the correlation between different datasets [31]. A DCCA

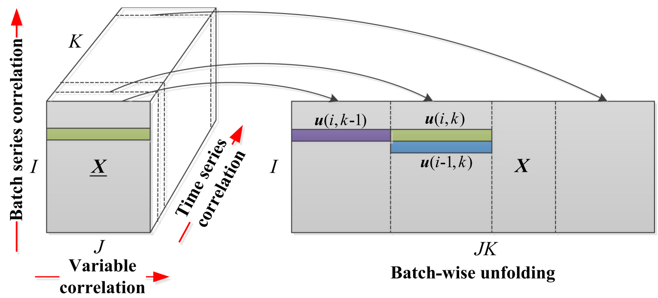


Fig. 2. Illustration of the 2-D dynamics in batch processes.

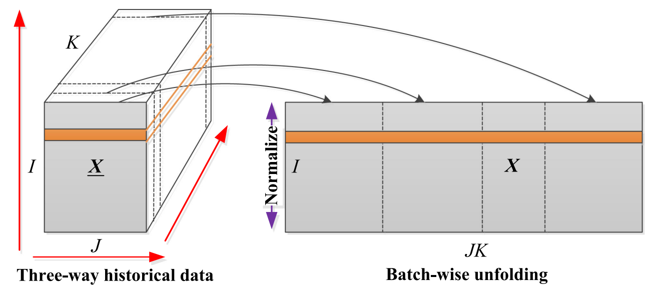


Fig. 3. Unfolding of the three-way batch process data.

comprises two SAEs and a top layer CCA neural network, which is presented in Fig. 1(c). Let $\boldsymbol{u} \in \Re^p$ and $\boldsymbol{y} \in \Re^q$ be the input of the two SAEs. The goal of a DCCA is to find all parameters Θ_1 and Θ_2 to realize the maximal correlation of the outputs of the SAEs, namely, $f_u(\boldsymbol{u})$ and $f_y(\boldsymbol{y})$. That is

$$(\Theta_1^*, \Theta_2^*) = \underset{(\Theta_1,\Theta_2)}{\arg\max}\ \mathrm{corr}\left(f_u\left(\boldsymbol{u};\Theta_1\right), f_y\left(\boldsymbol{y};\Theta_2\right)\right). \quad (5)$$

During the training process, the two SAEs are first pretrained individually (self-learning). Then, the parameters are optimized by the fine-tuning process conducted by the top layer CCA (mutual learning). Herein, the DCCA is not elaborated but the additional details are provided in [31].

B. Problem Formulation and Motivations

For a batch process, the sample of the current instant may be correlated with the samples of the previous instants and batches [16]–[19]. As illustrated in Fig. 2, a sample $\boldsymbol{u}(i,k)$ may be correlated with the sample of the previous instant $\boldsymbol{u}(i,k-1)$ and the sample of the previous batch $\boldsymbol{u}(i-1,k)$. Considering this time- and batch-series correlation is important to improve the process monitoring performance. CCA has recently been employed for the process monitoring [32], [33] and is proven optimal for monitoring a sample considering the correlation with the other samples. That is, once the relation between a set of random variables $(\boldsymbol{u})$ and another set of variables $(\boldsymbol{y})$ can be expressed as

$$\boldsymbol{A}\left(\boldsymbol{u} + \boldsymbol{\varepsilon}\right) = \boldsymbol{B}\boldsymbol{y} \quad (6)$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are the coefficient matrices, the CCA can provide the optimal solution when a fault that affects the variables in $\boldsymbol{u}$ is detected. An M2D-CCA was recently proposed to consider the 2-D dynamics [19]. The time- and batch-series correlations are expressed as follows:

$$\boldsymbol{A}\left(\boldsymbol{u}(i,k) + \boldsymbol{\varepsilon}\right) = \boldsymbol{B}\boldsymbol{y}(i,k) \quad (7)$$

where $\boldsymbol{y}(i,k) = [\boldsymbol{u}(i,k-1)^T \boldsymbol{u}(i-1,k)^T \ \cdots \ \boldsymbol{u}(i-a,k-b)^T]^T$, with a and b denoting the orders of the dynamic model. In an M2D-CCA, an optimization approach is employed to determine the optimal orders in $\boldsymbol{y}(i,k)$. Therefore, the M2D-CCA provides optimal monitoring performance for successive batch processes in form (7). However, the following issues are not addressed in M2D-CCA monitoring. First, the measured variables may be nonlinearly correlated, that is, the nonlinearity may exist among the variables, and the linear CCA may fail in exploring the nonlinear correlation. Second, the samples of the previous instants and batches may be nonlinearly correlated with the current sample, that is, the nonlinearity may exist along with the time and batch directions. Mathematically, the time- and batch-series correlation is expressed as follows:

$$\boldsymbol{A}\left(f_u\left(\boldsymbol{u}(i,k)\right) + \boldsymbol{\varepsilon}\right) = \boldsymbol{B} f_y(\boldsymbol{y}(i,k)) \quad (8)$$

where f_u and f_y can be the nonlinear functions. Linear CCA does not consider these nonlinear correlation information, and finding the optimal solution for monitoring these batch processes with the nonlinearity is recommended.

III. 2D-DCRL-Based Batch Process Monitoring

The proposed 2D-DCRL batch process monitoring comprises offline modeling and online monitoring procedures. Herein, the proposed monitoring scheme is introduced step-by-step.

A. Offline Modeling

Step 1: Data unfolding and preprocessing.

A set of data $(\underline{\boldsymbol{X}}(I \times J \times K))$ of the successive batch processes with *I* batches, *J* variables, and each batch with *K* time instants is collected. The data can be normalized with batchwise unfolding, as illustrated in Fig. 3. The three-way data are first unfolded into two-way data with dimensions $I \times JK$, and then the two-way data are normalized.

The measurement is arranged as $\boldsymbol{u}(i,k)$ for the kth time instant. The extended sample with time- and batch-lagged samples is constructed as $\boldsymbol{y}(i,k) = [\boldsymbol{u}(i,k-1)^T \boldsymbol{u}(i-1,k)^T \cdots \boldsymbol{u}(i-a,k-b)^T]^T$. The data are collected as matrices $\boldsymbol{U}$ and $\boldsymbol{Y}$. In this step, the optimal order numbers (a and b) should be determined, which is detailed in the last part of the offline modeling.

Step 2: The nonlinearity among variables is modeled using an SAE-based representation learning.

An SAE-based DNN is constructed to explore and characterize the variable relations in data $\boldsymbol{U}$, and representations are extracted as $\boldsymbol{h}$. A full-layer neural network is then added to the top layer of the SAE to reconstruct the input variables.

The following two statistics are constructed for process monitoring:

$$T_h^2(k) = \boldsymbol{h}^T(k)\,\boldsymbol{\Sigma}_h^{-1}\boldsymbol{h}(k) \tag{9}$$

$$Q(k) = \boldsymbol{r}^T(k)\,\boldsymbol{r}(k) = (\boldsymbol{u}(k) - \hat{\boldsymbol{u}}(k))^T(\boldsymbol{u}(k) - \hat{\boldsymbol{u}}(k)) \tag{10}$$

where $\boldsymbol{\Sigma}_h$ denotes the covariance. Notably, no prior assumption on the process data distribution is made. Therefore, the threshold of the statistics cannot be determined by a deterministic distribution. Herein, the kernel density estimation (KDE) is used, and the additional details are provided in [34]. Once a fault is detected, finding the most responsible variables is preferred. Given that an SAE aims to minimize the data reconstruction errors, the deviation between the input and reconstructed data can be used to evaluate the variable contributions. Then, the contribution of the i variable at time instant k is calculated as

$$\mathrm{CONT}_i(k) = |u_i(k) - \hat{u}_i(k)|. \tag{11}$$

A variable with a large deviation indicates that the variable considerably contributes to the fault, which helps in finding the fault location.

An SAE-based monitoring considers the correlation among the variables but ignores the time- and batch-series correlations. Then, the following dynamic monitoring is established based on the 2D-DCRL scheme.

Step 3: The time- and batch-series correlations are modeled with the 2D-DCRL.

$\boldsymbol{u}(i,k)$ and $\boldsymbol{y}(i,k)$ are treated as the input of a DCCA neural network to explore the correlations between them. Based on the data matrices $\boldsymbol{U}$ and $\boldsymbol{Y}$, the DCCA neural network is trained, and transformations $f_u(\boldsymbol{u}(i,k))$ and $f_y(\boldsymbol{y}(i,k))$ are obtained. Let the extracted correlated representations be $\boldsymbol{z}_u(k) = [f_{u,1}(\boldsymbol{u}(i,k)),\ f_{u,2}(\boldsymbol{u}(i,k)),\ldots,f_{u,s}(\boldsymbol{u}(i,k))]^T = [z_{u,1}(k),\ z_{u,2}(k),\ldots,z_{u,s}(k)]^T$ and $\boldsymbol{z}_y(k) = [f_{y,1}(\boldsymbol{y}(i,k)),\ f_{y,2}(\boldsymbol{y}(i,k)),\ldots,f_{y,s}(\boldsymbol{y}(i,k))]^T = [z_{y,1}(k),\ z_{y,2}(k),\ldots,z_{y,s}(k)]^T$. Then, the CCA-based modeling is performed between $\boldsymbol{z}_u$ and $\boldsymbol{z}_y$ as follows.

First, a covariance matrix Ω is constructed to include the variance–covariance information of z_u and z_y as

$$\Omega = \boldsymbol{\Sigma}_{zu}^{-1/2}\boldsymbol{\Sigma}_{zuzy}\boldsymbol{\Sigma}_{zy}^{-1/2} = \boldsymbol{R\Sigma V}^T \tag{12}$$

where $\boldsymbol{\Sigma} = \begin{bmatrix}\sigma_1 & \cdots & 0\\ \vdots & \ddots & \vdots\\ 0 & \cdots & \sigma_s\end{bmatrix} \in \Re^{s\times s}$ is a diagonal matrix, and σ_1 represents the largest correlation coefficient that can be obtained through a DCCA. Then, canonical vectors Φ_{zu} and Φ_{zy} are, respectively, obtained as

$$\Phi_{zu} = \boldsymbol{\Sigma}_{zu}^{-1/2}\boldsymbol{R} \tag{13}$$

$$\Phi_{zy} = \boldsymbol{\Sigma}_{zy}^{-1/2}\boldsymbol{V}. \tag{14}$$

For the process monitoring, a fault detection residual is constructed as

$$\boldsymbol{r}_{zu}(k) = \Phi_{zu}^T\boldsymbol{z}_u(k) - \boldsymbol{\Sigma}\Phi_{zy}^T\boldsymbol{z}_y(k). \tag{15}$$

The T^2 statistic is used to examine the status as

$$T_u^2(k) = \boldsymbol{r}_{zu}^T(k)\,\boldsymbol{\Sigma}_{rzu}^{-1}\boldsymbol{r}_{zu}(k). \tag{16}$$

According to the CCA transformation, $\boldsymbol{\Sigma}_{zu}^{-1/2}\boldsymbol{\Sigma}_{zuzy}\boldsymbol{\Sigma}_{zy}^{-1/2} = \boldsymbol{R\Sigma V}^T$; then, $\boldsymbol{\Sigma V}^T = \boldsymbol{R}^T\boldsymbol{\Sigma}_{zu}^{-1/2}\boldsymbol{\Sigma}_{zuzy}\boldsymbol{\Sigma}_{zy}^{-1/2}$ and $\boldsymbol{\Sigma}\Phi_{zy}^T\boldsymbol{z}_y(k) = \boldsymbol{\Sigma V}^T\boldsymbol{\Sigma}_{zy}^{-1/2}\boldsymbol{z}_y(k) = \boldsymbol{R}^T\boldsymbol{\Sigma}_{zu}^{-1/2}\boldsymbol{\Sigma}_{zuzy}\boldsymbol{\Sigma}_{zy}^{-1}\boldsymbol{z}_y(k)$. Thus, the following can be obtained:

$$\boldsymbol{r}_{zu}(k) = \boldsymbol{R}^T\boldsymbol{\Sigma}_{zu}^{-1/2}\left(\boldsymbol{z}_{zu}(k) - \boldsymbol{\Sigma}_{zuzy}\boldsymbol{\Sigma}_{zy}^{-1}\boldsymbol{z}_y(k)\right) \tag{17}$$

where $\hat{\boldsymbol{z}}_u(k) = \boldsymbol{\Sigma}_{zuzy}\boldsymbol{\Sigma}_{zy}^{-1}\boldsymbol{z}_y(k)$ is the least square estimation of $\boldsymbol{z}_u(k)$ using $\boldsymbol{z}_y(k)$. That is, for a batch process with $\boldsymbol{A}(f_u(\boldsymbol{u}(i,k)) + \boldsymbol{\varepsilon}) = \boldsymbol{B}f_y(\boldsymbol{y}(i,k))$, residual $\boldsymbol{r}_{zu}(k)$ has a minimal covariance, and $T_u^2(k)$ is an optimal statistic for monitoring a fault that affects the variables in $\boldsymbol{u}$. Threshold $T_{u,\mathrm{th}}^2$ of T_u^2 is also determined using the KDE method.

In constructing the 2-D measurements, determining the appropriate order numbers a and b is important. Involving too many lagged samples will lead to the redundant information in the pretraining of the SAEs, whereas involving too few lagged samples will lose useful correlation information. Given that the CCA-based monitoring is a correlation-based monitoring method, only the related samples must be included. In an M2D-CCA, a stochastic optimization algorithm is employed to determine the lagged variables involved in $\boldsymbol{y}$. However, the optimization algorithm generally involves a large computation cost, which is suitable for the linear CCA case but not for the DNN methods. Given that the correlations between the two sets are reflected by a diagonal matrix $\boldsymbol{\Sigma}$, using a grid search method to determine the appropriate numbers of a and b is proposed, as summarized in Algorithm 1.

Algorithm 1: *Determination of the Time and Batch Orders.*

Step 1: Set the time and batch orders to zero.

Step 2: Generate a set of random signals ($\boldsymbol{\eta}$), and the generated data should, therefore, have no batch- or time-series correlations. Construct a 2-D sample using only the random signals and calculate $\sigma_1^{(0,0)}$ (or the cumulative correlation coefficient instead).

Step 3: Increase the time order to 1 and perform DCCA to calculate the maximum correlation coefficient ($\sigma_1^{(1,0)}$). Compare $\sigma_1^{(1,0)}$ and $\sigma_1^{(0,0)}$. If $\sigma_1^{(1,0)}$ is smaller or approximately equal to $\sigma_1^{(0,0)}$, then no time-series correlation exists and time-series measurements need not be included. Otherwise, the dynamics along the time direction should be considered.

Step 4: Increase the time order and calculate $\sigma_1^{(2,0)}$. Compare $\sigma_1^{(2,0)}$ and $\sigma_1^{(1,0)}$. If $\sigma_1^{(2,0)} > \sigma_1^{(1,0)}$, then continue the time direction search; otherwise, maintain the optimal time order (a) and turn to the batch direction search as in Step 4.

Step 5: Increase the batch order to 1 and calculate $\sigma_1^{(0,1)}$. Compare $\sigma_1^{(1,0)}$ and $\sigma_1^{(0,0)}$. If $\sigma_1^{(0,1)} \approx \sigma_1^{(0,0)}$, then stop the batch direction search; otherwise, continue to increase the batch order until the optimal batch order (b) is obtained.

Given that the time-series correlation is significant, the grid search is suggested to start from the time direction. Furthermore, in some conditions, the correlation coefficient monotonically increases with the order number. In this condition, the order search can be stopped by setting a convergence threshold (ξ), that is, once the solutions with $|\sigma_1^{(i)} - \sigma_1^{(i+1)}| \le \xi$ are obtained, the search is stopped. A small value of ξ will make the search involve more samples.

B. Online Monitoring

Step 1: Online data preprocessing.

Normalize query sample $\boldsymbol{u}(i,k)$. Construct instant $\boldsymbol{y}(i,k)$ using the previous instant and batch samples and normalize sample $\boldsymbol{y}(i,k)$.

Step 2: Calculate the statistics.

Based on the established 2D-DCRL model, calculate statistics $T_h^2(k)$, $Q(k)$, and $T_u^2(k)$ of the query sample.

Step 3: Identify the process status.

Identify the process status using the following decision logic:

$$\begin{cases} T_h^2(k) > T_{h,th}^2 \text{ or } Q(k) > Q_{th} \text{ or } T_u^2(k) > T_{u,th}^2 \Rightarrow \text{faulty} \\ T_h^2(k) \le T_{h,th}^2 \text{ and } Q(k) \le Q_{th} \text{ and } T_u^2(k) \le T_{u,th}^2 \\ \qquad \Rightarrow \text{normal} \end{cases} . \tag{18}$$

Step 4: Identify the fault location. If a fault is detected, then look at the affected variables using the contribution plot in (11).

An illustration of the proposed 2D-DCRL-based monitoring is provided in Fig. 4. During the offline modeling procedures, DNNs are trained and monitoring statistics are established; and during the online monitoring procedures, online samples are examined by the statistics and decisions are made.

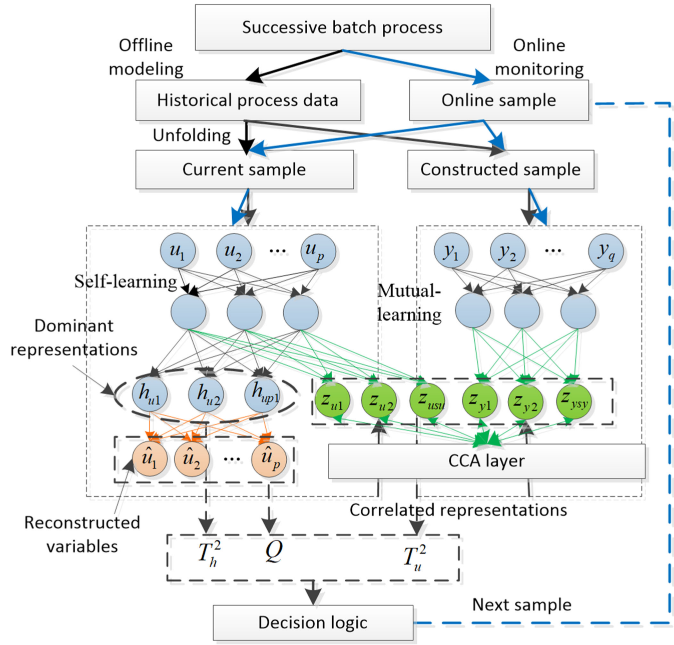


Fig. 4. Monitoring scheme of 2D-DCRL.

C. Remarks

Remark 1: On the multiple operation phases and starting point at each batch.

A batch process generally operates repeatedly, and a batch run generally comprises several operating phases. The current 2D-DCRL monitoring unfolds the data in the timewise direction and normalizes the data using the different batches, removing the multiphase influence. Meanwhile, the 2D-DCRL considers the correlation along the time and batch directions. Therefore, at the first sample of each batch run, only the batchwise correlation can be considered because the time-series correlation is generally destroyed artificially. In this situation, the monitoring scheme is designed to involve only the batchwise correlation.

Remark 2: Computation complexity analysis.

The key computation complexity lies in the offline modeling procedure. Let the dimension of the collected batch process data be $\underline{\boldsymbol{X}}(I \times J \times K)$. After the unfolding, $\boldsymbol{U}$ and $\boldsymbol{Y}$ are obtained. The dimension of matrix $\boldsymbol{U}$ is $(J \times IK)$, and the dimension of matrix $\boldsymbol{Y}$ is $(2J \times IK)$. Based on these data, a SAE and a DCCA neural network are trained. The computational complexity of these DNNs can be easily handled by the current computers. For example, for a set of batch process data with the dimensions of $100 \times 17 \times 400$, the modeling process is completed within 90 s. The computation complexity satisfies the practical application requirement.

Remark 3: Relation to the existing related methods.

The proposed 2D-DCRL method is related to some existing methods, such as M2D-CCA batch process monitoring [19], regularized correlation representation (RCR) based nonlinear process monitoring [28], and SAE-based nonlinear process monitoring [27]. Compared with the M2D-CCA monitoring, the 2D-DCRL generalizes the relation between a process sample and the constructed samples, which is suitable for the nonlinear cases. Compared with the RCR-based monitoring, the current work extracts the additional correlated representations because the RCR generates representations in an unsupervised manner. Compared with the SAE-based monitoring, the current DCRL method makes use of information from the previous instants and batches, providing superior monitoring performance.

IV. Experimental Studies

Two experimental studies are provided in this section. The computations are performed through the Python software with the hardware Intel (R) Core (TM) I7-5500U @ 2.4 GHz 16.00 G RAM.

A. Case Study 1: FBPP Process

Penicillin is an antibiotic with an extensive clinical value. Its production is a typical nonlinear, dynamic, and semibatch production process. The FBPP process is a benchmark problem designed by the Illinois Institute of Technology [35]. The benchmark has been widely used for testing the batch process modeling, monitoring, and optimization schemes. A simulator is illustrated in Fig. 5, and the software is available at the university page at [1]. Through the simulator, the process operation data, which comprise five manipulated, nine processes, and five quality variables can be obtained. Herein, 14 of these variables are considered, as listed in Table I. Each batch run is set to

[1][Online]. Available: http://simulator.iit.edu/web/software.html

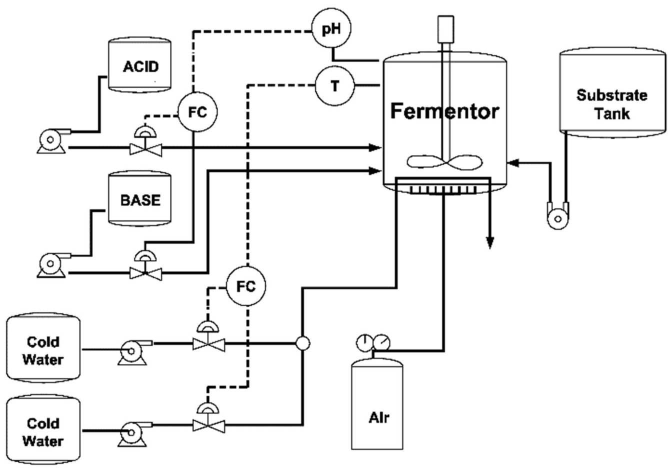


Fig. 5. Illustration of the FBPP process.

TABLE I
CONSIDERED VARIABLES IN THE FBPP PROCESS

Number	Description	Number	Description
1	Ventilation rate	8	Penicillin conc.
2	Stirring power	9	Volume
3	Substrate feed rate	10	Carbon dioxide conc.
4	Substrate feed temp.	11	pH
5	Substrate conc.	12	Temp.
6	Dissolved oxygen conc.	13	Generated heat
7	Biomass conc.	14	Cooling water flow rate

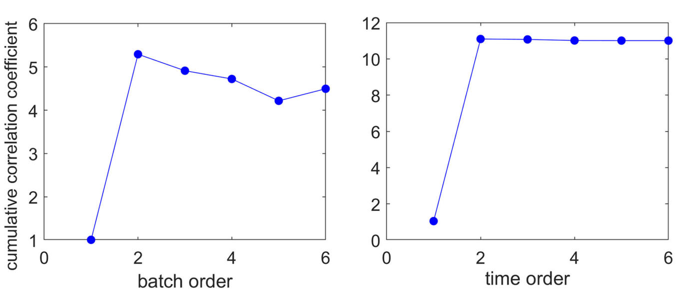


Fig. 6. Determination of the batch and time order for the FBPP.

be 200 h, and the sampling interval is 0.5 h, obtaining a set of 400 observations. Initial conditions are set the same as in [35]. Under the normal operation condition, 100 batch runs are conducted, and then a set of three-way data with the dimensions of $150 \times 14 \times 400$ is available. The process is automatic with time-series correlation. A batchwise nonlinear correlated sinusoidal signal is added to the ventilation rate of each batch to simulate the batchwise correlation. First, the optimal order search results are examined. The cumulative correlation coefficients in the search process are presented in Fig. 6(a) and (b). Compared with the randomly generated samples, a correlation exists in the time and batchwise series. Furthermore, introducing one lagged sample produces the largest correlation information. According to the results, the order is determined as one in the time and

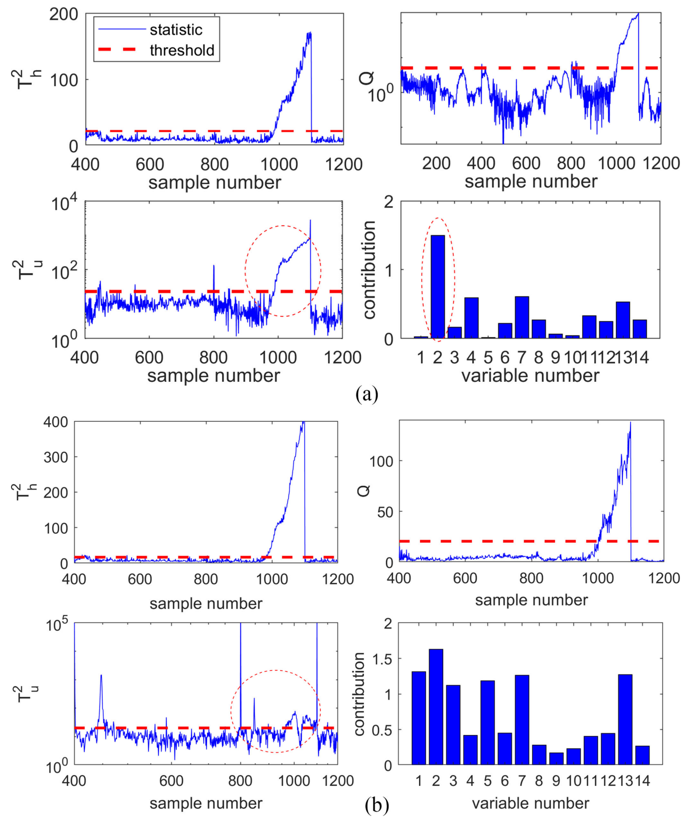


Fig. 7. Monitoring results for the FBPP fault 1. (a) 2D-DCRL. (b) 2D-PCA-CCA.

batch directions. Then the 2D-DCRL monitoring model is established. The structure of the SAE is set as 14-12-9-12-14. The structure of the 2D-DCRL neural networks are set as 14-12-10-8 and 28-25-21-15-8. The thresholds of statistics are determined through KDE by controlling the false alarm rate under 2.5%.

The simulator provides the following two predesigned faults. Fault 1: A change is introduced to the aeration rate, which will affect the dissolved oxygen concentration. Fault 2: A change is introduced to the agitator power, which will affect the dissolved oxygen concentration and temperature. Two sets of data, with each comprising four successive batches, are generated. A fault is introduced for each dataset at the fourth batch between the 151st and the 300th points. Herein, the results of the 2D-DCRL method are compared with those of the existing linear monitoring methods, that is, the multiway PCA integrated with the 2D-CCA (2D-PCA-CCA) method.

The monitoring results of Fault 1 using the 2D-DCRL and 2D-PCA-CCA methods are, respectively, presented in Fig. 7(a) and (b). The results show that both the methods can perform fault detection when the fault magnitude is large. However, at the beginning of the fault, the 2D-DCRL detects the fault much earlier than the 2D-PCA-CCA method. Especially the 2-D correlation statistic T_u^2 (highlighted by ellipses) of the 2D-DCRL detects the fault at the 990th point while the statistic of the 2D-PCA-CCA fails to detect the fault, indicating that the 2D-DCRL provides better performance than the existing

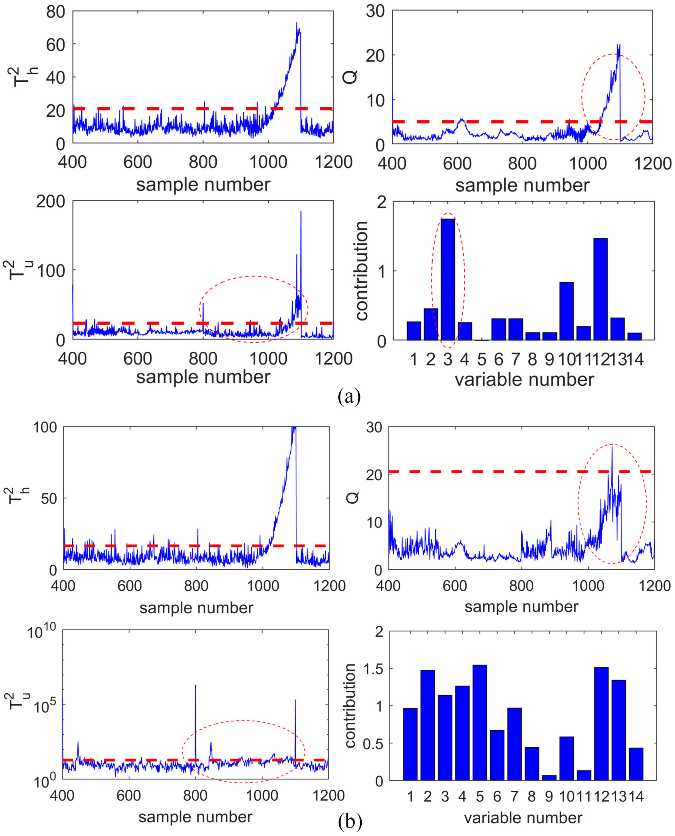


Fig. 8. Monitoring results for the FBPP fault 2. (a) 2D-DCRL. (b) 2D-PCA-CCA.

linear method. The variable contribution results (plotted at the 1000th point) are then examined. The discrimination between the normal and abnormal variables is evidently significant in the 2D-DCRL method. Similar results can be identified for Fault 2, which are presented in Fig. 8. The effectiveness of the 2D-DCRL is verified.

B. Case Study 2: IIM Process

The IIM process is a complex batch process with dynamics and nonlinearity. A general IIM machine comprises the injection device, the clamping device, the hydraulic system, and the electric control system, as illustrated in Fig. 9 [19]. With the wide use of advanced sensors, such as temperature, pressure, stroke, and screw speed sensors, the status of these measurands is available. Then, the abundant process data are available for analyzing the process status and characteristics. In the current work, 17 measured variables are considered, as listed in Table II. The normal operating data of 100 successive batches are collected, with each batch comprising 100 samples. Based on the training data, a 2D-DCRL model is established. The structure of the SAE is set as 17-15-13-11-13-15-17. The structure of the 2D-DCRL neural networks are set as 17-20-15-12 and 34-25-20-15-12. First, the optimal order searching results are examined, as illustrated in Fig. 10. The time and batch correlations clearly exist in the

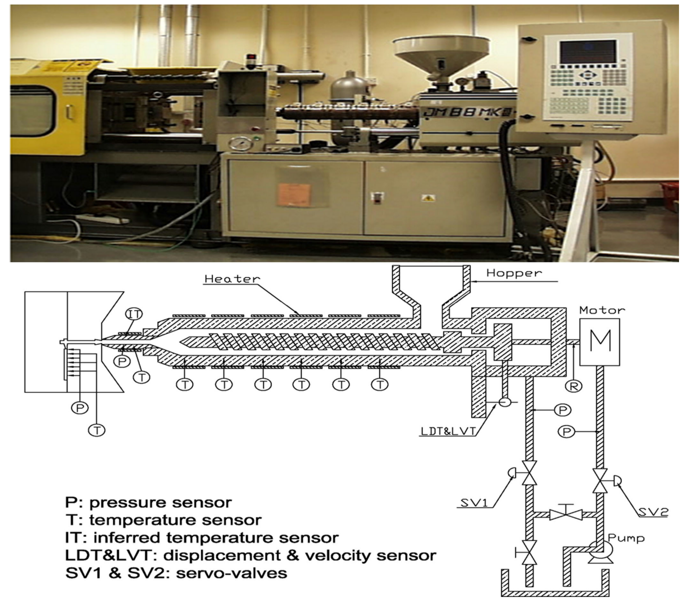


Fig. 9. Schematic of the IIM.

TABLE II
CONSIDERED VARIABLES IN THE IIM PROCESS

Number	Name	Number	Name (unit)
1	Mold position	10	Mold velocity
2	Ejector pin position	11	Ejector pin velocity
3	Injection position	12	Screw velocity
4	System press.	13	Nozzle temp.
5	Mold adjustment	14	Zone one temp.
6	Plasticization	15	Zone two temp.
7	Nozzle press.	16	Zone three temp.
8	Injection speed	17	Zone four temp.
9	Back press.		

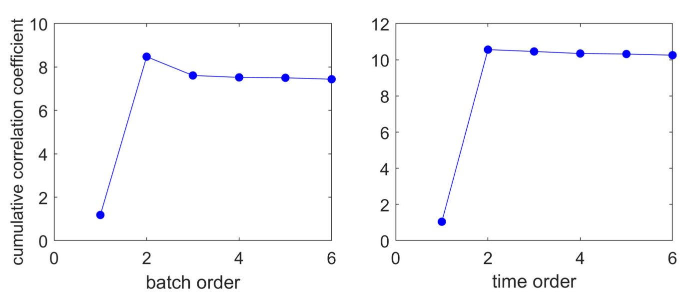


Fig. 10. Batch and time order determination for the IIM.

process data, and involving one lagged sample in the time and batch directions leads to the increased and sufficient correlation information for the data. Then, the following datasets with faults are generated.

Case 1: Four successive batches are considered, and a ramp change of $0.02*k$ is added to the temperature sensor of the nozzle between the 51st and the 100th samples of the fourth batch.

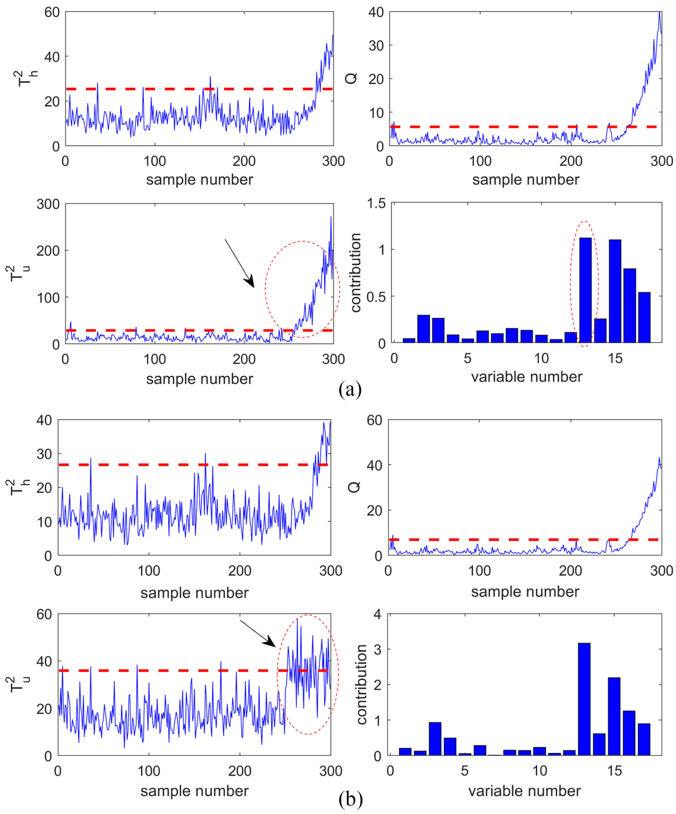


Fig. 11. Monitoring results for the IIM fault 1. (a) 2D-DCRL. (b) 2D-PCA-CCA.

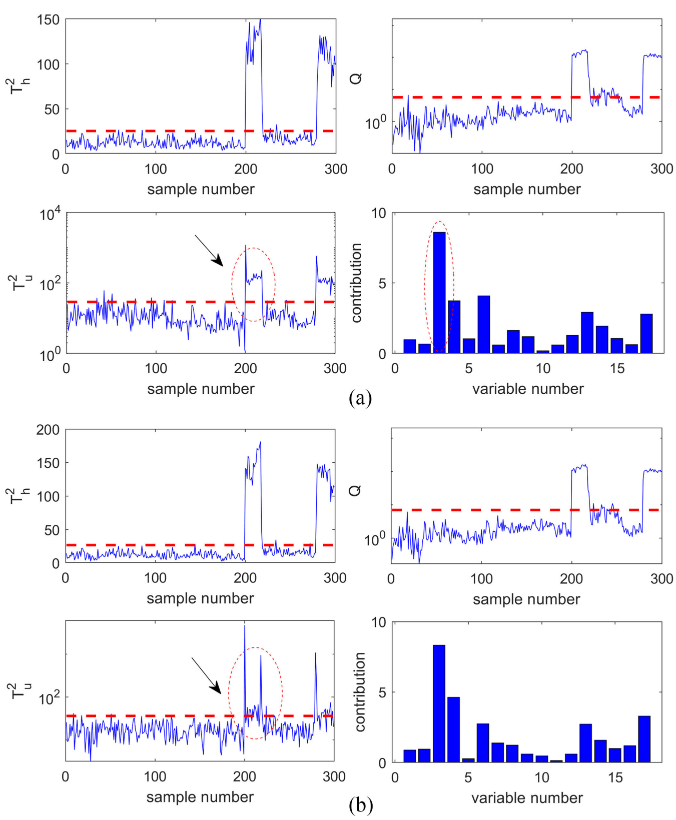


Fig. 12. Monitoring results for the IIM fault 2. (a) 2D-DCRL. (b) 2D-PCA-CCA.

Case 2: Four successive batches are considered, and the insufficient injection occurs at approximately the 20th point of the fourth batch.

Monitoring results using the 2D-DCRL and 2D-PCA-CCA for Case 1 are presented in Fig. 11(a) and (b), respectively. The two methods can evidently detect the fault. At the beginning of the fault, the 2D-DCRL detects the fault much earlier than the 2D-PCA-CCA method. Especially, the 2-D correlation statistic (T_u^2) exhibits better performance in the 2D-DCRL than in the 2D-PCA-CCA method, reflecting the improved monitoring performance. The fault isolation results (plotted at the 280th point) show that the 13th variable provides the most contributions to the fault in the 2D-DCRL method, which is consistent with the real process condition. The monitoring results for Case 2 are presented in Fig. 12(a) and (b). The contribution plots are plotted at the 210th point. The fault is evidently successively detected. It is evident that the T_u^2 of the 2D-DCRL performs better than that of the 2D-PCA-CCA. Therefore, the effectiveness of 2D-DCRL is verified.

V. Conclusion

In this article, the 2D-DCRL-based fault detection and isolation method was proposed for a successive nonlinear batch process monitoring. First, an SAE-based DNN was established to explore the nonlinear relation among the measured variables and extract dominant representations for reconstructing the measured variables. Based on the extracted representations and the generated residuals, fault detection statistics and isolation and contribution plots were constructed. Then, the 2D-DCRL method was applied between a query sample and the constructed 2-D samples to extract the correlated representations that consider the time- and batch-series correlations. Given the superior representation ability of DNNs, the complex correlations along the variable, time, and batch directions were well characterized, realizing satisfactory monitoring performance. The proposed 2D-DCRL monitoring method was applied to the simulated FBPP process and an IIM process. The monitoring results demonstrated the effectiveness and feasibility of the proposed method. Advantages of the proposed 2D-DCRL compared with some existing methods, such as the 2D-PCA-CCA, the M2D-CCA, the SAE, and the RCR, were analyzed.

Notably, the fault detection and isolation of the batch processes were studied, while the fault diagnosis was not addressed. Given that DNNs had been widely used in dealing with the classification problems, introducing DNNs to achieve batch process fault classification is promising and can be the direction of future work.

References

[1] D. Bonvin, B. Srinivasan, and D. Hunkeler, "Control and optimization of batch processes," *IEEE Control Syst. Mag.*, vol. 26, no. 6, pp. 34–45, Dec. 2006.

[2] R. Rendall, L. H. Chiang, and M. S. Reis, "Data-driven methods for batch data analysis—A critical overview and mapping on the complexity scale," *Comput. Chem. Eng.*, vol. 124, pp. 1–13, 2019.

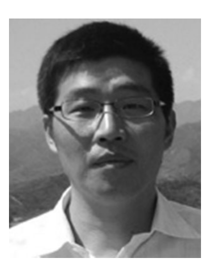
[3] T. J. Rato, J. Blue, J. Pinaton, and M. S. Reis, "Translation-invariant multiscale energy-based PCA for monitoring batch processes in semiconductor manufacturing," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 894–904, Apr. 2017.
[4] K. Peng, K. Zhang, B. You, J. Dong, and Z. Wang, "A quality-based nonlinear fault diagnosis framework focusing on industrial multimode batch processes," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2615–2624, Apr. 2016.
[5] X. Lu, C. Liu, and M. Huang, "Online probabilistic extreme learning machine for distribution modeling of complex batch forging processes," *IEEE Trans. Ind. Informat.*, vol. 11, no. 6, pp. 1277–1286, Dec. 2015.
[6] S. Zhang and C. Zhao, "Slow-feature-analysis-based batch process monitoring with comprehensive interpretation of operation condition deviation and dynamic anomaly," *IEEE Trans. Ind. Electron.*, vol. 66, no. 5, pp. 3773–3783, May 2019.
[7] D. Wang, "Robust data-driven modeling approach for real-time final product quality prediction in batch process operation," *IEEE Trans. Ind. Informat.*, vol. 7, no. 2, pp. 371–377, May 2011.
[8] S. J. Qin, "Process data analytics in the era of big data," *AIChE J.*, vol. 60, pp. 3092–3100, 2014.
[9] S. X. Ding, *Data-Driven Design of Fault Diagnosis and Fault-tolerant Control Systems*. London, U.K.: Springer, 2014.
[10] Z. Ge, Z. Song, S. X. Ding, and B. Huang, "Data mining and analytics in the process industry: The role of machine learning," *IEEE Access*, vol. 5, pp. 20590–20616, 2017.
[11] S. Yin, S. X. Ding, X. Xie, and H. Luo, "A review on basic data-driven approaches for industrial process monitoring," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6418–6428, Nov. 2014.
[12] Q. Jiang, X. Yan, and B. Huang, "Review and perspectives of data-driven distributed monitoring for industrial plant-wide processes," *Ind. Eng. Chem. Res.*, vol. 58, pp. 12899–12912, 2019.
[13] P. Nomikos and J. F. MacGregor, "Monitoring batch processes using multiway principal component analysis," *AIChE J.*, vol. 40, pp. 1361–1375, 1994.
[14] Q. Jiang, F. Gao, H. Yi, and X. Yan, "Multivariate statistical monitoring of key operation units of batch processes based on time-slice CCA," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 3, pp. 1368–1375, May 2019.
[15] J. Chen and K.-C. Liu, "On-line batch process monitoring using dynamic PCA and dynamic PLS models," *Chem. Eng. Sci.*, vol. 57, pp. 63–75, 2002.
[16] J. Flores-Cerrillo and J. F. Macgregor, "Multivariate monitoring of batch processes using batch-to-batch information," *AIChE J.*, vol. 50, pp. 1219–1228, 2004.
[17] N. Lu, Y. Yuan, F. Gao, and F. Wang, "Two-dimensional dynamic PCA for batch process monitoring," *AIChE J.*, vol. 51, pp. 3300–3304, 2005.
[18] Y. Yuan and F. Gao, "Statistical monitoring and fault diagnosis of batch processes using two-dimensional dynamic information," *Ind. Eng. Chem. Res.*, vol. 49, pp. 9961–9969, 2010.
[19] Q. Jiang, F. Gao, X. Yan, and H. Yi, "Multiobjective two-dimensional CCA-based monitoring for successive batch processes with industrial injection molding application," *IEEE Trans. Ind. Electron.*, vol. 66, no. 5, pp. 3825–3834, May 2019.
[20] J. Chen and C.-M. Liao, "Dynamic process fault monitoring based on neural network and PCA," *J. Process Control*, vol. 12, pp. 277–289, 2002.
[21] J.-M. Lee, C. K. Yoo, S. W. Choi, P. A. Vanrolleghem, and I.-B. Lee, "Nonlinear process monitoring using kernel principal component analysis," *Chem. Eng. Sci.*, vol. 59, pp. 223–234, 2004.
[22] J.-M. Lee, C. K. Yoo, and I.-B. Lee, "Fault detection of batch processes using multiway kernel principal component analysis," *Comput. Chem. Eng.*, vol. 28, pp. 1837–1847, 2004.
[23] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE J.*, vol. 37, pp. 233–243, 1991.
[24] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504–507, 2006.
[25] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.
[26] P. Bangalore and L. B. Tjernberg, "An artificial neural network approach for early fault detection of gearbox bearings," *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 980–987, Mar. 2015.
[27] G. Jiang, P. Xie, H. He, and J. Yan, "Wind turbine fault detection using denoising autoencoder with temporal information," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 1, pp. 89–100, Feb. 2018.
[28] Q. Jiang and X. Yan, "Learning deep correlated representations for nonlinear process monitoring," *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2018.2886048.
[29] 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.
[30] X. Yuan, B. Huang, Y. Wang, C. Yang, and W. Gui, "Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 3235–3243, Jul. 2018.
[31] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 1247–1255.
[32] Z. Chen, S. X. Ding, T. Peng, C. Yang, and W. Gui, "Fault detection for non-Gaussian processes using generalized canonical correlation analysis and randomized algorithms," *IEEE Trans. Ind. Electron.*, vol. 65, no. 2, pp. 1559–1567, Feb. 2018.
[33] Z. Chen, S. X. Ding, K. Zhang, Z. Li, and Z. Hu, "Canonical correlation analysis-based fault detection methods with application to alumina evaporation process," *Control Eng. Pract.*, vol. 46, pp. 51–58, 2016.
[34] J.-M. Lee, C. K. Yoo, and I.-B. Lee, "Statistical process monitoring with independent component analysis," *J. Process Control*, vol. 14, pp. 467–485, 2004.
[35] G. Birol, C. Ündey, and A. Çinar, "A modular simulation package for fed-batch fermentation: Penicillin production," *Comput. Chem. Eng.*, vol. 26, pp. 1553–1565, 2002.

Qingchao Jiang (M'17) received the B.E. degree in measurement and control technology and instrument and the Ph.D. degree in control theory and engineering from the East China University of Science and Technology, Shanghai, China, in 2010 and 2015, respectively.

He had been a Postdoctoral Fellow with the Department of Chemical and Materials Engineering, University of Alberta, Canada, a Humboldt Research Fellow with the Institute for Automatic Control and Complex Systems, University of Duisburg–Essen, Germany, and a Visiting Research Fellow with the Department of Chemical and Biomolecular Engineering, The Hong Kong University of Science and Technology. He is currently an Associate Professor with the Department of Automation, East China University of Science and Technology. His research interests include data mining and analysis, data-driven soft sensing, multivariate statistical process monitoring, and Bayesian fault diagnosis.

Shifu Yan received the B.E. degree in automation in 2017 from the East China University of Science and Technology, Shanghai, China, where he is currently working toward the Ph.D. degree in control science and engineering.

His research interests include multivariate statistical process monitoring and deep learning based process modeling.

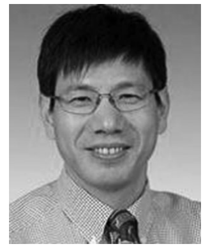


Xuefeng Yan (M'19) received the B.S. degree in biochemical engineering and the Ph.D. degree in control theory engineering from Zhejiang University, Hangzhou, China, in 1995 and 2002, respectively.

He is currently a Professor with the Department of Automation, East China University of Science and Technology, Shanghai, China. His current research interests include complex chemical process modeling, optimizing and controlling, process monitoring, fault diagnosis, and intelligent information processing.

Hui Yi received the B.E. degree in biomedical engineering and the Ph.D. degree in control theory and engineering from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2005 and 2012, respectively.

In 2012, he joined as a Lecturer with the Department of Automation, College of Electrical Engineering and Control Science, Nanjing Tech University, where he is currently an Associate Professor. His research interest covers fault diagnosis, health monitoring, and artificial intelligence.



Furong Gao received the B.E. degree in automation from the China University of Petroleum, Beijing, China, in 1985, and the M.E. and Ph.D. degrees in chemical engineering from McGill University, Montreal, QC, Canada, in 1989 and 1993, respectively.

He was a Senior Research Engineer with Moldflow International Company, Ltd. Since 1995, he has been with the Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong, where he is currently a Chair Professor with the Department of Chemical and Biomolecular Engineering. His research interests include the areas of process monitoring and control and polymer processing.