Just-In-Time Reorganized PCA Integrated with SVDD for Chemical Process Monitoring

Qingchao Jiang and Xuefeng Yan
Key Laboratory of Advanced Control and Optimization for Chemical Processes of Ministry of Education, East China University of Science and Technology, Shanghai 200237, P.R. China

DOI 10.1002/aic.14335
Published online January 16, 2014 in Wiley Online Library (wileyonlinelibrary.com)

Although principal component analysis (PCA) is widely used for chemical process monitoring, improvements in the selection of principal components (PCs) are still needed. Given that the determination of complicated and changing fault information is not guaranteed using offline-selected PCs, this study proposes a just-in-time reorganized PCA model that objectively selects the PCs online for process monitoring. The importance of the PCs is evaluated online by kernel density estimation. The PCs indicating more varied information are then selected to reorganize the PCA model. Given that the most useful fault information is concentrated, support vector data description is used to replace traditional statistics, thereby relaxing the Gaussian assumption of the process data. The monitoring performances of the proposed method are evaluated under three cases. Compared with conventional PCA methods, more varied information is captured online, and the monitoring performances are improved. © 2014 American Institute of Chemical Engineers *AIChE J*, 60: 949–965, 2014

Keywords: principal component analysis, support vector data description, process monitoring, just-in-time reorganizing

Introduction

Process monitoring is gaining significant attention because of the increasing demand in plant safety and product quality. With the development of data gathering equipment and computing technology, multivariate statistical process monitoring (MSPM) methods have rapidly progressed[1–10]; among these methods, principal component analysis (PCA) is usually the most fundamental technology. The general objectives of PCA are data reduction and interpretation. Several studies on PCA transformation in chemical engineering have been reported. Considering that direct modeling of the original process data is inappropriate when such data are highly collinear, PCA is usually used to remove collinearity.[11] In addition, PCA can explore the latent factors of the data, thereby providing better explanation and description of the process.[1,3,5,11,12] Given the simplicity and efficiency of PCA, it has been extended to kernel PCA,[13,14] dynamic PCA (DPCA),[15,16] multiway PCA,[17,18] multiscale PCA,[19,20] and among others to solve various process monitoring problems.[21–27] Although numerous successful applications of PCA for process monitoring have been reported, discussion of several problems, such as the following points, is still needed: (1) selection of retained principal components (PCs) is still an open question; (2) online process information are not considered when building a PCA model; and (3) the use of two statistics assumes Gaussian distribution of the process data.[10,28]

Correspondence concerning this article should be addressed to X. Yan at xfyan@ecust.edu.cn.

Various PC selection methods, such as cumulative percent variance (CPV), cross-validation, and variance of reconstruction error (VRE), have been reported.[28–33] CPV selects the first several PCs that represent the major variance information of original data, and given its simplicity, CPV is currently the most widely used method.[3,34] The cross-validation method considers the prediction residuals and divides the training sample into two parts for model construction and prediction.[32] VRE method indicates that when the error of fault reconstruction is at minimum, the corresponding PCs are considered with optimal number.[30,31] The fault signal-to-noise ratio method examines the relationship between the sensitivity of fault detection and the number of PCs.[35] Most of the classical methods just consider normal operational observations and select the first several PCs with larger variance; however, PCs with larger variance of normal data cannot guarantee the capture of the largest variations in fault data online. Togkalidou et al.[11,36–38] discussed the PCs selection in the inferential modeling of pharmaceutical crystallization problem and compared the results of five different PC regression (PCR) methods. They found that the most common top-down PCR may produce lower quality predictions than the other PCR methods and suggested that PCs with highest variability may not be the most informative for prediction.[11] Other examples, which demonstrate that the last PCs may be as important as those with large variance, are also available in different fields.[28,29,39] However, this issue is insufficiently discussed in PCA-based process monitoring, and the standard PC selection is still not established.

Generally, the PCA model is generated using normal training data, in which the components are constructed as linear combinations of the monitored variables. The first several

components, which represent the most variance information of the training data, construct the dominant subspace, whereas the rest of the components are left in the residual subspace. Correspondingly, T^2 and Q (also known as Squared Prediction Error) statistics are constructed to monitor the variations in the two subspaces.[3,40,41] When a fault occurs in a process, such fault usually causes one or several monitored changes in variables, and then results in some PCs with larger variation which are beneficial for monitoring the fault (informative PCs), and other PCs with less or without variation. Therefore, different faults usually cause varying changes in PCs, and these variations are not in the order, which can be determined offline by the maximum variance rule, that is, fault information has no definite mapping to a certain PC.[25,40,42] Dividing the two subspaces may also divide the beneficial information for fault detection and diagnosis into the two subspaces, leading to beneficial information dispersion. Therefore, developing a method to (1) consider the real-time process information and evaluate the importance of each PC timely, as well as to (2) reorganize the PCA model and concentrate the most informative PCs into one subspace, is desired.

Several MSPM methods that consider just-in-time process information have also been reported. DPCA has been proposed and extensively studied.[16,21] DPCA considers the real-time process information and updates the PCA model timely; however, its computational complexity is rather challenging because the PCA decomposition should be performed at every point. Jiang and Yan proposed a sensitive PCA (SPCA) method, which examines the change rate of T^2 statistic directly along each PC and selects the sensitive PCs online for process monitoring. The SPCA method concentrates most of the fault information into one subspace; however, it only selects the PCs at the beginning of the fault.[25] As the fault functions, the other PCs may change into informative PCs. Considering the change in fault information on the PCs, Jiang and Yan proposed an adaptively weighted PCA (WPCA) method, which examines the change in each of the PC and sets different weights on the PCs according to their importance.[24] The WPCA method highlights the fault information; however, the WPCA only operates on the first several PCs retained in the dominant subspace, thereby ignoring that every PC, which includes the ones with smaller variance, might be a key component in reflecting a fault. In addition, Rashid and Yu used multidimensional mutual information to evaluate the statistical dependency between the independent component subspaces of the normal benchmark and monitored datasets, and they constructed a dissimilarity index, which considers the online fault information.[43,44] This method highlights the fault information in the independent component analysis (ICA) dominant subspace; however, online ICA decomposition is also needed in such method. Ge and Song proposed a performance-driven ensemble learning ICA (EICA) method for component number selection and improvement of non-Gaussian process monitoring performance.[45] EICA measures the importance of each IC, however, EICA assumes that the fault data are available, which is rarely satisfied in practice.

The fault detection problem could be considered as a one-class classification problem because the task is to separate the faulty data samples from the normal ones. Among the classification methods, the Fisher discriminant analysis (FDA) is usually the optimal technology.[3,46,47] However, the training data from different classes is used in FDA and the faulty data are usually difficult to obtain in practical industry. Therefore, the advantage of FDA for process monitoring is the fault diagnosis.[3,46,47] Several classification methods for fault detection, such as the support vector data description (SVDD), which is a relatively new method, have been reported.[48–51] Based on normal training data only and by representing all of the normal process data samples as one class, an SVDD model can be constructed to differentiate the abnormal data samples from the normal ones. Compared with the traditional statistics in PCA monitoring, SVDD has no Gaussian limitation of the process data, which is more feasible for practical use.[48] However, several disadvantages of the SVDD-based monitoring also exist. First, the computational complexity of SVDD is increased for high-dimensional process variables set.[48] Second, the detailed process analysis and interpretation based on the SVDD is more difficult, thereby increasing the difficulties in fault diagnosis.

In the present article, a just-in-time reorganized PCA model integrated with SVDD (JIR-PCA-SVDD) is proposed to improve the performance of PCA-based process monitoring. The high-dimensional process data are first handled using JIR-PCA model. To obtain the informative PCs in time and concentrate the PCs into the dominant subspace, the probability density of each PC is examined by the kernel density estimation (KDE). First, the probability density function of each PC under normal condition is estimated using KDE with large amount of normal process data. Second, when monitoring online, the importance of each PC is evaluated and the PCs with smaller probability density values are deemed to have increased variations in information of the process. The PCs are then reordered according to the density values, and the first several PCs with smaller densities are selected to construct the dominant subspace. Given that most of the deviations in information are concentrated into the reorganized dominant subspace, the SVDD is used to replace the traditional statistics in PCA monitoring, thereby resulting in a more reliable indication of the current process status.

The rest of the article is structured as follows. First, the basics of PCA, KDE, and SVDD are briefly reviewed, followed by a motivational example that illustrates the necessities of the JIR-PCA model. Second, the JIR-PCA-SVDD is proposed for process monitoring and some details are presented. The proposed monitoring method is then tested in a numerical, continuous stirred-tank reactor (CSTR), and Tennessee Eastman (TE) processes. Finally, the conclusions are given in the last section.

Preliminaries

In this section, the basics of PCA, KDE, and SVDD are briefly reviewed, followed by a motivational example that illustrates the necessities of the JIR-PCA model.

PCA process monitoring

As a powerful tool in handling high-dimensional and highly correlated data, PCA has been widely used in MSPM. The general objective of PCA is to produce a set of new uncorrelated variables (components) that can represent the variance information of the original measured variables. Let $\boldsymbol{X} \in R^{N\times m}$ denote a scaled data matrix with zero mean and unit variance, where N is the sample number and m is the number of variables in process. Based on singular value

decomposition (SVD) algorithm (or other transformation algorithms), the matrix $\boldsymbol{X}$ can be decomposed as[3,34]

$$\boldsymbol{X}=\boldsymbol{T}\boldsymbol{P}^T+\boldsymbol{E} \tag{1}$$

where $\boldsymbol{T}\in R^{N\times k}$ and $\boldsymbol{P}\in R^{m\times k}$ are the score matrix and the loading matrix, respectively; k is the number of PCs retained; and $\boldsymbol{E}$ is the residual matrix. The number of PCs is commonly determined using CPV method, which is introduced as[3,34]

$$\sum_{i=1}^{k}\lambda_i\Big/\sum_{i=1}^{m}\lambda_i\times 100\%\geq 85\% \tag{2}$$

where λ_i is the variance of score vector. When CPV is $>85\%$, the corresponding number of PCs is determined. T^2 and Q statistics are constructed to monitor the dominant subspace and the residual subspace. Given an observation vector $\boldsymbol{x}\in R^{m\times 1}$, the T^2 statistic of the first k PCs can be calculated as[3,34]

$$T^2=\boldsymbol{x}^T\boldsymbol{P}(\boldsymbol{\Lambda})^{-1}\boldsymbol{P}^T\boldsymbol{x} \tag{3}$$

where $\boldsymbol{\Lambda}\in R^{k\times k}$ is a diagonal matrix, which denotes the estimated covariance matrix of the PC scores. The Q statistic is a squared 2-norm of the deviation of the observation from the first k PCs. Q can be calculated as[3,41]

$$Q=\boldsymbol{e}^T\boldsymbol{e},\ \boldsymbol{e}=(\boldsymbol{I}-\boldsymbol{P}\boldsymbol{P}^T)\boldsymbol{x} \tag{4}$$

where $\boldsymbol{e}$ is the residual vector. Determination of the confidence limits of the two statistics is based on the assumption that the process data are Gaussian distributed.

Kernel density estimation

KDE is an effective approach for nonparametric density estimation.[52–54] It is widely used in determining the control limits when the data distribution is nonnormal or unknown. A univariate kernel estimator with the kernel function K_1 is defined by[52,54]

$$\hat{f}(t)=\frac{1}{nh}\sum_{i=1}^{n}K_1\left\{\frac{t-t(i)}{h}\right\} \tag{5}$$

where t is the data point under consideration; $t(i)$ is an observation value from the normal training dataset; h is the window width, and n is the number of observations. The kernel function K_1, which determines the shape of the bumps, has a number of possible forms, in which the Gaussian kernel function is the most commonly used.[55] In the present study, the Gaussian kernel is also used and the kernel estimator becomes[52,54]

$$\hat{f}(t)=\frac{1}{n}\frac{1}{h\sqrt{2\pi}}\sum_{i=1}^{n}\exp\left(-\frac{(t-t(i))^2}{2h^2}\right) \tag{6}$$

In KDE, the window width h usually has a crucial effect on the performance of density estimation. If h is significantly small, the density estimator is constructed with sharp peaks, which are positioned at the sample points. If h is significantly large, the density estimate is overly smooth and the structure in the probability density estimate is lost. The optimal choice of h depends on several factors, such as the number of data points, data distributions, and choice of the kernel function. Sufficient research on the choice of h has been reported, in which a proper empirical value for each specific case is suggested.[55] Numerous studies on the KDE[52–55] claimed that a satisfactory performance of KDE can be obtained with sufficient information on the training data.

Support vector data description

SVDD tries to find a sphere with minimum volume containing all (or most) of the data objects. To model a nonlinear process, a nonlinear transformation function Φ, which can transform the data into higher dimensions, is introduced and the SVDD is to solve the following optimization problem[48,49]

$$\begin{aligned}&\min_{R,\boldsymbol{a},\xi}\ R^2+C\sum_{i=1}^{n}\xi_i\\&\text{s.t.}\ \ \|\Phi(\boldsymbol{y}_i)-\boldsymbol{a}\|^2\leq R^2+\xi_i\end{aligned} \tag{7}$$

where $\boldsymbol{a}$ is the center of the hypersphere; C shows the tradeoff between the volume of the hypersphere and the number of errors; R^2 denotes the squared distance from the center $\boldsymbol{a}$ to the boundary; and ξ_i represents the slack variable, which allows a probability that some of the training samples can be wrongly classified. The dual form of the optimization problem can be obtained as[48,49]

$$\begin{aligned}&\min_{\alpha_i}\sum_{i=1}^{n}\alpha_i K_2\left(\boldsymbol{y}_i,\boldsymbol{y}_j\right)-\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j K_2\left(\boldsymbol{y}_i,\boldsymbol{y}_j\right)\\&\text{s.t.}\ \ 0\leq\alpha_i\leq C,\sum_{i=1}^{n}\alpha_i=1\end{aligned} \tag{8}$$

where $K_2(\boldsymbol{y}_i,\boldsymbol{y}_j)=\langle\Phi(\boldsymbol{y}_i),\Phi(\boldsymbol{y}_j)\rangle$ is a kernel function, which is introduced to compute the inner product in the feature space, and α_i is a Lagrange multiplier. Solving Eq. 8 gives a set α_i, and the samples $\boldsymbol{y}_i$ with $\alpha_i>0$ are called support vectors (SVs) of the description. The squared distance from the center $\boldsymbol{a}$ to the boundary R^2 is calculated as follows[48,49]

$$R^2=K_2(\boldsymbol{y},\boldsymbol{y})-2\sum_{i=1}^{n}\alpha_i K_2(\boldsymbol{y}_i,\boldsymbol{y})+\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j K_2(\boldsymbol{y}_i,\boldsymbol{y}_j) \tag{9}$$

for any $\boldsymbol{y}\in$ SV.

Problem statement and motivational example

In PCA monitoring, the PC selection is usually operated as follows: first, the components are arranged according to the corresponding eigenvalues; second, the number of retained PCs is determined according to some rules, such as CPV, cross validation, or VRE. In any selected rule, the first several PCs, which represent information with the most significant variance of the training data, are used to construct the dominant subspace. However, the training data are collected under normal condition and these PCs do not guarantee capture of the largest online process variation. The selection of PCs offline is subjective.

Meanwhile, when a fault occurs in a process, the fault usually causes one or several variations in the monitored variables, and then after the PCA mapping, such variations are reflected on one or more PCs. Some PCs with larger variation, which contain more fault information, are beneficial to determine fault timely. In addition, the PCs with less variation or without variation contain less or no fault information, and they are not used in finding fault. When dividing the

PCs into two subspaces, the fault information might also get divided and mapped into two subspaces. If the information for finding a fault is limited, such an information loss could be damaging and can directly lead to fault nondetection. Therefore, selecting the most informative PCs and concentrating the beneficial information for monitoring the fault into a specific subspace for process monitoring are important.

To analyze the monitoring performance of PCA and illustrate the necessity of the reorganization, the following simple numerical multivariate process is used

$$V=\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}=\begin{bmatrix} 1.5\eta \\ 1.8\eta \\ 1.3\eta \end{bmatrix}+\begin{bmatrix} 0.28e \\ 0.385e \\ 0.42e \end{bmatrix}$$

$$U=\begin{bmatrix} 1.57 & 2.37 & 1.8 \\ 2.73 & 1.05 & 1.4 \\ 1.22 & 1.60 & 2.4 \\ 1.65 & 2.2 & 1.5 \end{bmatrix}V+\begin{bmatrix} 0.35e \\ 0.42e \\ 0.315e \\ 0.49e \end{bmatrix}$$

where $\eta, e \sim N(0,1)$. This simple linear process has seven monitored variables $[x_1, x_2, \ldots, x_7]^T = [\boldsymbol{V}^T \ \boldsymbol{U}^T]$ and three factors, namely, v_1, v_2, and v_3. For illustration purposes, normal data with 200 samples are generated and used for building the conventional PCA model. In PCA, the first three PCs, which occupy >95% cumulative variance, are retained in the dominant subspace. The simulated faults for monitoring are created as:

Fault 1: a step change of 2.5 is introduced to x_3 at the 201th sample, and simultaneously, a step change of 1.22 is introduced to x_4.

Fault 2: a step change of 4 is added to x_1 and a ramp change of $0.02(i-200)$ is added to x_4 from the 201th sample (i is the sample number).

The PCA monitoring results of the Fault 1 are plotted in Figure 1. From Figure 1, we can see that Fault 1 can be

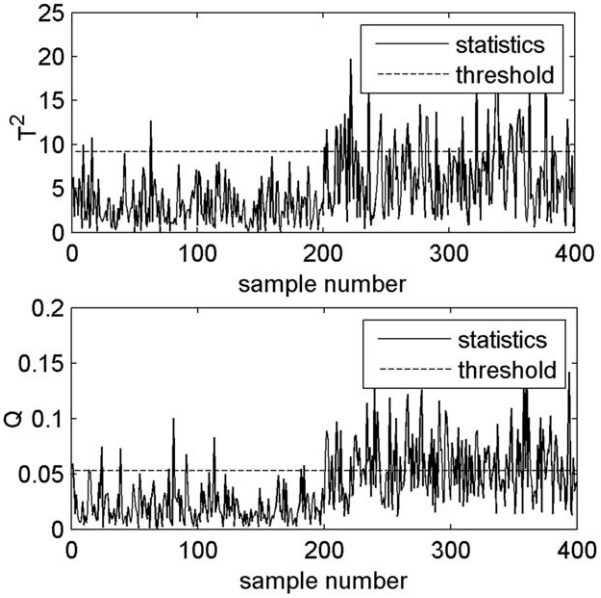

Figure 1. Monitoring results of simple Fault 1 using PCA.

detected by the two statistics. However, the statistics do not stay above the confidence limits, thereby resulting in high nondetection rates (the percentage of faulty samples classified as normal, Type II error). To analyze the reason further, the T^2 statistic along each PC is constructed as[25]

$$T_i^2 = \boldsymbol{x}^T \boldsymbol{p}_i (\lambda_i)^{-1} \boldsymbol{p}_i^T \boldsymbol{x} \tag{10}$$

where $i=1,2,\cdots,7$; $\boldsymbol{p}_i$ is the ith loading vector, and λ_i is the ith eigenvalue of $\boldsymbol{X}^T\boldsymbol{X}$. T_i^2 statistic directly monitors the variation along each PC. The monitoring results along each PC are illustrated in Figure 2.

Figure 2 shows that when the fault occurs, the PCs 3, 6, and 7 change substantially. Therefore, most of the fault information, which are beneficial for detecting the fault, are reflected on these PCs. However, the three PCs are divided into two subspaces; as a result, the fault information is also divided into two subspaces, leading to dispersion of beneficial information. Meanwhile, in each subspace, the other

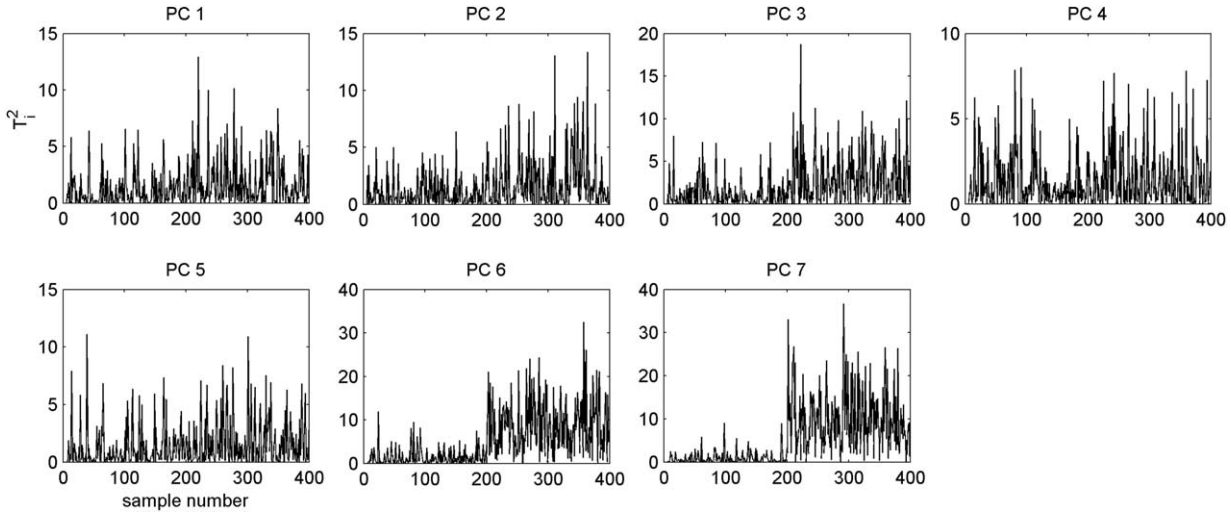

Figure 2. T^2 monitoring results for Fault 1 along each PC.

PCs do not show significant deviation from the normal condition, and the information reflected on the three PCs might be suppressed. Such dispersed and submerged information may lead to poor performance in the monitoring. Therefore, finding the most important PCs during online monitoring and concentrating the most informative PCs into one subspace are important. The motivational example and two faults described are utilized in subsequent sections for further analysis.

JIR-PCA-SVDD for Process Monitoring

In this section, the proposed JIR-PCA-SVDD method is presented in detail. Some characteristics of JIR-PCA-SVDD are discussed.

JIR-PCA-SVDD

Given a set of normal training data, the PCs can be obtained using SVD algorithm. The components, which occupy 80% or 95% variance contributions, are usually selected as the PCs retained in dominant subspace. However, the PCs with smaller variance may also reflect large amount of fault information. Therefore, the cumulative variance should be increased, even up to 100%. However, in the SVD decomposition, some eigenvalues, which are significantly small and close to zero, are found. In PCA monitoring, the PCs are scaled by dividing them by their corresponding eigenvalues, and if the eigenvalues are much smaller or near to zero, such PCs are more easily influenced by the process noise. Therefore, in the current study, the PCs occupying 98–99.5% cumulative variance are retained as basic PCs, and the PCs with much smaller variance (or the corresponding eigenvalues are near to zero) are rejected. Theoretically, this selection causes 0.5–2% information loss, but the influence of process noise is significantly reduced or avoided, which is an important consideration in practical application. Based on the basic PCs, the PCA model are timely reorganized.

Using the normal benchmark data, the probability density function of each PC score can be estimated using KDE. In online monitoring, the current sample point is initially projected onto the basic PCs subspace, and then the probability density value of each PC score can be estimated according to the obtained probability density function. For a continuous process, given that the process data are mean-variance normalized, the PCs would usually deviate on both sides of zero. Given a confidence interval, the points within the interval could be regarded as normal and usually own relatively larger density values. However, the abnormal points that usually have small densities (as illustrated in the following sections) are found outside the interval. In such situation, for a definite fault, the PCs with relatively smaller densities more likely indicate deviations from the normal condition and contain more fault information. These PCs are regarded as informative PCs, and are beneficial for finding fault timely. If the most informative PCs are concentrated into one subspace, the fault detection and diagnosis is more efficient. The formulation of JIR-PCA-SVDD consists of offline modeling and online monitoring.

Offline modeling

1. Based on the normal training data $\boldsymbol{X} \in R^{N\times m}$, the basic PCs $\boldsymbol{T} \in R^{N\times l}$ can be obtained using SVD algorithm as

$$\boldsymbol{T}=[\boldsymbol{t}_1,\boldsymbol{t}_2,\ldots,\boldsymbol{t}_l]=\boldsymbol{XP} \tag{11}$$

where $\boldsymbol{P} \in R^{m\times l}$ is the loading matrix that projects the monitored data onto the PCs, and l is the number of basic PCs retained that occupy 98–99.5% cumulative variance. Each column in $\boldsymbol{P}$ is a loading vector that determines the projection relationship between the observed data and the corresponding PC. The loading matrix $\boldsymbol{P}$ could be denoted as

$$\boldsymbol{P}=[\boldsymbol{p}_1,\boldsymbol{p}_2,\ldots,\boldsymbol{p}_l] \tag{12}$$

2. The PCs are scaled to the same level using the estimated standard variances of the PCs. That is

$$\boldsymbol{t}_i=\frac{\boldsymbol{x}^T\boldsymbol{p}_i}{\sqrt{\lambda_i}} \tag{13}$$

where $\boldsymbol{x}=[x_1,\ldots,x_m]^T \in R^{m\times 1}$ is the measured variables and λ_i is the variance of ith score vector.

3. The probability density function of each basic PC is estimated using KDE, which is denoted as

$$\boldsymbol{f}=[f_{t1},f_{t2},\ldots,f_{tl}] \tag{14}$$

The window width h in KDE usually has crucial effect on the performance of density estimation. However, given that the training data of normal conditions is easily obtained, a satisfactory performance of KDE can be achieved with sufficient information in the training data.

4. The number of PCs that should be retained in the dominant subspace of the reorganized PCA is determined. Given that the most beneficial information is concentrated, the number of PCs does not need to be significantly large. Generally, the number that results 75–85% CPV occupation of the PCs in the original PCA model can be used, thereby effectively reducing data dimension.

5. The reorganized PC scores of the normal training data are calculated. For the jth sample, the density values of the point that belong to the normal set could be calculated using the obtained density function. That is, the probability density value of the PC score $t_i(j)$ is calculated as

$$f_{ti(j)}=f_{ti}(t_i(j)) \tag{15}$$

Given that the density value is an estimation, the status of previous samples can be considered, and then the mean value of the previous several samples could be used. That is

$$f_{ti(j)}=f_{ti}(\overline{t_i(j)}) \tag{16}$$

where $\overline{t_i(j)}$ is calculated as

$$\overline{t_i(j)}=\frac{1}{w}\sum_{r=j-w}^{r=j} t_i(r) \tag{17}$$

where w is a window width. This moving window could consider the time series information and significantly reduce the influence of process noise. For a definite process, the selection of the window width could be determined based on the normal training data. One helpful rule is to ensure that the effect of process noise is acceptable. The recommended values of the w are 6 or 7 in the current study.

6. The PC scores $t_i(j)$ are rearranged according to the density values. The PCs with smaller densities are retained in the reorganized dominant subspace as follows

$$\boldsymbol{t}_r(j)=[t_{r1}(j),t_{r2}(j),\ldots,t_{rk}(j)]\left(f_{tr1,j}\le f_{tr2,j}\le\ldots\le f_{trk,j}\right) \quad (18)$$

7. The SVDD model is constructed based on the reorganized PCs of the normal training data. The reorganized PC scores are used as the input variables to the SVDD model. That is

$$\boldsymbol{Y}=[\boldsymbol{y}_1,\boldsymbol{y}_2,\ldots,\boldsymbol{y}_k]^T=[\boldsymbol{t}_{r1},\boldsymbol{t}_{r2},\ldots,\boldsymbol{t}_{rk}]^T \quad (19)$$

Given that SVDD aims to differentiate the abnormality from normal samples, the assumption that the data should follow certain distribution is not needed, which is suitable for the current study. The center $\boldsymbol{a}$ of the hypersphere and the boundary R^2 can be determined by solving Eqs. 7–9. The boundary R^2 varies with the reorganization, and the scores with different PCs provide different boundary R^2. The boundary is calculated online for each reorganization. However, given that the process data are normalized and the scores are scaled to the same level, the center and boundary of the hypersphere is relatively identical eventhough the used PCs vary. This observation is similar to that of the calculation of T^2 statistic. The hypersphere is influenced only by the number of PCs used, not by the type of PCs. Therefore, the hypersphere that is obtained directly using the reorganized scores of normal training data is reliable, thereby significantly reducing the online computational complexity.

Online reorganizing and monitoring

1. The current sample data are normalized and projected onto the basic PCs.
2. The corresponding probability density values of the PC scores are calculated using the obtained density function.
3. The PCs are reordered according to the densities, thereby reorganizing the PCA model. The PCs with smaller probabilities, which indicate larger deviation from the normal condition, are selected to reorganize the PCA dominant subspace. Suppose the reorganized dominant PC scores at the sample point j are

$$z(j)=[t_{r1}(j),t_{r2}(j),\ldots,t_{rk}(j)]\left(f_{tr1,j}\le f_{tr2,j}\le\ldots\le f_{trk,j}\right) \quad (20)$$

4. For the z obtained online, the squared distance from the center of the sphere is calculated as

$$D^2=\|z-\boldsymbol{a}\|^2=K_2(z,z)-2\sum_{i=1}^{n}\alpha_iK_2(z,\boldsymbol{y}_i)+\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_jK_2(\boldsymbol{y}_i,\boldsymbol{y}_j) \quad (21)$$

For fault detection purpose, the index is defined as follows

$$DR=\frac{\|z-\boldsymbol{a}\|^2}{R^2} \quad (22)$$

and the control limit $DR_l=1$. That is, the sample z is accepted when the distance is $\le R^2$; otherwise, the sample z is rejected and regarded as a fault point.

Fault isolation

After fault detection, the root cause and responsible variables are then determined. In PCA monitoring, the contribution plot is the most widely used method. The contribution plot method is described in other references[3,41]; however, only the PCs with T^2 values exceeding the control limit should be used. Therefore, the calculation of the contributions still relies on the Gaussian assumption, which can be rarely satisfied in practical use.

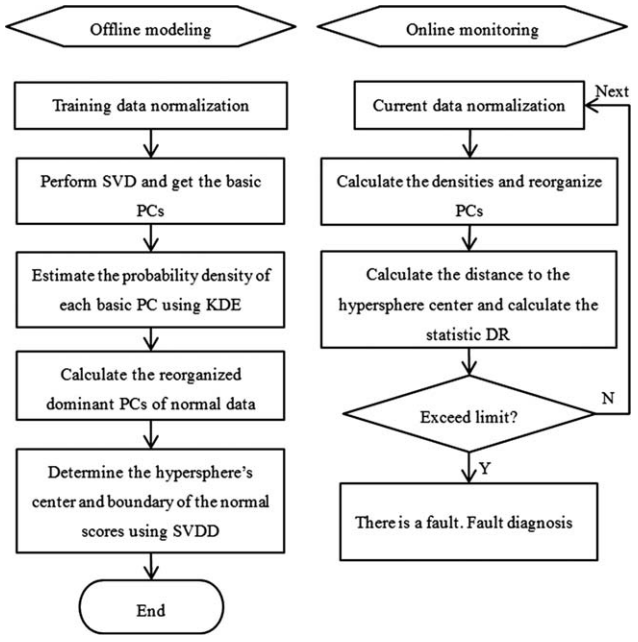

Figure 3. Illustration of the steps of JIR-PCA-SVDD monitoring.

In the current study, given that the most informative PCs are concentrated, all of the PCs in the reorganized PCA dominant subspace are used to calculate the variables' contributions. The contribution of each variable $x_j(j=1,2,\ldots,m;$ m is the number of variables) in the reorganized dominant score t_i $(i=1,2,\ldots,k;$ k is the number of retained PCs) is[3]

$$\text{cont}_{i,j}=\frac{t_i}{\lambda_i}p_{i,j}(x_j) \quad (23)$$

where $p_{i,j}$ is the (i,j)th element of the loading matrix $\boldsymbol{P}$. The total contribution of the jth process variable is[3]

$$\text{CONT}_j=\sum_{i=1}^{k}(\text{cont}_{i,j}) \quad (24)$$

Given that the most useful PCs for describing the fault are collected, the fault information is enhanced, and therefore, the fault information would not be easily influenced by process noise. Thus, the contribution plots provide more reliable results. The detailed procedures of JIR-PCA-SVDD process monitoring are summarized and illustrated in Figure 3. In addition, considering that slight drift in the characteristics of variables usually exists, the offline modeling should be updated according to the practical necessity for slow time varying in a practical process.

Characteristics of JIR-PCA-SVDD

Compared with the classical PCA, computation of the density values of each basic PC is needed in the JIR-PCA. However, the probability density functions are estimated offline, and the online computational complexity is a simple numerical calculation that can be handled in a reasonable timeframe. Moreover, given that the dimension is significantly reduced by JIR-PCA, the computational complexity in SVDD is also significantly reduced.

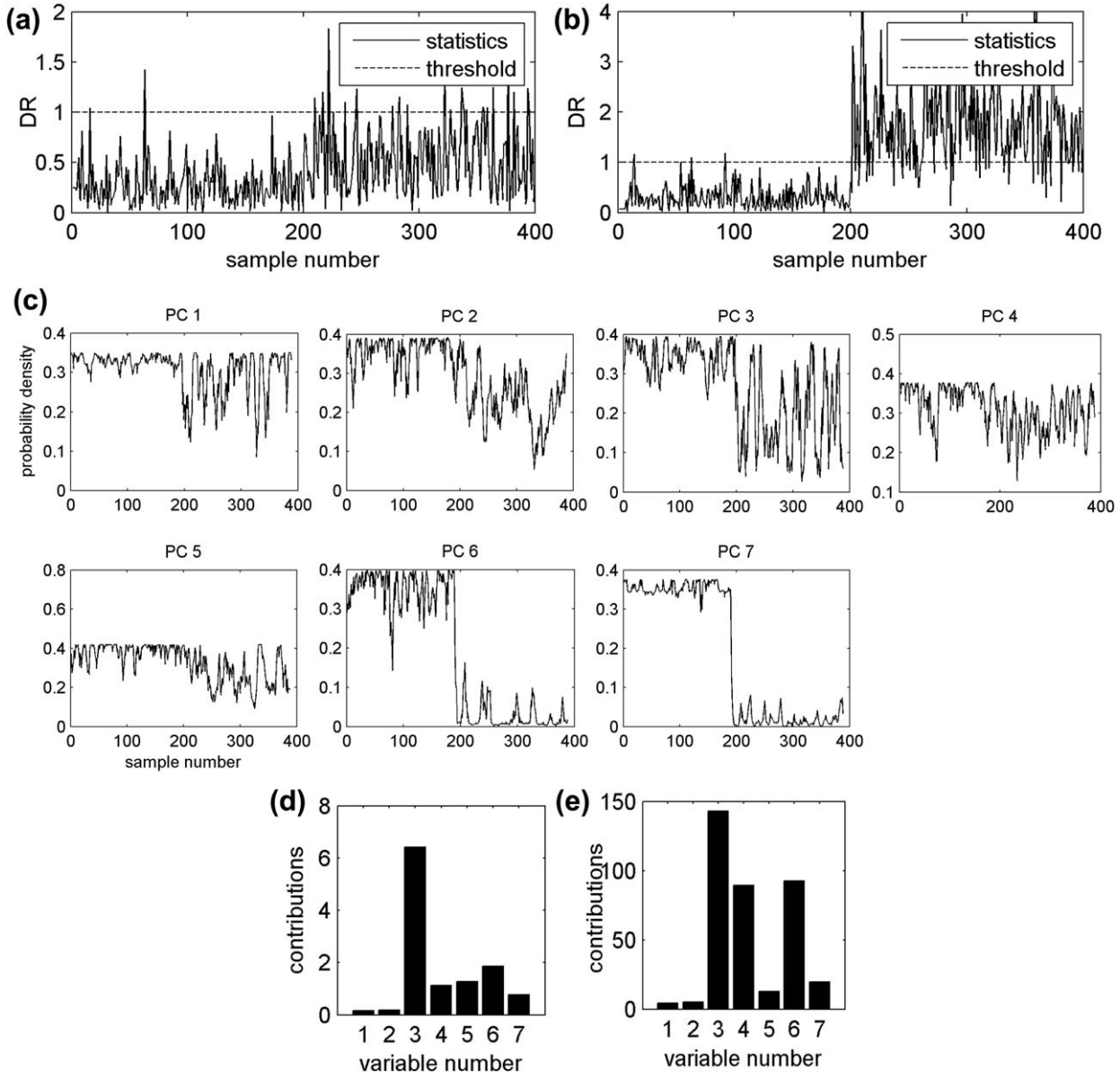

Figure 4. (a) PCA-SVDD monitoring results; (b) JIR-PCA-SVDD monitoring results; (c) estimated density values along each PC; and (d) PCA and (e) JIR-PCA contribution plots for Fault 1.

JIR-PCA selects PCs with larger variation to reorganize the PCA model. However, the monitoring performance is not influenced by the process noise because of the following reasons: in offline modeling, the process noise level reflected on each PC is considered with the use of KDE to estimate the data distribution; the previous several points are used to comprehensively evaluate the importance of the PC, and the time series relationships are then considered; and the PCs with the smallest eigenvalues are also not examined in the JIR-PCA model. JIR-PCA-SVDD shows reliable monitoring performance considering the process noise.

The main advantages of the JIR-PCA-SVDD monitoring method are given. First, the JIR-PCA-SVDD can efficiently concentrate the most varied information in a process. Therefore, the problem on fault information dispersion is solved, thereby providing efficient methodology for fault detection and diagnosis. Second, the Gaussian assumption of the process data is relaxed, which is more suitable for practical use, because of the use of SVDD.

Examples and Applications

In this section, the proposed JIR-PCA-SVDD method is applied in the numerical, CSTR, and TE benchmark processes. The JIR-PCA-SVDD method is compared with the conventional PCA and PCA-SVDD.

Numerical process

The numerical case used in the current study has been presented in the motivational example section. The PCA monitoring results for Fault 1 have been shown to be poor, and now the PCA-SVDD and JIR-PCA-SVDD methods are examined. The monitoring results of PCA-SVDD and JIR-PCA-SVDD for Fault 1 are presented in Figures 4a, b. Figure 4a shows the monitoring results of PCA-SVDD, whereas Figure 4b shows the monitoring results of the proposed JIR-PCA-SVDD. Compared with the conventional PCA and PCA-SVDD, the JIR-PCA-SVDD has shown better monitoring results. The fault was detected timely, and the

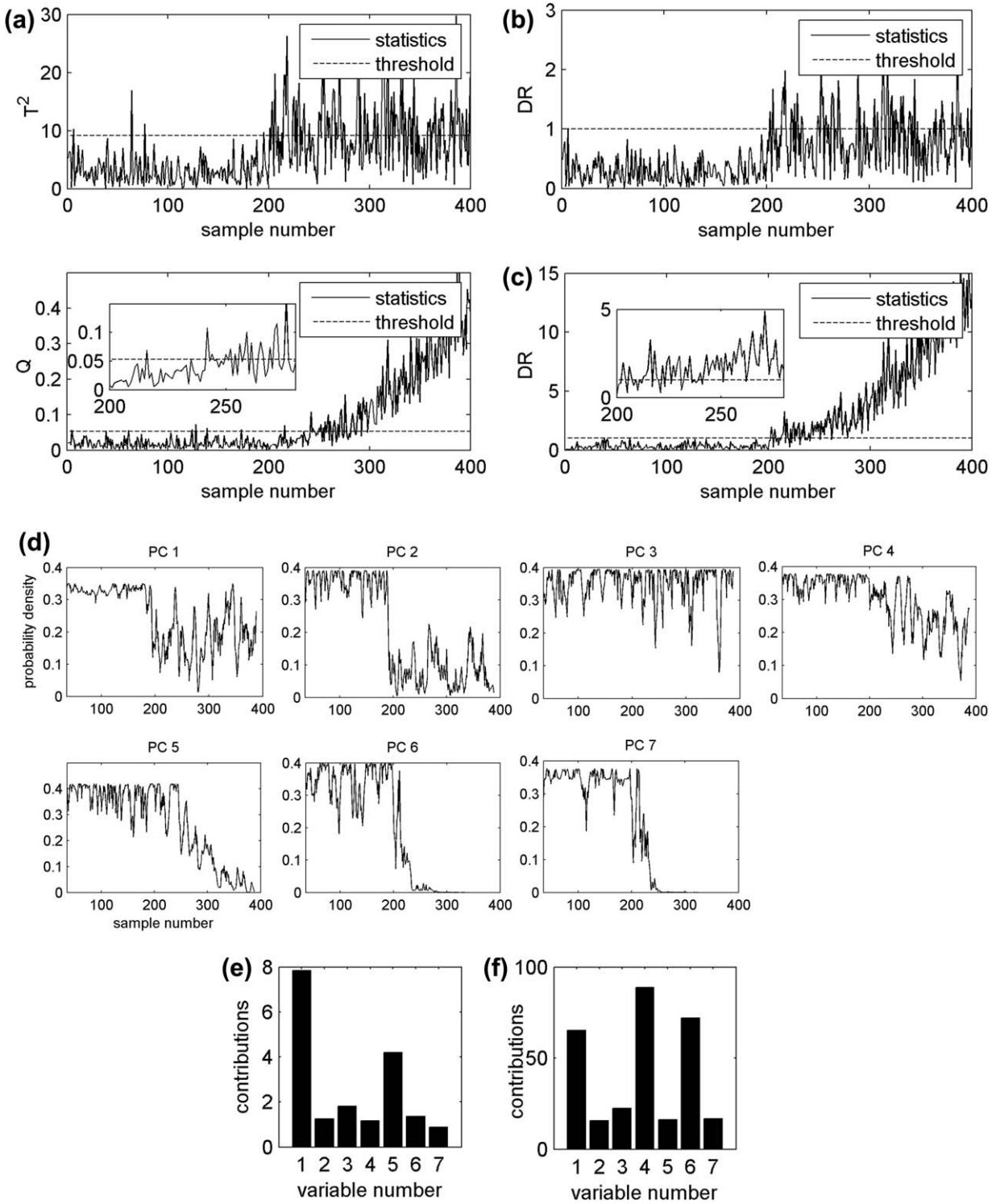

Figure 5. (a) PCA, (b) PCA-SVDD, and (c) JIR-PCA-SVDD monitoring results; (d) estimated density values along each PC; and (e) PCA contribution plots and (f) JIR-PCA contribution plots for Fault 2.

nondetection rate is significantly reduced. To analyze the monitoring behavior further, the estimated probability density value of each basic PC is plotted in Figure 4c. The PCs 3, 6, and 7 (arranged according to decreasing eigenvalues) show more significant changes and smaller density values, thereby indicating that the most fault information are reflected on these three PCs. This is in accordance with the results mentioned in motivational section. When reorganizing online, the three PCs would be selected to concentrate the fault information and construct the dominant subspace.

The concentration of the fault information is also beneficial for finding the root cause of the fault. The contribution plots in PCA and JIR-PCA are presented in Figures 4d, e. Figure 4d shows the variables' contributions, which are plotted at the 250th point. Variables 3–7 show larger contributions, in which the variable 3 has the most significant

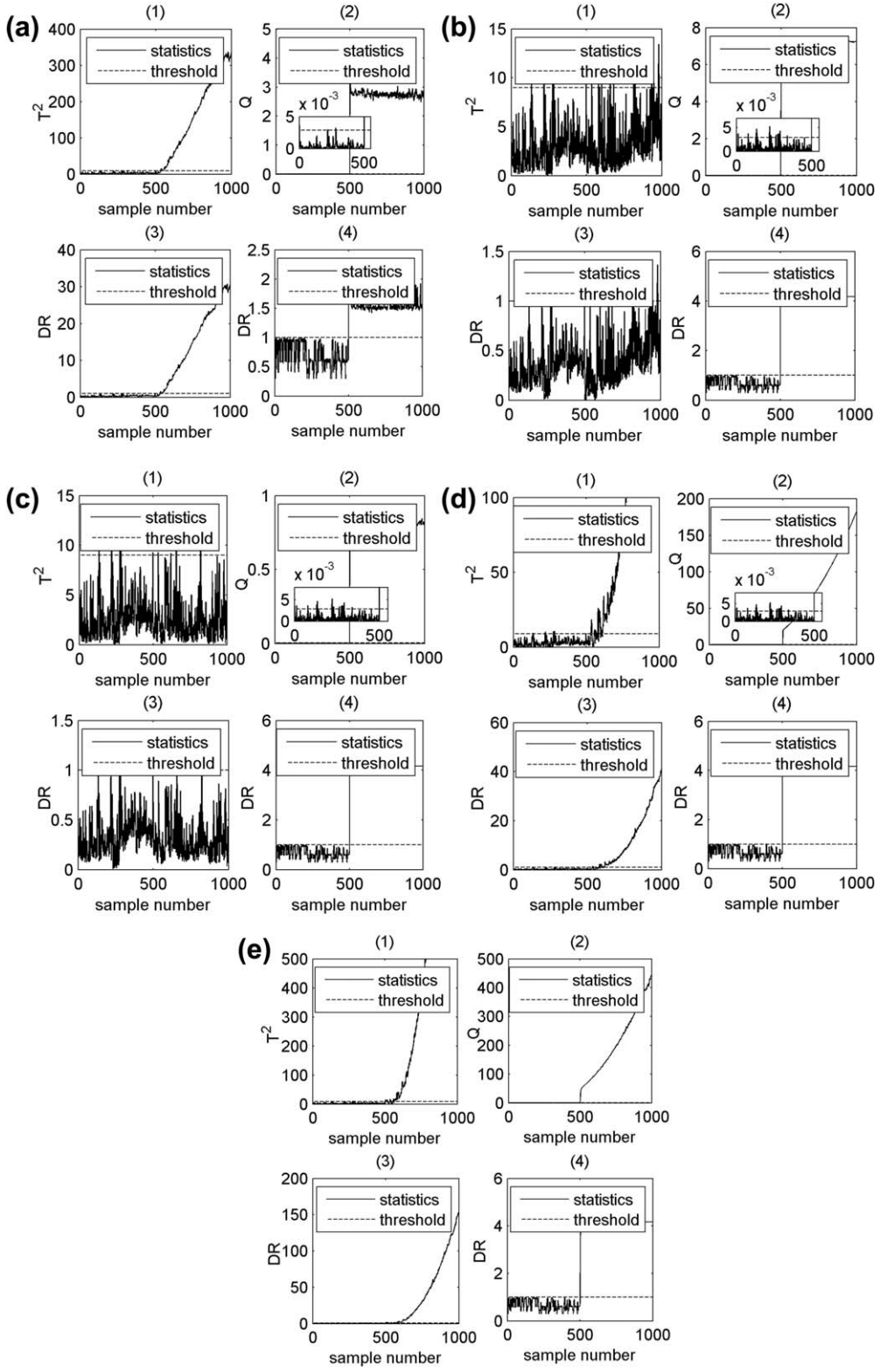

Figure 6. Monitoring performance for faults in CSTR using (1) PCA T^2, (2) PCA Q, (3) PCA-SVDD, and (4) JIR-PCA-SVDD for (a) Fault 1, (b) Fault 2, (c) Fault 3, (d) Fault 4, and (e) Fault 5.

contribution to the fault. This result is in accordance with the real condition, because the step change in variable 3 affects the variables 4–7. However, the results of the PCA and JIR-PCA are different. JIR-PCA shows that variables 4 and 6 are also significantly changed, which is much closer to the real condition, thereby providing more significant guidance in finding the cause. In the PCA method, the changes in variables 4 and 6 are not as significant because the information is dispersed. In addition, given that the informative PCs are concentrated and the fault information is enhanced, the influence of process noise is also reduced. Therefore, the identification result of JIR-PCA is relatively more reliable.

The Fault 2 in the numerical process is also examined. The monitoring results of PCA, PCA-SVDD, and JIR-PCA-

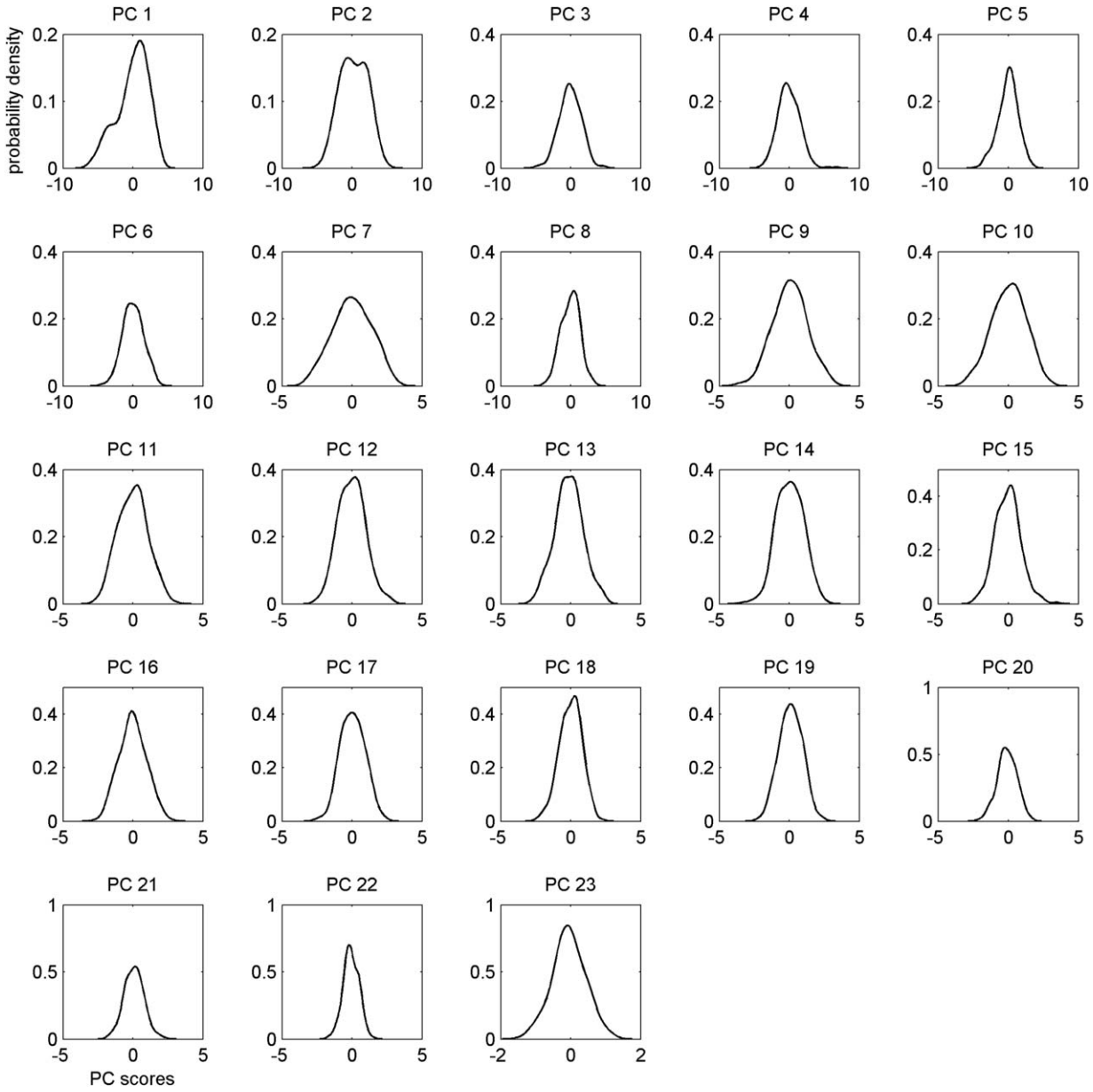

Figure 7. Estimated density along each PC of normal data of the TE process.

SVDD are presented in Figure 5. Figure 5a shows the monitoring results of PCA, including the T^2 and Q statistics; Figure 5b shows the monitoring results of PCA-SVDD; and Figure 5c shows the monitoring results of the proposed JIR-PCA-SVDD. From these figures, the JIR-PCA-SVDD shows the highest performance. The probability densities along each PC are presented in Figure 5d. At the beginning of Fault 2 (200–250 points), the step change is the major influence of the process. Most of the fault information is reflected on the dominant subspace (PCs 1 and 2). As the ramp change increases, the ramp change becomes the major influence, which is mainly reflected on the residual subspace (PCs 5, 6, and 7). In the JIR-PCA-SVDD, the most useful information is always reflected on the reorganized dominant subspace. Given that the most varied information is concentrated, the JIR-PCA-SVDD consistently shows better performance than the conventional PCA methods. To test the fault diagnosis ability, the contribution plots for Fault 2 in PCA and JIR-PCA are presented in Figures 5e, f. From the figures, the corresponding variables are identified successfully. Similar to the monitoring of Fault 1, JIR-PCA provides more significant results of the changes in variables.

CSTR process

The simulation model parameters and conditions used in the study by Yoon and MacGregor[56] are adopted in the current case study. The diagram of the process is presented in Figure A1. Additional details on the process and the parameters are found in the references.[56–58] The process is monitored by measuring the cooling water temperature , inlet temperature T_0, inlet concentrations C_{AA} and C_{AS}, solvent flow F_S, cooling water flow F_C, outlet concentration C_A and temperature, and reactant flow F_A. These nine variables form the measurement vector

$$\boldsymbol{x}=[T_C\ T_0\ C_{AA}\ C_{AS}\ F_S\ F_C\ C_A\ T_S\ F_A]^T \tag{25}$$

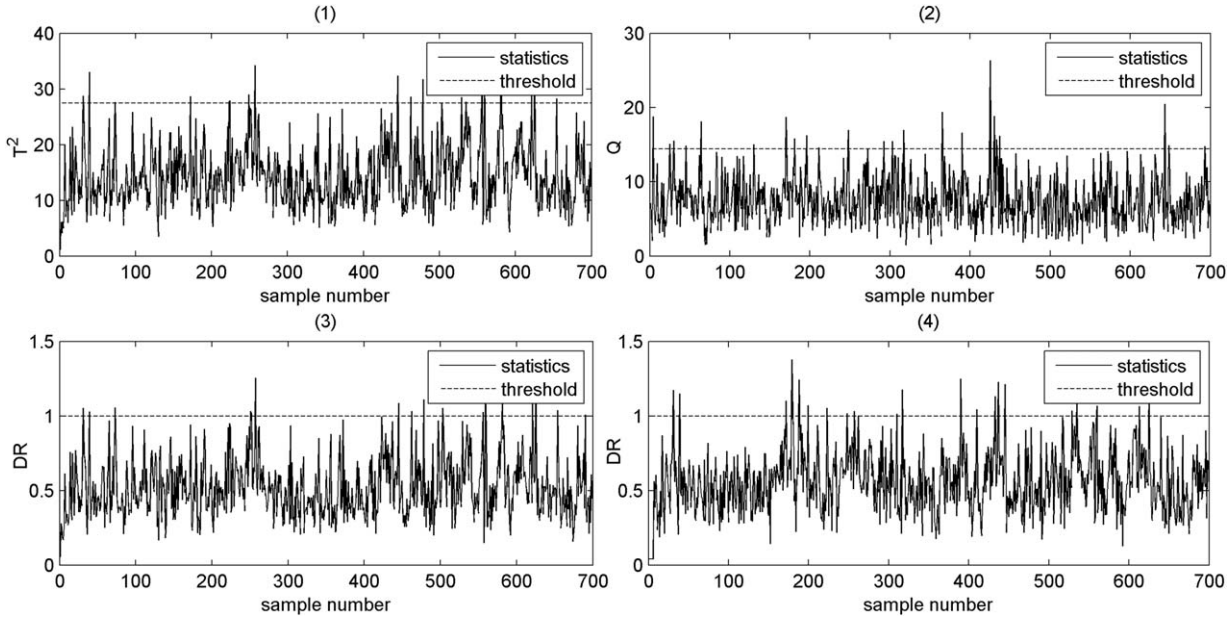

Figure 8. Monitoring results of (1) PCA T^2, (2) PCA Q, (3) PCA-SVDD, and (4) JIR-PCA-SVDD for the normal dataset in the TE process.

The variables are sampled every minute, and 500 samples under normal conditions are used as the training set. In building a conventional PCA model, the first four PCs, which occupy >85% CPV, are used, whereas all of the nine PCs are selected as basic PCs when building the JIR-PCA model. Five different faults are introduced into the system, and 1000 observations are simulated for each of the five scenarios. Fault 1 is a bias with magnitude of 0.2 (K) (the bias magnitude is 1 (K) in the reference, and the magnitude is reduced by 80% for acid testing)[57,58] in the sensor of the output temperature T_S. Faults 2 and 3 are biases in the sensors of the inlet temperature T_0 and inlet reactant concentration C_{AA}, respectively. The bias magnitude for T_0 is 0.3 (K) (reduced by 80% compared with that in the reference) and that for C_{AA} is 1.0 (kmol/m^3).[57,58] Fault 4 is a drift in the sensor of C_{AA}, and its magnitude is $dC_{AA}/dt=0.04$ $(\mathrm{kmol}/(\mathrm{m}^3\mathrm{min}))$($t$ represents the time here), which is reduced by 80% compared with that in the reference. Fault 5 is a slow drift in the reaction kinetics.[57,58] In this case, the reaction rate coefficient changes with time as $k_0(t+1)= 0.996 \times k_0(t)$. The faults are introduced at the 501st measurement.

When monitoring the Fault 1 (Figure 6a), the PCA, PCA-SVDD, and JIR-PCA-SVDD perform well. The fault is detected timely, and the number of nondetections is nearly zero, which is due to significant change in the process data caused by Fault 1, which is easily detected. In monitoring Fault 2, PCA T^2 and PCA-SVDD failed to detect the fault. Both the PCA Q and JIR-PCA-SVDD detect the fault timely and they have lowest nondetections. However, the false alarm rate in the JIR-PCA-SVDD is lower than that in the PCA Q. Therefore, JIR-PCA-SVDD performs the best in monitoring the Fault 2. The monitoring results of Faults 3, 4, 5 are similar to those of the Fault 2. Therefore, the JIR-PCA-SVDD performs the best with the lowest false alarm and nondetection rates.

TE process

The TE process, which was developed by Downs and Vogel,[59] is a benchmark case in process engineering. This process consists of five major unit operations, namely, a reactor, a product condenser, a vapor-liquid separator, a recycle compressor, and a product stripper. Two products are produced by two simultaneous gas–liquid exothermic reactions, and a byproduct is generated by two additional exothermic reactions.[3,59] The process has 12 manipulated variables, 22 continuous process measurements, and 19 compositions, as listed in Tables A1–A3. All of the process measurements are contaminated by Gaussian noise. The base control scheme for the TE process is shown in Figure A2, and the simulation code for the open loop can be downloaded from http://brahms.scs.uiuc.edu.[3,59–61] The second plant-wide control structure described in Lyman's study[3,60,61] is implemented in the present study to simulate the realistic conditions (closed loop).

In this article, 33 variables, including 22 continuous variables and 11 manipulated variables (agitation speed is not included because it is not manipulated), are selected for monitoring. To build the monitoring models, a normal process dataset (500 samples) is collected under base operation. A set of 21 programmed faults, which are listed in Table A4, are simulated and the corresponding process data are collected for testing. In building the JIR-PCA-SVDD model, the first 23 PCs (with CPV >99%) are selected as basic PCs, and the number of PCs of the reorganized JIR-PCA model are determined as 13 (with CPV >80%). The distributions of all of the basic PCs are estimated using KDE. The probability densities of the PCs under normal condition are presented in Figure 7.

Five typical testing sets, namely, Faults 0, 4, 11, 15, and 20, are used, and the monitoring results are presented. Fault 0 represents the normal operating condition in the TE process, which is usually used for determining the false alarm rates of monitoring schemes.[3,61] The monitoring performances of PCA, PCA-SVDD, and JIR-PCA-SVDD for Fault 0 are presented in Figure 8. The monitoring performance of the normal process is not degraded by introducing the JIR and SVDD. The false alarm rates of PCA T^2, PCA Q, PCA-SVDD, and JIR-PCA-SVDD are 0.0286, 0.0343, 0.0243, and 0.0329, respectively, which are negligible in engineering practice.

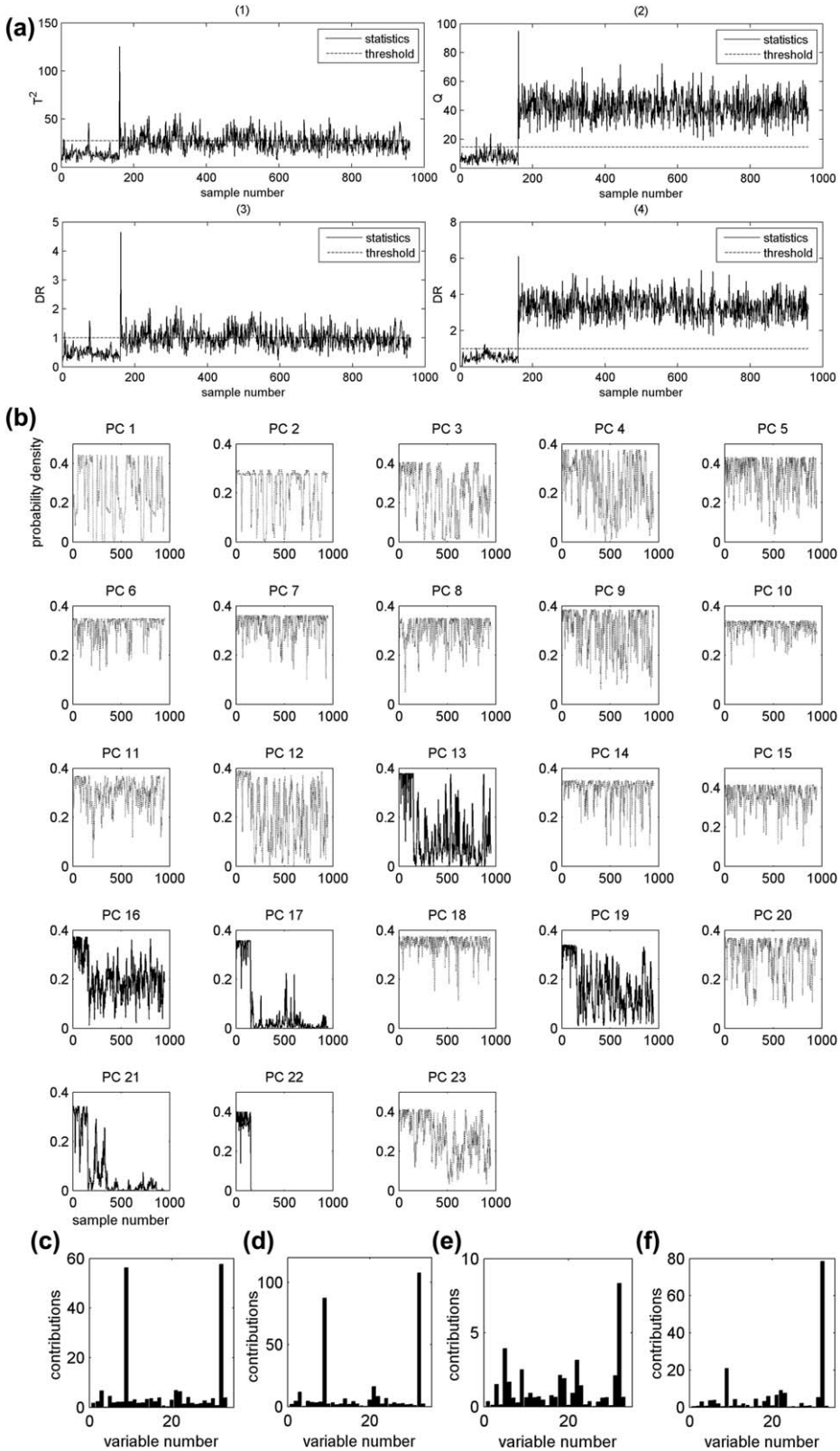

Figure 9. (a) Monitoring results of (1) PCA T^2, (2) PCA Q, (3) PCA-SVDD, and (4) JIR-PCA-SVDD; (b) density values along each PC; (c) PCA, (d) JIR-PCA contribution plots at the 161st point; (e) PCA, (f) JIR-PCA contribution plots at the 400th point for Fault 4.

Fault 4 involves a sudden temperature increase in the reactor, which is compensated by the control loops.[3,61] The monitoring results of Fault 4 using PCA, PCA-SVDD, and JIR-PCA-SVDD are presented in Figure 9a. The PCA Q and JIR-SVDD show good monitoring performance, whereas the results of PCA T^2 and PCA-SVDD are poor. To analyze the monitoring behavior further, the estimated density values along the PCs are presented in Figure 9b. The PCs 13, 16,

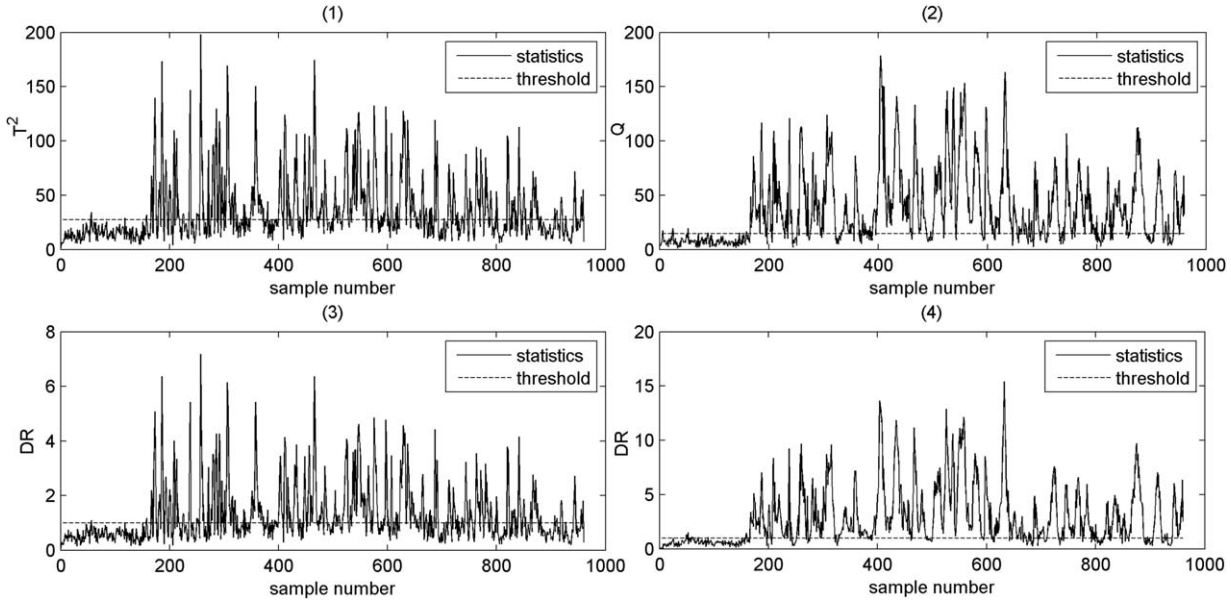

Figure 10. Monitoring results of (1) PCA T^2, (2) PCA Q, (3) PCA-SVDD, and (4) JIR-PCA-SVDD for Fault 11.

17, 19, 21, and 22, which are stressed by the full line, show larger variation with the occurrence of the fault. Most of the information are not captured by the first several PCs but are dispersed in the two subspaces, thereby leading to the poor detecting performance of PCA T^2 and PCA-SVDD. Given that the fault information is concentrated in JIR-PCA, the JIR-PCA-SVDD gives significantly better performance. The results variable identification are presented in Figures 9c–f, in which (c) and (d) are the results at the 161th point (start of the fault), whereas (e) and (f) are the results at the 400th point. At the 161th point, the two responsible variables [variable 9 (reactor temperature) and variable 32 (reactor cooling water flow)] are identified successfully by both PCA (c) and JIR-PCA (d). At the 400th point, the identification results are insignificant. The JIR-PCA (f) shows more clear indication of the two variables, whereas the contributions of variable 9 in PCA (e) are affected by process noise. These results are relatively insignificant compared with those in JIR-PCA.

When Fault 11 occurs, a random variation is introduced in the reactor cooling water inlet temperature; large oscillations are involved in the reactor cooling water flow rate. As a result, the reactor temperature fluctuates.[3,61] The monitoring results of Fault 11 using PCA, PCA-SVDD, and JIR-PCA-SVDD are presented in Figure 10. Some points under the confidence limit result in high nondetection rate when PCA T^2 and PCA-SVDD are used. The PCA Q captures more fault information and provides better monitoring results.

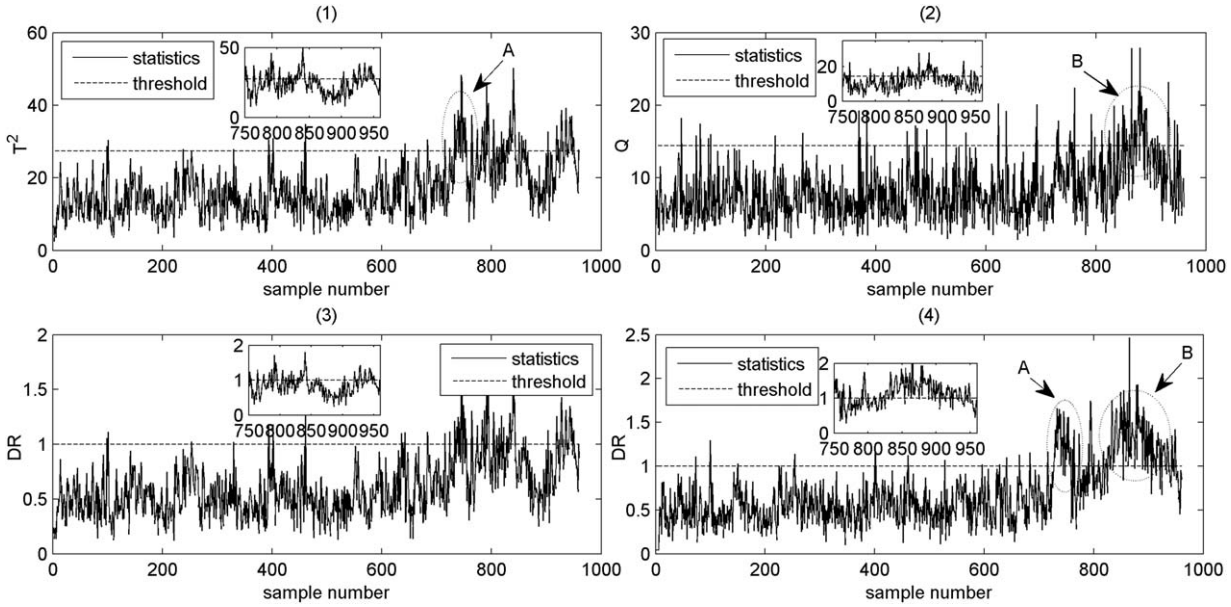

Figure 11. Monitoring results of (1) PCA T^2, (2) PCA Q, (3) PCA-SVDD, and (4) JIR-PCA-SVDD for Fault 15.

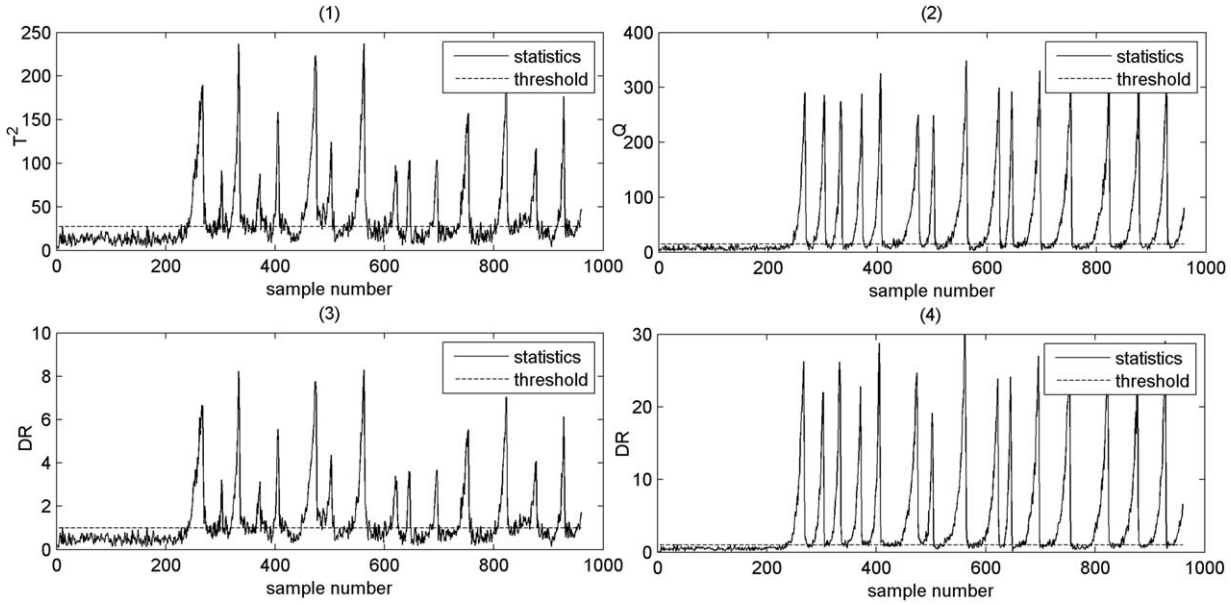

Figure 12. Monitoring results of (1) PCA T^2, (2) PCA Q, (3) PCA-SVDD, and (4) JIR-PCA-SVDD for Fault 20.

However, when JIR-PCA-SVDD is used, the number of nondetections is reduced further, thereby improving the monitoring performance.

Fault 15 is the sticking of condenser cooling water valve. Fault 15 is usually not used for monitoring performance comparison because no significant change in process data is found and the detection of the fault is difficult.[3,61] The monitoring results of the fault using PCA, PCA-SVDD, and JIR-PCA-SVDD are presented in Figure 11. The fault can be detected after the 720th point. Among the monitoring results of the three methods, the JIR-PCA-SVDD shows the best performance. In about 850th–930th point, the fault information is concentrated (both parts A and B in the PCA are concentrated in JIR-PCA-SVDD) in the JIR-PCA-SVDD model. In addition, the nondetection rates are significantly reduced.

Fault 20 is an unknown fault in the TE process[3,61]; the monitoring results are presented in Figure 12. In the figure, the results are similar to those in Fault 11 monitoring. JIR-PCA-SVDD is the most efficient among the three methods. The nondetection rates for each fault in the TE process using PCA,[3,41] WPCA,[24] DPCA,[3,16] PCA-SVDD, and JIR-PCA-SVDD are summarized in Table 1. Faults 3 and 9 are not used for comparison because no significant changes in the mean or the variance are found compared with the normal condition. Table 1 shows that the JIR-PCA-SVDD provides the best monitoring results in most of the cases, in which the nondetection rates of the faults are significantly reduced. JIR-PCA-SVDD shows the most efficient performance based on the comparison of the monitoring results.

Conclusions

In this study, a novel chemical process monitoring method, which integrates the JIR, PCA, and-SVDD, is

Table 1. Nondetection Rates of PCA, WPCA, DPCA, PCA-SVDD, and JIR-PCA-SVDD

Fault No.	PCA T^2	(W)PCA Q	WPCA T^2	DPCA T^2	DPCA Q	PCA-SVDD	JIR-PCA-SVDD
1	0.008	0	0.006	0.006	0.005	0.008	0.001
2	0.016	0.009	0.016	0.019	0.015	0.016	0.010
3	0.910	0.934	0.968	0.991	0.990	0.915	0.873
4	0.605	0	0.084	0.939	0	0.569	0
5	0.705	0.711	0.719	0.758	0.748	0.719	0.678
6	0.005	0	0.008	0.013	0	0.006	0.005
7	0	0	0	0.159	0	0	0
8	0.028	0.048	0.030	0.028	0.025	0.026	0.014
9	0.934	0.939	0.945	0.995	0.994	0.938	0.883
10	0.508	0.524	0.490	0.580	0.665	0.521	0.441
11	0.480	0.218	0.349	0.801	0.193	0.493	0.176
12	0.013	0.033	0.013	0.010	0.024	0.014	0.009
13	0.055	0.048	0.045	0.049	0.049	0.056	0.043
14	0.005	0	0	0.061	0	0.004	0
15	0.895	0.913	0.873	0.964	0.976	0.901	0.805
16	0.654	0.535	0.615	0.783	0.708	0.673	0.558
17	0.178	0.040	0.155	0.240	0.053	0.199	0.028
18	0.100	0.091	0.101	0.111	0.100	0.099	0.088
19	0.890	0.574	0.859	0.993	0.735	0.908	0.769
20	0.529	0.408	0.478	0.644	0.490	0.548	0.311
21	0.608	0.430	0.624	0.644	0.558	0.574	0.390

proposed to improve the monitoring performance of the PCA process. Given that unreasonable selection of PCs may result in useful fault information dispersion and submersion, the JIR-PCA algorithm is adopted. JIR-PCA evaluates the importance of each PC online using probability density estimation, and then the most useful PCs are concentrated into the reorganized dominant subspace. Given that the PCs in the reorganized dominant subspace no longer follow multivariate normal distribution, the SVDD is used for monitoring the subspace. Meanwhile, JIR-PCA concentrates the most useful information into one subspace, thereby significantly improving the effect of the contribution plots method on root-cause identification. The proposed fault detection and diagnosis methods are evaluated using numerical, CSTR process, and TE processes. The results in the present case study indicate that the SPCA method provides superior performance with regard to fault detection and diagnosis compared with the classical PCA-based methods.

This research is an extension and discussion of our previous work on WPCA and SPCA. Further studies on this subject are recommended. Future work could focus on extending JIR-PCA-SVDD to nonlinear process monitoring. Meanwhile, based on the PCA projection, only the mean-variances information, with high-order statistical information loss, is considered. Consideration of high-order statistical information with JIR technology is also recommended.

Appendix

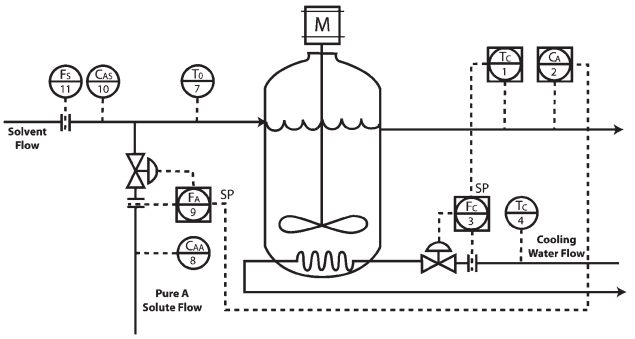

Figure A1. Diagram of the CSTR process.[56,57]

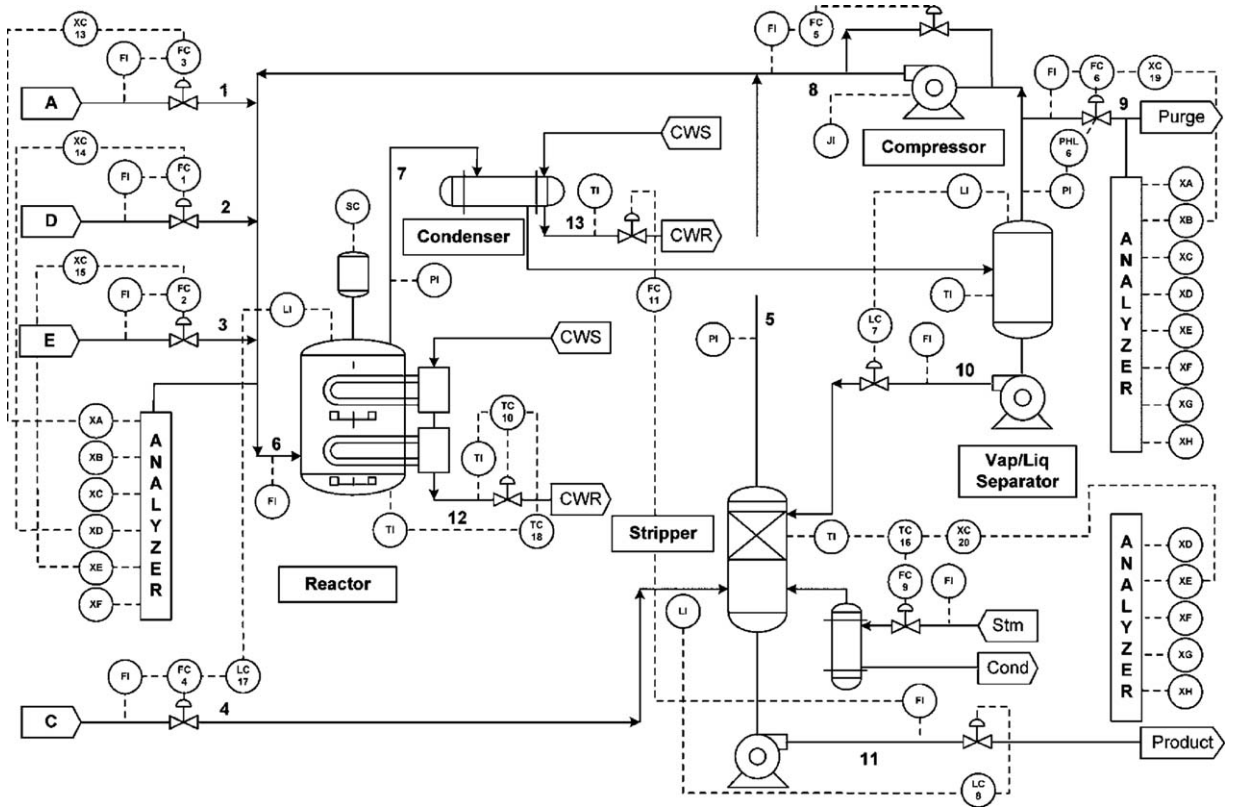

Figure A2. Control scheme for the TE process.[3,47]

Table A1. The Control Variable of TE Process[3,60]

Variables No.	State	Variables No.	State
XMV(1)	D feed flow (Stream 2)	XMV(7)	Separator pot liquid flow (Stream 10)
XMV(2)	E feed flow (Stream 3)	XMV(8)	Stripper liquid product flow (Stream 11)
XMV(3)	A feed flow (Stream 1)	XMV(9)	Stripper steam valve
XMV(4)	A and C feed flow (Stream 4)	XMV(10)	Reactor cooling water flow
XMV(5)	Compressor recycle value	XMV(11)	Condenser cooling water flow
XMV(6)	Purge valve (Stream 9)	XMV(12)	Stirring rate

Table A2. The Process Variable of TE Process[3,60]

Variables No.	Process Measurements	Unit
XMEAS(1)	A feed (Stream 1)	km^3/h
XMEAS(2)	D feed (Stream 2)	kg/h
XMEAS(3)	E feed (Stream 3)	kg/h
XMEAS(4)	A and C feed (Stream 4)	km^3/h
XMEAS(5)	Recycle flow (Stream 8)	km^3/h
XMEAS(6)	Reactor feed rate (Stream 6)	km^3/h
XMEAS(7)	Reactor pressure	kPa
XMEAS(8)	Reactor level	%
XMEAS(9)	Reactor temperature	°C
XMEAS(10)	Purge rate (Stream 9)	km^3/h
XMEAS(11)	Product separator temperature	°C
XMEAS(12)	Product separator level	%
XMEAS(13)	Product separator pressure	kPa
XMEAS(14)	Product separator underflow	m^3/h
XMEAS(15)	Stripper level	%
XMEAS(16)	Stripper pressure	kPa
XMEAS(17)	Stripper underflow (Stream 11)	m^3/h
XMEAS(18)	Stripper temperature	°C
XMEAS(19)	Stripper steam flow	kg/h
XMEAS(20)	Compress work	kW
XMEAS(21)	Reactor cooling water outlet temperature	°C
XMEAS(22)	Separator cooling water outlet temperature	°C

Table A3. The Process Variable of TE Process-2[3,60]

Variables	State	Stream No.	Sample Time/min
XMEAS(23)	Composition A	6	6
XMEAS(24)	Composition B	6	6
XMEAS(25)	Composition C	6	6
XMEAS(26)	Composition D	6	6
XMEAS(27)	Composition E	6	6
XMEAS(28)	Composition F	6	6
XMEAS(29)	Composition A	9	6
XMEAS(30)	Composition B	9	6
XMEAS(31)	Composition C	9	6
XMEAS(32)	Composition D	9	6
XMEAS(33)	Composition E	9	6
XMEAS(34)	Composition F	9	6
XMEAS(35)	Composition G	9	6
XMEAS(36)	Composition H	9	6
XMEAS(37)	Composition D	11	15
XMEAS(38)	Composition E	11	15
XMEAS(39)	Composition F	11	15
XMEAS(40)	Composition G	11	15
XMEAS(41)	Composition H	11	15

Acknowledgments

The authors gratefully acknowledge the support from the following foundations: 973 project of China (2013CB733 600), National Natural Science Foundation of China (21176073), Program for New Century Excellent Talents in University (NCET-09-0346), and the Fundamental Research Funds for the Central Universities. The authors also appreciate the valuable comments and suggestions of the anonymous reviewers.

Literature Cited

1. Jackson JE. Quality control methods for several related variables. *Technometrics*. 1959;1:359–377.
2. Raich A, Cinar A. Statistical process monitoring and disturbance diagnosis in multivariable continuous processes. *AIChE J*. 1996;42: 995–1009.
3. Chiang LH, Russell E, Braatz RD. *Fault Detection and Diagnosis in Industrial Systems*. London: Springer Verlag, 2001.
4. Negiz A, Çlinar A. Statistical monitoring of multivariable dynamic processes with state-space models. *AIChE J*. 1997;43:2002–2020.
5. Qin SJ. Statistical process monitoring: basics and beyond. *J Chemom*. 2003;17:480–502.
6. Venkatasubramanian V, Rengaswamy R, Kavuri SN, Yin K. A review of process fault detection and diagnosis: part III: process history based methods. *Comput Chem Eng*. 2003;27:327–346.
7. Qin SJ, Yu J. Recent developments in multivariable controller performance monitoring. *J Process Control*. 2007;17:221–227.
8. Kano M, Nakagawa Y. Data-based process monitoring, process control, and quality improvement: recent developments and applications in steel industry. *Comput Chem Eng*. 2008;32:12–24.
9. Qin SJ. Survey on data-driven industrial process monitoring and diagnosis. *Annu Rev Control*. 2012;36:220–234.
10. Ge Z, Song Z, Gao F. Review of recent research on data-based process monitoring. *Ind Eng Chem Res*. 2013;52:3534–3562.
11. Togkalidou T, Braatz RD, Johnson BK, Davidson O, Andrews A. Experimental design and inferential modeling in pharmaceutical crystallization. *AIChE J*. 2001;47:160–168.
12. Jackson JE. *A User's Guide to Principal Components*. New York: Wiley, 1991.
13. Lee JM, Yoo C, Choi SW, Vanrolleghem PA, Lee IB. Nonlinear process monitoring using kernel principal component analysis. *Chem Eng Sci*. 2004;59:223–234.
14. Schölkopf B, Smola A, Müller KR. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput*. 1998;10:1299–1319.
15. Choi SW, Lee IB. Nonlinear dynamic process monitoring based on dynamic kernel PCA. *Chem Eng Sci*. 2004;59:5897–5908.

Table A4. Process Faults for the TE Process[3,60]

Faults No.	Disturbance State	Type
IDV(1)	A/C feed ratio, B composition constant (Stream 4)	Step
IDV (2)	B composition, A/C ratio constant (Stream 4)	Step
IDV (3)	D feed temperature (Stream 2)	Step
IDV (4)	Reactor cooling water inlet temperature	Step
IDV (5)	Condenser cooling water inlet temperature	Step
IDV (6)	A feed loss (Stream 1)	Step
IDV (7)	C header pressure loss reduced availability (Stream 4)	Step
IDV (8)	A,B,C feed composition (Stream 4)	Random variation
IDV (9)	D feed temperature (Stream 2)	Random variation
IDV (10)	C feed temperature (Stream 4)	Random variation
IDV (11)	Reactor cooling water inlet temperature	Random variation
IDV (12)	Condenser cooling water inlet temperature	Random variation
IDV (13)	Reaction kinetics	Slow drift
IDV (14)	Reactor cooling water valve	Sticking
IDV (15)	Condenser cooling water valve	Sticking
IDV (16)	Unknown	Unknown
IDV (17)	Unknown	Unknown
IDV (18)	Unknown	Unknown
IDV (19)	Unknown	Unknown
IDV (20)	Unknown	Unknown
IDV (21)	The valve for Stream 4 was fixed at the steady state position	Step constant position

16. Ku W, Storer RH, Georgakis C. Disturbance detection and isolation by dynamic principal component analysis. *Chemom Intell Lab Syst*. 1995;30:179–196.
17. Lee JM, Yoo C, Lee IB. Fault detection of batch processes using multiway kernel principal component analysis. *Comput Chem Eng*. 2004;28:1837–1847.
18. Nomikos P, MacGregor JF. Monitoring batch processes using multiway principal component analysis. *AIChE J*. 1994;40:1361–1375.
19. Aradhye HB, Bakshi BR, Strauss RA, Davis JF. Multiscale SPC using wavelets: theoretical analysis and properties. *AIChE J*. 2003; 49:939–958.
20. Bakshi BR. Multiscale PCA with application to multivariate statistical process monitoring. *AIChE J*. 1998;44:1596–1610.
21. Chen J, Liu KC. On-line batch process monitoring using dynamic PCA and dynamic PLS models. *Chem Eng Sci*. 2002;57:63–75.
22. Dong D, McAvoy TJ. Nonlinear principal component analysis—based on principal curves and neural networks. *Comput Chem Eng*. 1996;20:65–78.
23. Feital T, Kruger U, Dutra J, Pinto JC, Lima EL. Modeling and performance monitoring of multivariate multimodal processes. *AIChE J*. 2012;59:1557–1569.
24. Jiang Q, Yan X. Chemical processes monitoring based on weighted principal component analysis and its application. *Chemom Intell Lab Syst*. 2012;119:11–20.
25. Jiang Q, Yan X, Zhao W. Fault detection and diagnosis in chemical processes using sensitive principal component analysis. *Ind Eng Chem Res*. 2013;52:1635–1644.
26. Lu N, Gao F, Wang F. Sub-PCA modeling and on-line monitoring strategy for batch processes. *AIChE J*. 2004;50:255–259.
27. Rännar S, MacGregor JF, Wold S. Adaptive batch monitoring using hierarchical PCA. *Chemom Intell Lab Syst*. 1998;41:73–81.
28. Jolliffe IT. A note on the use of principal components in regression. *Appl Stat*. 1982;31:300–303.
29. Hill RC, Fomby TB, Johnson SR. Component selection norms for principal components regression. *Commun Stat Theory Methods*. 1977;6:309–334.
30. Qin SJ, Dunia R. Determining the number of principal components for best reconstruction. *J Process Control*. 2000;10:245–250.
31. Valle S, Li W, Qin SJ. Selection of the number of principal components: the variance of the reconstruction error criterion with a comparison to other methods. *Ind Eng Chem Res*. 1999;38:4389–4401.
32. Wold S. Cross-validatory estimation of the number of components in factor and principal components models. *Technometrics*. 1978;20: 397–405.
33. Zwick WR, Velicer WF. Comparison of five rules for determining the number of components to retain. *Psychol Bull*. 1986;99:432–442.
34. Johnson RA, Wichern DW. *Applied Multivariate Statistical Analysis*. New Jersey: Prentice Hall, 1992.
35. Li Y, Tang XC. Improved performance of fault detection based on selection of the optimal number of principal components. *Acta Autom Sin*. 2009;35:1550–1557.
36. Xie YL, Kalivas JH. Evaluation of principal component selection methods to form a global prediction model by principal component regression. *Anal Chim Acta*. 1997;348:19–27.
37. Draper NH, Smith H. *Applied Regression Analysis*. New York: Wiley, 1981.
38. Phatak A, Reilly P, Penlidis A. An approach to interval estimation in partial least squares regression. *Anal Chim Acta*. 1993;277:495–501.
39. Kung EC, Sharif TA. Regression forecasting of the onset of the Indian summer monsoon with antecedent upper air conditions. *J Appl Meteor*. 1980;19:370–380.
40. Chen Q, Kruger U, Meronk M, Leung A. Synthesis of T2 and Q statistics for process monitoring. *Control Eng Pract*. 2004;12(6):745–755.
41. Kourti T, MacGregor JF. Multivariate SPC methods for process and product monitoring. *J Qual Technol*. 1996;28:409–428.
42. Wang H, Song Z, Li P. Fault detection behavior and performance analysis of principal component analysis based process monitoring methods. *Ind Eng Chem Res*. 2002;41:2455–2464.
43. Rashid M, Yu J. Nonlinear and non-Gaussian dynamic batch process monitoring using a new multiway kernel independent component analysis and multidimensional mutual information based dissimilarity method. *Ind Eng Chem Res*. 2012;51:10910–10920.
44. Rashid M, Yu J. A new dissimilarity method integrating multidimensional mutual information and independent component analysis for non-Gaussian dynamic process monitoring. *Chemom Intell Lab Syst*. 2012;115:44–58.
45. Ge Z, Song Z. Performance-driven ensemble learning ICA model for improved non-Gaussian process monitoring. *Chemom Intell Lab Syst*. 2013;123:1–8.
46. Chiang LH, Kotanchek ME, Kordon AK. Fault diagnosis based on Fisher discriminant analysis and support vector machines. *Comput Chem Eng*. 2004;28:1389–1401.
47. Chiang LH, Russell EL, Braatz RD. Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares, and principal component analysis. *Chemom Intell Lab Syst*. 2000;50:243–252.
48. Ge Z, Gao F, Song Z. Batch process monitoring based on support vector data description method. *J Process Control*. 2011;21:949–959.
49. Tax DM, Duin RP. Support vector data description. *Mach Learn*. 2004;54:45–66.
50. Ghate VN, Dudul SV. Optimal MLP neural network classifier for fault detection of three phase induction motor. *Expert Syst Appl*. 2010;37:3468–3481.
51. Rengaswamy R, Venkatasubramanian V. A fast training neural network and its updation for incipient fault detection and diagnosis. *Comput Chem Eng*. 2000;24:431–437.
52. Parzen E. On estimation of a probability density function and mode. *Ann Math Stat*. 1962;33:1065–1076.
53. Scott DW. Multivariate Density Estimation. New York: Wiley, 1992.
54. Webb AR, Copsey K. *Statistical Pattern Recognition*, 3rd ed. UK: Wiley, 2011.
55. Silverman BW. Density Estimation for Statistics and Data Analysis. London: Chapman & Hall, 1986.
56. Yoon S, MacGregor JF. Fault diagnosis with multivariate statistical models part I: using steady state fault signatures. *J Process Control*. 2001;11:387–400.
57. Choi SW, Yoo CK, Lee IB. Overall statistical monitoring of static and dynamic patterns. *Ind Eng Chem Res*. 2003;42:108–117.
58. Yue HH, Qin SJ. Reconstruction-based fault identification using a combined index. *Ind Eng Chem Res*. 2001;40:4403–4414.
59. Downs JJ, Vogel EF. A plant-wide industrial process control problem. *Comput Chem Eng*. 1993;17:245–255.
60. Lyman PR, Georgakis C. Plant-wide control of the Tennessee Eastman problem. *Comput Chem Eng*. 1995;19:321–331.
61. McAvoy T, Ye N. Base control for the Tennessee Eastman problem. *Comput Chem Eng*. 1994;18:383–413.

Manuscript received July 13, 2013, and revision received Nov. 12, 2013.