Research Article

Multimode Process Monitoring Based on Sparse Principal Component Selection and Bayesian Inference-Based Probability

Xiaodong Jiang, Haitao Zhao, and Bo Jin

1Key Laboratory of Advanced Control and Optimization for Chemical Processes of Ministry of Education, East China University of Science and Technology, Shanghai 200237, China
2Software Engineering Institute, East China Normal University, Shanghai 200062, China

Correspondence should be addressed to Haitao Zhao; ht.zhao.ecust@gmail.com

Received 6 May 2015; Revised 27 July 2015; Accepted 28 July 2015

Copyright © 2015 Xiaodong Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

According to the demand for diversified products, modern industrial processes typically have multiple operating modes. At the same time, variables within the same mode often follow a mixture of Gaussian distributions. In this paper, a novel algorithm based on sparse principal component selection (SPCS) and Bayesian inference-based probability (BIP) is proposed for multimode process monitoring. SPCS can be formulated as a just-in-time regression between all PCs and each sample. SPCS selects PCs according to the nonzero regression coefficients which indicate the compact expression of the sample. This expression is necessarily discriminative: amongst all subset of PCs, SPCS selects the PCs which most compactly express the sample and rejects all other possible but less compact expressions. BIP is utilized to compute the posterior probabilities of each monitored sample belonging to the multiple components and derive an integrated global probabilistic index for fault detection of multimode processes. Finally, to verify its superiority, the SPCS-BIP algorithm is applied to the Tennessee Eastman (TE) benchmark process and a continuous stirred-tank reactor (CSTR) process.

1. Introduction

Over the past two decades, with the development of complex chemical processes and the growing demand of plant safety and stable product quality, timely process monitoring is gaining importance. Because large amounts of data can be gathered by the use of distributed control systems (DCSs), multivariate statistical process monitoring (MSPM) algorithms have received great attention. Among these algorithms, principal component analysis (PCA) and partial least squares (PLS) are the most widely used algorithms [1–8]. Both algorithms project high-dimensional data onto lower dimensional subspaces. Process normal and abnormal conditions can be isolated by the use of Hotelling’s $T^2$ or squared predicted error (SPE) [9–13]. Other complementary MSPM algorithms, including independent components analysis (ICA), Fisher discriminant analysis (FDA), and canonical variate analysis (CVA), are used to overcome some limitations in PCA/PLS-based monitoring schemes [14–17]. However, most of MSPM algorithms rely on the assumption that the system is in a single operating region and that the data follow a Gaussian distribution. In chemical processes, operating condition shifts are often encountered due to the changes of various factors such as feedstock, product specification, set points, and manufacturing strategy. When a process is running under substantially different operating conditions, only a small number of variables actually follow Gaussian distribution [18]. As a result, the multimodality of data distribution might lead to unseemliness for the monitoring of conventional MSPM algorithms. To address these problems, it is necessary to develop new algorithms.

In literature, multiple models can be built to fit each individual operating mode, but these are two essential issues that need to be addressed. One is how to divide the training data into multiple subsets correctly, corresponding to different operating modes. In order to solve this issue, many clustering algorithms are applied. In terms of the traditional approaches, Ge and Song [19] used fuzzy C-means clustering algorithm to separate the training data set according to the unique characteristics of each mode. He et al. [20] applied
the k-nearest-neighbor method. Srinivasan et al. [21, 22] identified the different operating modes by evaluating the Euclidean distances between samples in a constructed data window and then applied dynamic PCA-based similarity measures to cluster the samples. Liu and Chen [23] developed a method using Bayesian classification for selecting multiple regions from a training data set. Zhao et al. [24] presented a multiple principal component analysis (MPCA) algorithm that selects one suitable model to monitor multimode processes. The other issue is how to determine the final results. A proper measurement should be employed to determine which model is the most suitable one for monitoring at the current moment. Ng and Srinivasan [22] exploited the most suitable PCA model through a minimized distance reflecting both the $T^2$ and SPE values. Zhao et al. [24] close the local PCA model with the minimum SPE value. Natarajan and Srinivasan [21] used the distance between the sample and the center of local models as a criterion. Yu and Qin [25] performed Bayesian inference on the postprobabilities calculated by the Gaussian mixture model (GMM) or the nonlinear kernel GMM. Meanwhile, Ge and coworkers [26, 27] took advantage of Bayesian inference to softly combine the monitoring results computed by local models built by means of probabilistic PCA (PPCA), factor analysis (FA), or subspace algorithms.

To date, the problem of how to correctly divide the training data into multiple subsets can successfully be solved by many algorithms mentioned in the previous paragraph. However, there are still some issues that need to be resolved; the most important one is how to select the key principal components (PCs) when using one suitable model for process monitoring. Many algorithms for selecting PCs have been proposed, such as cumulative percent variance (CPV) [28], variance of reconstruction error (VRE) [29], and cross validation (CV) [30]. Generally, most of the classical algorithms just take normal operational observations into account and select the first several PCs with larger variance. While PCs with larger variance of normal data cannot guarantee the capture of the largest variations in fault data online. Jolliffe [31] suggested that the last PCs may be as important as those with large variance. Togkalidou et al. [32] noted that the PCs with larger variance do not always contain much information for prediction. However, this issue is insufficiently discussed in PCA-based process monitoring, and the standard PC selection is still not established.

Fortunately, many researches have been aware of the inherent defects of classical PCA algorithm. A lot of workers tried to seek a subspace spanned by key PCs, which contains the most important information for process monitoring. Peng et al. [33] suggested a new feature selection algorithm, named minimal-redundancy-maximal-relevance criterion (mRMR). It is based on mutual information and selects the features with highest relevance to the target class. Jiang et al. [34] put forward the sensitive principal component analysis for fault detection and diagnosis in chemical processes. They pointed out that PCs selected by PCA algorithm are not always the key PCs for fault detection. Their task was to find the sensitive PCs which have relationship with fault information. Arbel et al. [35] proposed that the process variables that are preponderant in achieving specific objectives need to be selected.

In this paper, a process monitoring algorithm using multisubspace sparse principal component analysis with the BIP algorithm is put forward. First, variables are divided into different subblocks corresponding to different units or pieces of equipment to reduce the complexity of process analysis. By using BIP algorithm, multimode data in each subblock are divided into multiple subgroups. BIP can compute the posterior probabilities of each monitored sample belonging to the multiple components and derive an integrated global probabilistic index for fault detection of multimode processes. The PCs selected by PCA algorithm with larger variances do not always have relationship with fault information. Sparse principal component selection (SPCS) takes the information of both normal and abnormal observations into account. The algorithm is formulated as a just-in-time form that constructs an elastic net regression between all PCs and each sample. SPCS selects PCs corresponding to the nonzero regression coefficients which indicate the compact expression of the sample. This expression is necessarily discriminative: amongst all subset of PCs, SPCS selects the PCs which most compactly express the sample and rejects all other possible but less compact expression. Third, the key PCs are selected by SPCS in each subgroup to solve the problem of fault information loss. It needs to be stressed that the subspace spanned by the key PCs selected is the feature subspace. Finally, in order to verify the superiority of the SPCS-BIP algorithm, it is applied to the Tennessee Eastman (TE) benchmark problem and a continuous stirred-tank reactor (CSTRT) process.

2. Preliminaries

2.1. Principal Component Analysis. Principal component analysis is a multivariate statistical analysis which is widely used in chemical process monitoring, fault detection, and so forth [36–38]. Let $x \in \mathbb{R}^m$ represent an $m$-dimensional sample vector and $X \in \mathbb{R}^{N \times m}$ denote a data matrix with zero mean and unit variance, where $N$ is the number of samples and $m$ is the number of variables in the process. From the statistical viewpoint, the PCA algorithm could be obtained by singular value decomposition (SVD) [28, 34]:

$$X = TP^T + E = \hat{X} + E,$$

(1)

where $T \in \mathbb{R}^{N \times k}$ and $P \in \mathbb{R}^{m \times k}$ are the score matrix and the loading matrix, respectively. $k$ is the principal components retained number. The loading matrix $P$ can be obtained by eigenvalue decomposition on the covariance matrix $\text{cov}(x)$ as follows:

$$\text{cov}(x) = \frac{1}{N-1}X^TX = \Lambda P P^T,$$

(2)

where $\Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m)$ denotes the eigenvalue matrix and $P = [P_1 P_2 \ldots P_k]$ contains the loading matrices of component subspace and residual subspace, respectively.
2.2. Construction of Finite Gaussian Mixture Model Based on EM. For the process running at multiple operating conditions, owing to the mean shifts or covariance changes, the assumption of multivariate Gaussian distribution becomes invalid [21, 22]. In this situation, the local Gaussian distribution is still appropriate to characterize each subset of measurement data from the same operating conditions. Therefore, the finite Gaussian mixture model is prime suited to represent the data sources driven by different operating modes [13, 24, 25].

To construct a FGMM, given a set of training samples \( \mathbf{X} \in \mathbb{R}^{N \times m} \), the log-likelihood function can be expressed as

\[
\log L(\mathbf{X}, \Theta) = \sum_{j=1}^{N} \log \left( \sum_{l=1}^{L} \omega_{l} g \left( x_{j} \mid \theta_{l} \right) \right)
\]

and the parameter estimation problem is formulated as

\[
\hat{\Theta} = \arg \max_{\Theta} \left( \log L(\mathbf{X}, \Theta) \right),
\]

where \( \Theta = \{ \omega_{1}, \mu_{1}, \Sigma_{1}, \ldots, \omega_{l}, \mu_{l}, \Sigma_{l} \} \), \( \omega_{l} (1 \leq i \leq L) \) is the prior probabilities, and \( L \) is the number of Gaussian components included in FGMM. \( \mu_{l} \) is the mean vector and \( \Sigma_{l} \) is the covariance matrix.

There are a lot of learning algorithms, such as maximum likelihood estimation (MLE), EM, and the F-J algorithm, that have been put forward for mixture model estimation [39, 40]. As a more tractable numerical strategy, the EM algorithm has been well used in practice to estimate the maximum likelihood distribution parameter [39]. EM algorithm is implemented iteratively by means of repeating the expectation step (E-step) and maximization step (M-step) to calculate the posterior probabilities and then the corresponding distribution parameters until a convergence criterion of the log-likelihood function is satisfied. Given the training data \( \mathbf{X} \) and an initial estimate \( \Theta^{(0)} = \{ \{\omega_{0}^{(0)}, \mu_{1}^{(0)}, \Sigma_{1}^{(0)}\}, \ldots, \{\omega_{L}^{(0)}, \mu_{l}^{(0)}, \Sigma_{l}^{(0)}\} \} \), the iterative E-step and M-step are expressed as follows:

(i) E-step:

\[
p^{(s)}(C_{l} \mid x_{j}) = \frac{\omega_{l}^{(s)} g \left( x_{j} \mid \mu_{l}^{(s)}, \Sigma_{l}^{(s)} \right)}{\sum_{l=1}^{L} \omega_{l}^{(s)} g \left( x_{j} \mid \mu_{l}^{(s)}, \Sigma_{l}^{(s)} \right)},
\]

where \( p^{(s)}(C_{l} \mid x_{j}) \) denotes the posterior probability of the \( j \)th training sample within the \( l \)th Gaussian component at the \( s \)th iteration;

(ii) M-step:

\[
\hat{\mu}_{l}^{(s+1)} = \frac{\sum_{j=1}^{N} p^{(s)}(C_{l} \mid x_{j}) x_{j}}{\sum_{j=1}^{N} p^{(s)}(C_{l} \mid x_{j})},
\]

\[
\Sigma_{l}^{(s+1)} = \frac{\sum_{j=1}^{N} p^{(s)}(C_{l} \mid x_{j}) (x_{j} - \hat{\mu}_{l}^{(s+1)}) (x_{j} - \hat{\mu}_{l}^{(s+1)})^{T}}{\sum_{j=1}^{N} p^{(s)}(C_{l} \mid x_{j})},
\]

\[
\omega_{l}^{(s+1)} = \frac{\sum_{j=1}^{N} p^{(s)}(C_{l} \mid x_{j})}{N},
\]

where \( \mu_{l}^{(s+1)}, \Sigma_{l}^{(s+1)} \), and \( \omega_{l}^{(s+1)} \) are the mean, covariance, and prior probability of the \( l \)th Gaussian component at the \( (s+1) \)th iteration, respectively.

3. Fault Detection with Sparse Principal Component Selection and Bayesian Inference-Based Probability

In this section, the idea of SPCS-BIP algorithm for multimode process monitoring is demonstrated in detail. We first introduce the Bayesian inference-based probability which can derive the confidence boundary around the normal operating regions for process monitoring and fault detection. Then, the sparse principal component selection was introduced for selecting the key Pcs related with fault information. Finally, the steps of this algorithm were given.

3.1. Bayesian Inference-Based Probability. In the previous section, the FGMM has been constructed, and it is essential to further derive the confidence boundary around the normal operating regions for process monitoring and fault detection. Due to the multimodality of mixture distribution, it is really difficult to capture the analytical boundary of the density function \( p(x \mid \Theta) \) in a certain confidence level.

In the proposed monitoring approach, given an arbitrary monitored sample \( x_{t} \) belonging to each Gaussian component, Bayesian inference strategy is used to calculate the posterior probability as follows:

\[
P (x_{t} \in C_{l}) = \frac{P (C_{l} \mid x_{t}) P (x_{t} \mid C_{l})}{P (x_{t})} = \frac{P (C_{l}) \cdot P (x_{t} \mid C_{l})}{\sum_{l=1}^{L} P (C_{l}) \cdot P (x_{t} \mid C_{l})}
\]

which can also be formulated as

\[
P (x_{t} \in C_{l}) = \frac{\omega_{l} g \left( x_{t} \mid \mu_{l}, \Sigma_{l} \right)}{\sum_{l=1}^{L} \omega_{l} g \left( x_{t} \mid \mu_{l}, \Sigma_{l} \right)} (l = 1, 2, \ldots, L).
\]

Given that each component \( C_{l} \) follows a unimodal Gaussian distribution, the squared Mahalanobis distance of \( x_{t} \) from the center of \( C_{l} \) follows \( \chi^{2} \) distribution, provided that \( x_{t} \) belongs to \( C_{l} \),

\[
D \left( x_{t}, C_{l} \right) = (x_{t} - \mu_{l})^{T} \Sigma_{l}^{-1} (x_{t} - \mu_{l}) : \chi_{m}^{2}.
\]

Under the assumption that \( x_{t} \in C_{l} \) and \( \chi_{m}^{2} \) has \( m \) degree of freedom, \( D \left( x_{t}, C_{l} \right) \) denotes the squared Mahalanobis distance between \( x_{t} \) and the mean center of \( C_{l} \). Owing to colinearity, \( \Sigma_{l} \) is usually ill-conditioned, and the following regularized Mahalanobis distance is utilized instead to avoid too wide confidence regions:

\[
D_{r} \left( x_{t}, C_{l} \right) = \left( x_{t} - \mu_{l} \right)^{T} (\Sigma_{l} + \varepsilon I)^{-1} (x_{t} - \mu_{l}),
\]
where the function of $\epsilon$ is to remove the ill condition of covariance matrix $\Sigma_l$ by adding a positive number to all the diagonal entries.

For the monitored sample $x_l$, a local Mahalanobis distance-based probability index relative to each Gaussian component $C_i$ can be defined as

$$P_L^{(i)}(x_l) = \Pr \{ D((x, C_i) | x \in C_i) \leq D((x_l, C_i) | x_l \in C_i) \}, \quad (11)$$

or

$$P_L^{(i)}(x_l) = \Pr \{ D_r((x, C_i) | x \in C_i) \leq D_r((x_l, C_i) | x_l \in C_i) \}. \quad (12)$$

Given the appropriate degree of freedom, $P_L^{(i)}(x_l)$ can be computed by integrating the $\chi^2$ probability density function. Under a given confidence level, this index has the function of indicating whether the monitored sample is normal or abnormal provided that it belongs to the corresponding Gaussian component. A global BIP index is proposed to combine the local probability metrics across all the Gaussian clusters because the random characteristic of each monitored sample may come from multiple Gaussian components with the corresponding posterior probabilities. The formulation of BIP index for the monitored sample $x_l$ is given by

$$\text{BIP} = \sum_{i=1}^{L} P(C_i | x_l) P_L^{(i)}(x_l), \quad (13)$$

where the posterior probability $P(C_i | x_l)$ is used to incorporate the contribution of each local Gaussian component to the overall probabilistic index. As $0 \leq P_L^{(i)}(x_l) \leq 1$, we have

$$0 \leq \text{BIP} \leq \sum_{i=1}^{L} P(C_i | x_l) = 1. \quad (14)$$

Under the preset confidence level $(1-\alpha) 100\%$, the process is determined within normal operation if

$$\text{BIP} \leq 1 - \alpha. \quad (15)$$

Otherwise, the process operation is treated out of control.

3.2. Sparse Principal Component Selection. Sparse representation has proven to be an extremely powerful tool for acquiring, representing, and compressing high-dimensional data [41–43]. This success is mainly because of the fact that the important reconstruction information of data such as process data and time series data has naturally sparse representations with respect to fixed bases, or concatenations of such bases. Qiao et al. [44] proposed that the graphs constructed by the $L_1$-norm have the advantage of greater robustness to data noise, automatic sparsity, and adaptive neighborhood for individual datum. What is more another important advantage is that sparse representation has the potential discriminative ability since most nonzero elements are located on the samples in the same class as the represented sample.

Given the training sample $L = [l_1, l_2, \ldots, l_N] \in \mathbb{R}^{N \times m}$, a test sample $H \in \mathbb{R}^m$, the solution to the sparse representation problem can be obtained by solving the following $\ell^1$-minimization problem:

$$\hat{\beta} = \arg \min_{\beta} \| \beta \|_1 \quad \text{subject to } L\beta = H \quad (16)$$

where $\hat{\beta} = [\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_N]$ are the sparse representation coefficients and $\text{Card}(&\hat{\beta})$ denotes the number of nonzero elements of $\beta$. From the perspective of statistics, formula (16) can be named the Lasso criterion. Lasso is a penalized least squares algorithm which was originally by quadratic programming imposing a constraint on the $L_1$ norm of the regression coefficients. Thus, the Lasso estimates $\hat{\beta}$ are obtained by minimizing the Lasso criterion:

$$\hat{\beta} = \arg \min_{\beta} \| H - \sum_{i=1}^{N} \lambda_i | \beta_i | \|^{2} + \lambda \sum_{i=1}^{N} | \beta_i | \quad (17)$$

subject to $\text{Card}(&\hat{\beta}) \leq k$,

where $\lambda$ is nonnegative. However, only using the $L_1$-norm penalty in Lasso has its limitation. Zou et al. [45] proposed that if there is a group of variables among which the pairwise correlations are very high, lasso tends to select any variable from the group and does not consider which one is selected. Fortunately, elastic net was put forward by Zou et al. which overcomes the limitation of only using the $L_1$-norm penalty. It is known that combining the $L_1$-norm and $L_2$-norm penalty can result in grouping effectiveness in regression and thus enhance the prediction accuracy. For any nonnegative $\lambda_1$ and $\lambda_2$, the elastic net estimates $\tilde{\beta}_{en}$ are given by

$$\tilde{\beta}_{en} = \arg \min_{\beta} \| H - \sum_{i=1}^{N} L_i | \beta_i | \|^{2} + \lambda_1 \sum_{i=1}^{N} | \beta_i | \quad (18)$$

subject to $\text{Card}(&\hat{\beta}) \leq k$.

In brief, it is expected that the elastic net is used to group a set of sparse coefficients to construct the sparse alignment matrices, in which the sparse representation information or the potential discriminative information is encoded to enhance the discriminative ability in an unsupervised manner.

3.3. Fault Detection with SPCS and BIP. The key problem for monitoring the multimode process is to select a suitable model and choose the subspace spanned by key PCs. In the Introduction, we had put forward the fact that the subspace
spanned by the first several PCs with largest explained variance does not always have fault information.

In the following part, a novel multimode process monitoring approach based on SPCS and BIP is proposed. This approach is in a just-in-time form. For each sample, an elastic net regression between all PCs and the sample is constructed and solved. The PCs which have nonzero regression coefficients are retained, while other PCs are rejected. That means, for each sample, we can pick out the most discriminative bases and the others are set to zero. Its concrete calculating steps are summarized in Figure 1.

**Offline Modeling**

1. Collect a set of historical training data under all possible operating conditions.
2. Use the EM algorithm to learn the Gaussian mixture model and estimate the model parameter set \( \Theta = \{\mu_1, \Sigma_1, \omega_1, \ldots, \mu_k, \Sigma_k, \omega_k\} \) based on the iterative steps.
3. For each submodel, get a normal operational observation set \( \mathbf{X} = \mathbb{R}^{N \times m} \), where \( N \) is the number of samples and \( m \) is the number of variables. This set is denoted as the training set for threshold determining. A testing set \( \mathbf{Y} = \mathbb{R}^{S \times m} \) with both normal and abnormal observations is given for testing.
4. Normalize the training data through the mean value and variance of each variable.
5. Obtain all principal components using SVD decomposition. The training data \( \mathbf{X} \) is reconstructed by \( \mathbf{X} = \sum_{i=1}^{m} t_i p_i^T \), where \( t_i \) is the score vector and \( p_i \) is the loading vector.
6. For training sample \( \mathbf{x}_j \ (j = 1, 2, \ldots, N) \), construct an elastic net regression between each observation value of training data and loading vector \( \mathbf{P} \) made of PCs in step (5), according to \( \hat{\beta}_j = \arg\min_\beta \| \mathbf{y}_j - \sum_{i=1}^{m} p_i \beta_i \|_2^2 + \lambda_1 \sum_{i=1}^{m} | \beta_i | + \lambda_2 \sum_{i=1}^{m} \| \beta \|_2 \) subject to \( \text{Card}(\beta) \leq k \).
7. Corresponding to the nonzero representation coefficients \( \{\hat{\beta}_j, \hat{\beta}_j^1, \ldots, \hat{\beta}_j^k\} \), construct a new loading vector \( \hat{\mathbf{P}}_j = [p_{j1}, p_{j2}, \ldots, p_{jk}] \).
8. Specify a confidence \((1-\alpha)100\%\) and constant control limit \(1-\alpha\).

**Online Monitoring**

1. Normalize the current time point data by using mean values and variance of the training data.
2. Obtain the loading vector \( \mathbf{P} \) from offline modeling.
3. When a test sample \( \mathbf{y}_j \in \mathbb{R}^{m} \ (j = 1, 2, \ldots, S) \) is coming, construct an elastic net regression between the sample and loading vector \( \mathbf{P} \) made of PCs in step (2), according to \( \hat{\beta} = \arg\min_\beta \| \mathbf{y}_j - \sum_{i=1}^{m} p_i \beta_i \|_2^2 + \lambda_1 \sum_{i=1}^{m} | \beta_i | + \lambda_2 \sum_{i=1}^{m} \| \beta \|_2 \) subject to \( \text{Card}(\beta) \leq k \).
(4) Corresponding to the nonzero representation coefficients \( \{ \hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_k \} \), construct a new loading vector \( P_j = [p_{j1}, p_{j2}, \ldots, p_{jk}] \).

(5) Generate the BIP control chart with the calculated BIP index values for all the monitored samples. If the BIP index of a test sample is lower than the control limit, which means the sample is normal, go to step (1). Else, there is a fault in the process.

4. Case Studies on the TE and CSTR Process

In this case study, the TE benchmark and CSTR process are introduced to verify the effectiveness of the SPCS-BIP algorithm. PCA-GMM is the classic algorithm for multimode processing monitoring. And the fault detection index (FDI) is similar to Bayesian inference probability (BIP). So here, a comparison was made between SPCS-BIP and PCA-GMM. In addition, to verify the improvements of SPCS algorithm, which can select sparse PCs, a comparison was performed between the SPCS-BIP algorithm and the MPPCA algorithm.

4.1. Tennessee Eastman Process. As a well-known benchmark process, the Tennessee Eastman process, which was presented by Downs and Vogel, has been widely applied to evaluate and compare the efficiency of process monitoring techniques [46, 47]. The schematic diagram of the process is illustrated in Figure 2. This process consists of five major unit operations: a reactor, a product condenser, a vapor-liquid separator, a recycle compressor, and a product stripper. In addition, there are six modes of process operation, as listed in Table 1. The variables can be divided into three categories: composition variables, continuous process variables, and manipulated variables. In our study, only modes 1 and 3 were simulated through the Simulink programs developed on the basis of the decentralized control strategy designed by Ricker [48]. The Simulink programs can be downloaded from http://depts.washington.edu/control/LARRY/TE/download.html. The 31 selected monitoring variables contained 9 manipulated variables and 22 continuous process variables. Thus, these variables were divided into five subblocks according to five units. However, given that only two variables were allocated to each compressor unit and the condenser unit, there were four variables assigned to the other three related subblocks. As a result, the total of 31 variables was divided into three subblocks.

There are 20 faults in the multimode TE process, which are listed in Table 2, are simulated and the corresponding process data are collected for testing. The following simulations are run in MATLAB 8.3.0 (2014a) environment. Here, two indicators, which are FR (FR) and MR (MR), are often introduced to measure the result of process monitoring. FR is the rate of normal data classified as fault data. MR is the rate of fault data classified as normal rate.

In the MPPCA algorithm and PCA-GMM algorithm, when the variance contribution was selected as 85%, the dimension of feature space in MPPCA and the number of PCs in PCA-GMM were each selected as 18. In order to compare the monitoring performances of these algorithms in the same situation, the selected sparse PCs of each mode in SPCS-BIP were selected as 18. The 99% control limit was assigned to all three algorithms.

First, Figure 3 shows that the different submodes can be successfully divided by the EM algorithm used in this paper. And by using other algorithms, the modes also can be divided correctly. In other words, how to correctly divide the training data into multiple subset is not a problem by many related algorithms.

The normal process was tested by different algorithm, and the results are shown in Figure 4. In this figure, it is hard to figure out which algorithm’s FR is lower. In the figure, most samples of each algorithm are lower than the control limit. And by calculation, the FR of these algorithms are 0.333%, 0.08%, 0.25%, and 1.08%, respectively, corresponding to Figures 4(a), 4(b), 4(c), and 4(d). The monitoring performances of these three algorithms suggest that the FR are acceptable.

Next, the data sets of 12 faults in mode 3 were tested, and the MR of these three algorithms are listed in Table 3, with the smallest MR shown in bold.

From Table 3, we observe that the monitoring performance of SPCS-BIP is the best, compared to the MPPCA and PCA-GMM algorithms for all 12 faults. Here, we take the further analysis. In comparison with MPPCA and PCA-GMM algorithms, the SPCS-BIP algorithm can exactly divide the process data into subgroups corresponding to different modes by using the E-M algorithm, and, in each submode, SPCS-BIP can select the most important PCs that have most relation with the fault. Due to the fact that the subspace spanned by the PCs was monitored by BIP, most of the PCs are related to the main process of chemical industrial process, and only little PCs are related to the fault process. SPCS are discriminative by constructing an elastic net regression between all PCs and each sample. So, in Table 3, we observe
Table 2: Process faults for the multimode TE process.

<table>
<thead>
<tr>
<th>Faults number</th>
<th>Disturbance state</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDV(1)</td>
<td>A/C feed ratio, B composition constant (Stream 4)</td>
<td>Step</td>
</tr>
<tr>
<td>IDV(2)</td>
<td>B composition, A/C ratio constant (Stream 4)</td>
<td>Step</td>
</tr>
<tr>
<td>IDV(3)</td>
<td>D feed temperature (Stream 2)</td>
<td>Step</td>
</tr>
<tr>
<td>IDV(4)</td>
<td>Reactor cooling water inlet temperature</td>
<td>Step</td>
</tr>
<tr>
<td>IDV(5)</td>
<td>Condenser cooling water inlet temperature</td>
<td>Step</td>
</tr>
<tr>
<td>IDV(6)</td>
<td>A feed loss (Stream 1)</td>
<td>Step</td>
</tr>
<tr>
<td>IDV(7)</td>
<td>C header pressure loss reduced availability (Stream 4)</td>
<td>Step</td>
</tr>
<tr>
<td>IDV(8)</td>
<td>A, B, and C feed composition (Stream 4)</td>
<td>Random variation</td>
</tr>
<tr>
<td>IDV(9)</td>
<td>D feed temperature (Stream 2)</td>
<td>Random variation</td>
</tr>
<tr>
<td>IDV(10)</td>
<td>C feed temperature (Stream 4)</td>
<td>Random variation</td>
</tr>
<tr>
<td>IDV(11)</td>
<td>Reactor cooling water inlet temperature</td>
<td>Random variation</td>
</tr>
<tr>
<td>IDV(12)</td>
<td>Condenser cooling water inlet temperature</td>
<td>Random variation</td>
</tr>
<tr>
<td>IDV(13)</td>
<td>Reaction kinetics</td>
<td>Slow drift</td>
</tr>
<tr>
<td>IDV(14)</td>
<td>Reactor cooling water valve</td>
<td>Sticking</td>
</tr>
<tr>
<td>IDV(15)</td>
<td>Condenser cooling water valve</td>
<td>Sticking</td>
</tr>
<tr>
<td>IDV(16)</td>
<td>Unknown</td>
<td>Unknown</td>
</tr>
<tr>
<td>IDV(17)</td>
<td>Unknown</td>
<td>Unknown</td>
</tr>
<tr>
<td>IDV(18)</td>
<td>Unknown</td>
<td>Unknown</td>
</tr>
<tr>
<td>IDV(19)</td>
<td>Unknown</td>
<td>Unknown</td>
</tr>
<tr>
<td>IDV(20)</td>
<td>Unknown</td>
<td>Unknown</td>
</tr>
</tbody>
</table>

Figure 2: Control scheme for the TE process.
Figure 3: Different modes of the training data.

Table 3: Missed detection rates (%) of 12 faults.

<table>
<thead>
<tr>
<th>Faults number</th>
<th>MPPCA-T²</th>
<th>MPPCA-SPE</th>
<th>PCA-GMM</th>
<th>SPCS-BIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.75</td>
<td>0.25</td>
<td>1</td>
<td>0.125</td>
</tr>
<tr>
<td>2</td>
<td>3.5</td>
<td>8.5</td>
<td>5.5</td>
<td>1.875</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0.375</td>
<td>0</td>
<td>1.125</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>3.375</td>
<td>3.875</td>
<td>4.25</td>
<td>2.500</td>
</tr>
<tr>
<td>10</td>
<td>82.375</td>
<td>16</td>
<td>89.875</td>
<td>7.875</td>
</tr>
<tr>
<td>11</td>
<td>2.25</td>
<td>8.5</td>
<td>4.5</td>
<td>1.125</td>
</tr>
<tr>
<td>12</td>
<td>1.375</td>
<td>1.625</td>
<td>1.5</td>
<td>0.750</td>
</tr>
<tr>
<td>13</td>
<td>16.125</td>
<td>26.5</td>
<td>1.9625</td>
<td>11.625</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>3.5</td>
<td>0.125</td>
<td>0</td>
</tr>
</tbody>
</table>

4.2. CSTR. This study simulated the CSTR process described by Yoon and MacGregor [49]. The diagram of the process is presented in Figure 6. Due to the fact that the CSTR process consisted of only one operating unit, the number of subblocks was selected as 1.

Table 4: Missed detection rates (%) of two faults.

<table>
<thead>
<tr>
<th>Faults number</th>
<th>MPPCA-T²</th>
<th>MPPCA-SPE</th>
<th>PCA-GMM</th>
<th>SPCS-BIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>90</td>
<td>100</td>
<td>98.8</td>
<td>65.6</td>
</tr>
<tr>
<td>2</td>
<td>28.8</td>
<td>99.6</td>
<td>31.8</td>
<td>28</td>
</tr>
</tbody>
</table>

In the modeling stage, 1000 samples which include 500 mode 1 samples and 500 mode 2 samples were collected as the training data set. In the testing stage, 1000 samples of mode 2 were tested, and two faults were introduced to the process as follows.

Case 1. A step of 1 K was added in the cooling water temperature $T_C$ from the 500th sample.

Case 2. A 2 kmol/($m^3 \cdot min$) step was added in the inlet solute concentration $C_{AA}$ from the 500th sample.

In the MMPCA algorithm, when the variance contribution was selected as 85%, the dimension of feature space in MPPCA is 10. So, in order to compare the monitoring performances of these algorithms in the same situation, the number of PCs in PCA-GMM and the selected sparse PCs in SPCA-BIP were both selected as 10. The 99% control limit was assigned to all three algorithms.

The same as TEP, the FR of these algorithms are 0.4%, 0, 1.2%, and 1%, respectively. In an industry process, FR lower than 0.05 is acceptable [28].

The data sets of two faults in mode 2 were tested, and the MR were listed in Table 4. In the table, the smallest missed detection rates are shown in bold.

As shown in Table 4, the SPCS-BIP algorithm has shown the best performance for these two faults compared with other algorithms listed in the table. It is obvious that neither MPPCA nor PCA-GMM algorithms can detect the fault because their missed detection rates were high. In those four algorithms, only the SPCS-BIP was based on the selection PCs, so the improvements in the proposed sparse principal components selection can be demonstrated through the better monitoring performance of the SPCS-BIP algorithm.

Fault 1 is a bias in cooling water temperature $T_C$. Due to the control loop in the CSTR process, there would be biases in outlet temperature $T$, then the cooling water flow rate $F_C$ would increase. In Figure 7, both the MPPCA and PCA-GMM algorithms could not detect fault 1 effectively according to the performances of those shown in Figures 7(a), 7(b), and 7(c). In Figure 7(d), it is obvious that the monitoring performance of SPCS-BIP is much better than the others. The reason is that the correct classification for each subgroup by using E-M algorithm and the PCs selected by SPCS are discriminative and could construct the subspace that contains the important fault information for abnormal data.

Fault 2 is a bias in inlet solute concentration $C_{AA}$. Then, due to the control loop in the CSTR process, there would be biases in outlet concentration $C$ and outlet temperature $T$. According to the performances shown in Figure 8(b), the FDI of MPPCA-SPE could not detect fault 2. Compared to the MPPCA-SPE, the FDI of MPPCA-T² showed some
Figure 4: Monitoring performance of the normal process.

Figure 5: Monitoring performances of fault 10 in TEP.
improvements. Both PCA-GMM and the proposed SPCS are all using BIP. In Figures 8(c) and 8(d), we could hardly see which algorithm is better. However, in Table 4, we could obviously find that the SPCS-BIP is better. Even, compared to MPPCA-$T^2$, the proposed algorithm has a little advantage than MPPCA-$T^2$.

5. Conclusions

An algorithm using sparse principal component selection and Bayesian inference-based probability (SPCS-BIP) was proposed in this study. Given that the modern industrial processes typically have multiple operating modes, BIP is utilized to compute the posterior probabilities of each monitored sample belonging to the multiple components and derive an integrated global probabilistic index for fault detection of multimode processes. In each submode, we use the sparse principal component selection to select the key PCs that have the best relation with fault. This algorithm constructs an elastic net regression between all PCs and each sample and then selects PCs according to the nonzero regression coefficients which indicate the discriminative expression of the sample. Finally, the TE and CSTR processes were employed to verify the superiority of the SPCS-BIP algorithm. The monitoring performances of MPPCA, PCA-GMM, and SPCS-BIP methods are discussed compared to those of the MPPCA and PCA-GMM algorithms, and the monitoring performances of the SPCS-BIP algorithm were found to be the best ones among the three algorithms.
Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant no. 61375007) and Shanghai Science and Research Projects (Grant nos. 15JC1400600, 15JC1401700).

References


