Chemometrics and Intelligent Laboratory Systems 158 (2016) 102–116

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

journal homepage: www.elsevier.com/locate/chemolab

Fault detection in time-varying chemical process through incremental principal component analysis

Yong Gao [a], Xin Wang [b], Zhenlei Wang [a,*], Liang Zhao [a]

[a] Key Laboratory of Advanced Control and Optimization for Chemical Processes (East China University of Science and Technology), Ministry of Education, Shanghai 200237, China
[b] Center of Electrical & Electronic Technology, Shanghai Jiao Tong University, Shanghai, China

ARTICLE INFO

Article history:
Received 8 April 2016
Received in revised form 7 July 2016
Accepted 12 July 2016
Available online 15 July 2016

Keywords:
Time-varying
Slow ramp fault
Incremental PCA
Fault detection

ABSTRACT

Incremental principal component analysis (IPCA) is proposed to improve the detection performance of a slow ramp fault in the time-varying chemical process. Conventional monitoring methods of the time-varying process such as recursive method and moving window strategy, which update the monitoring model and control limit when the newly monitored sample is detected as a normal one, track the slow ramp fault and lose the ability to detect this kind of fault. In this study, the incremental principal components (IPCs) describing time-varying information are proposed to extract the normal time-varying information. This study proposes IPCA method based on IPCs for process monitoring of the time-varying processes. The monitoring model remained unchanged because the normal time-varying information has already been identified by IPCs. The method can distinguish between the slow ramp fault from the normal time-varying process. Two numeric case studies demonstrate the efficiency of the method. Application of the method to an acetylene hydrogenation reactor is also provided.

1. Introduction

Modern industrial processes are large-scale interconnected systems. With the increase in demand for plant safety and product quality, process monitoring has played an increasingly important role in chemical industrial procedures. In past decades, multivariate statistical process monitoring (MSPM) has received significant attention [1–4]. Conventional MSPMs, such as principal component analysis (PCA) and partial least squares (PLS) have been widely applied to process monitoring. These methods can acquire satisfactory results for the time-invariant process with a single steady operation mode [5,6]. However, many process variables are time-varying because several reasons, including deactivation of a catalyst, measurement instrument drifting, and other reasons. In this situation, conventional MSPMs often yield undesirable results.

To date, many complementary MSPM methods have been designed to solve the time-varying problems. Three adaptive monitoring methods have been proposed: recursive, exponential weighting, and moving window methods. Recursive methods absorb the new normal sample into the monitoring model to adapt to the time-varying processes. In [7,8], two recursive PCA methods are proposed. Wang developed a recursive PLS method [9]. In [10], a recursive kernel PCA method is further studied to deal with the nonlinear processes. Exponential weighting methods update the monitoring model by absorbing new normal data and forgetting the old data exponentially. In [11], an exponential weighting moving average PCA (EWMA-PCA) method updates the monitoring model by recalculating the mean and variance of the modeling data. EWMA-PCA [12] and EWPLS [13] refresh the monitoring model by updating the covariance matrix. Moving window methods update the monitoring model by discarding the oldest sample and inserting a new normal sample. In [14], a fast moving window PCA method enabled the online application of a generic moving window-based PCA with larger window size. A variable MWPCA method for adaptive process monitoring with a variable window size is proposed in [15]. A moving window local outlier factor method [16] monitors the non-Gaussian time-varying process with complex data distribution. A moving window local outlier probabilities method is proposed in [17], and a dynamic Gaussian mixture model (GMM) based on the moving window strategy in [18]. The just-in-time learning strategy is introduced for modeling and process monitoring in [19]. However, most methods are based on the idea that the monitoring model updates online when the newly monitored sample is judged normally, which will track the slow ramp fault.

In solving the time-varying problem and distinguishing the normal time-varying process from slow ramp fault process, an incremental PCA (IPCA) method is proposed. The normal time-varying information is first extracted using the incremental principal components (IPCs). The changeless monitoring model with normal time-varying information is then built based on the IPCs. Finally, a new statistic is constructed to detect the process fault online.

The rest of this paper is organized as follows. The basic PCA and proposed IPCA methods are introduced in Section 2. Section 3 presents the

* Corresponding author.
E-mail address: wangzhen_l@ecust.edu.cn (Z. Wang).

http://dx.doi.org/10.1016/j.chemolab.2016.07.005
0169-7439/© 2016 Published by Elsevier B.V.

details of the fault detection method based on IPCs. A numeric simulation, which demonstrates the drawback of the conventional adaptive process monitoring method and feasibility of the incremental PCA method, is introduced in Section 4. Sections 5 and 6 discuss several examples to show the effectiveness of the proposed method. Finally, the conclusion is given in Section 7.

2. PCA and IPC

2.1. PCA

PCA is widely used in data compression, modeling, singular value detection, and classification, among others. The objective of PCA is to acquire the irrelevant scores that express most variations in the covariance matrix. Geometrically, PCA rotates the coordinate axes to the directions with maximum variances of the covariance matrix.

Let $X \in R^{N\times n}$ be the standardized data matrix with zero mean and unit variance, where N denotes the sample number, and n denotes the variable number. The PCA model can be described as Eq. (1):

$$X = TP^T + E = \hat{X} + E \quad , \tag{1}$$

where $T \in R^{N\times r}$ is the score matrix and $P \in R^{n\times r}$ is the loading matrix. The parameter r denotes the number of retained principal components (PCs). The score matrix T and loading matrix P can be acquired through eigenvalue decomposition algorithm, and the loading matrix is composed of the eigenvectors of the covariance matrix. The number of retained PCs is usually determined by the cumulative percentage of variance (CPV) method as shown in Eq. (2):

$$\sum_{i=1}^{r} \lambda_i \Big/ \sum_{i=1}^{n} \lambda_i \times 100\% \geq 85\% \quad , \tag{2}$$

where λ_i denotes the variance of the corresponding score vector and is sorted in descending order. The r PCs with CPV larger than 85% are selected.

Two statistics, T^2 and SPE, are constructed in subspaces spanning the score subspace $\hat{X}$ and residual subspace E for a new sample $x \in R^{n\times 1}$, as shown in Eqs. (3) and (4):

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

$$SPE = e^T e = \left(x - PP^T x\right)^T \left(x - PP^T x\right) \quad , \tag{4}$$

where $\Lambda \in R^{r\times r}$ is a diagonal matrix consisting of variances of the score vectors. The corresponding control limits are determined as Eqs. (5) and (6):

$$T^2_{\lim} = \frac{r(N-1)}{N-r} F_{r,(N-r),\alpha} \tag{5}$$

$$SPE_{\lim} = g\chi^2_{h,\alpha} \quad , \tag{6}$$

where $F_{r,(N-r),\alpha}$ is the quantile of F-distribution with the freedom degree $r, (N-r)$ and the confidence level α, $\chi^2_{h,\alpha}$ is the quantile of χ^2-distribution with the freedom degree h and the confidence level α, $g = v/2m$, $h = 2m^2/v$, and m and v denote the mean and variance of the SPE statistics of the modeling samples, respectively.

2.2. IPC

The PCA method is effective under the assumption that the process is not time-varying, which indicating that both the mean and variance of the process variables are steady. For the time-varying process, which the mean and variance of the process variables may be changeable, some adaptive methods are usually employed, such as moving window PCA. However, the adaptive methods usually update the monitoring model with judged normal samples and easily track the slow ramp fault. The monitoring model should be changeless, and the normal time-varying information should be added to the monitoring model to distinguish between the normal time-varying and slow ramp fault processes.

For the time-varying process, the irrelevant PCs are also time-varying. Time-varying information can be described by the increments of the PCs. The increments of the PCs can be calculated by subtracting two PCs in two different moments. Therefore, to describe the normal time-varying information, the IPC is proposed to denote the variations of the PCs. The i^{th} dimension of the k^{th} IPC can be calculated by Eq. (7):

$$IPC_i = \text{mean}(PC_i(k-L:k)) - \text{mean}(PC_i(k-W-2L:k-W\text{-}L)) \tag{7}$$

In Eq. (7), IPC_i refers to the difference between two PCs in two moments. To make IPC_i more sensitive to the process fault, two mean PCs are subtracted. In Eq. (7), L denotes the number of PCs used to calculate the mean to be subtracted, and W denotes the interval of the two moments. Moreover, $W > L$. Parameter L should be as large as possible to detect the tiny fault of the process empirically.

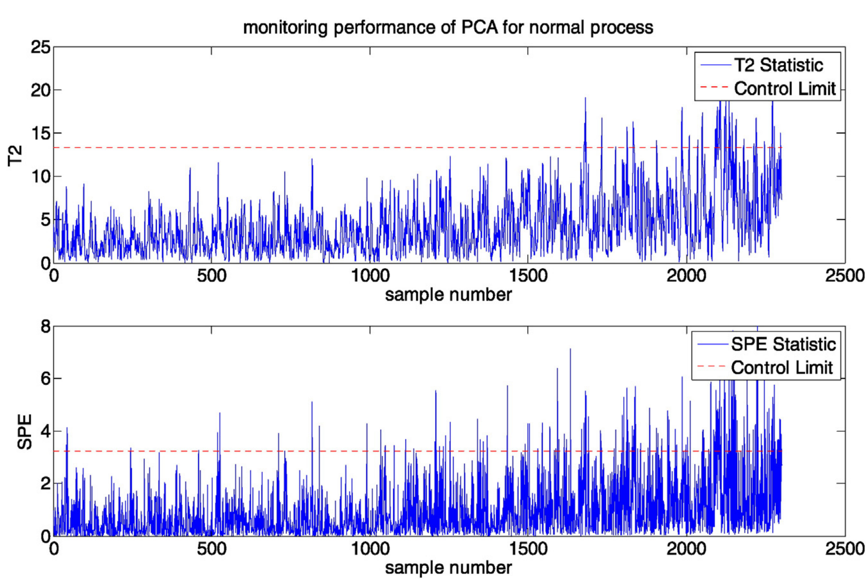


Fig. 1. Monitoring performance of conventional PCA for normal process.

However, the false rate alarm increases sharply when parameter L is excessively large. Parameter W affects the missing alarm rate, which should choose an appropriate value. The time-varying information can be extracted with IPC. The procedure of acquiring the IPC is described as follows.

A modeling dataset with normal operating samples is required in illustrating the IPCs. Suppose $A \in N \times m$ is a normal process dataset used for modeling, where N is the sample number of the dataset and m is the number of the variables. Dataset A is used to build the conventional PCA model, which is used to calculate the PCs and to acquire the modeling IPCs. After the loading matrix and PCs are calculated, the corresponding IPCs are calculated according to Eq. (7). $IPC_{i,j}$ describes the increment of the jth PC of the i^{th} sample in the normal time-varying process. $IPC_{i,j}$ should be under the control limit CL_j under normal conditions. The control limit CL_j can be determined easily using the kernel density estimate (KDE) [20,21], which is an effective nonparametric probability density estimation method. After the probability density function of $IPC_{i,j}$ is estimated, the control limit CL_j can be determined with a 99% confidence level. In the online monitoring stage, the IPC_i of the i^{th} sample is calculated online using Eq. (7). The IPCs generate a step change for the slow ramp fault and a pulse for the step change fault. Therefore, the missing alarm rate of the step change fault will be high as the IPC_i when it resumes normally after the pulse. The change in IPCs should be maintained after the pulse to guarantee low missing alarm rate. When $IPC_{i,j} > CL_j$, the corresponding $PC_{i,j}$ of the last sample is replaced by $PC_{i-1,j}$. With this operation, the IPCs generate a step change for the step change fault and a ramp change for the slow ramp fault, resulting in the enhancement of fault detection performance. With the IPC_i of the i^{th} sample, a new statistic IT_i^2 can be constructed as Eq. (8):

$$IT_i^2 = IPC_i^T \Lambda_{IPCs}^{-1} IPC_i \quad , \tag{8}$$

where Λ_{IPCs} is a diagonal matrix and each element is the variance of IPC_j of the dataset A. Thereafter, the control limit CL can be estimated by the KDE algorithm with a 99% confidence level.

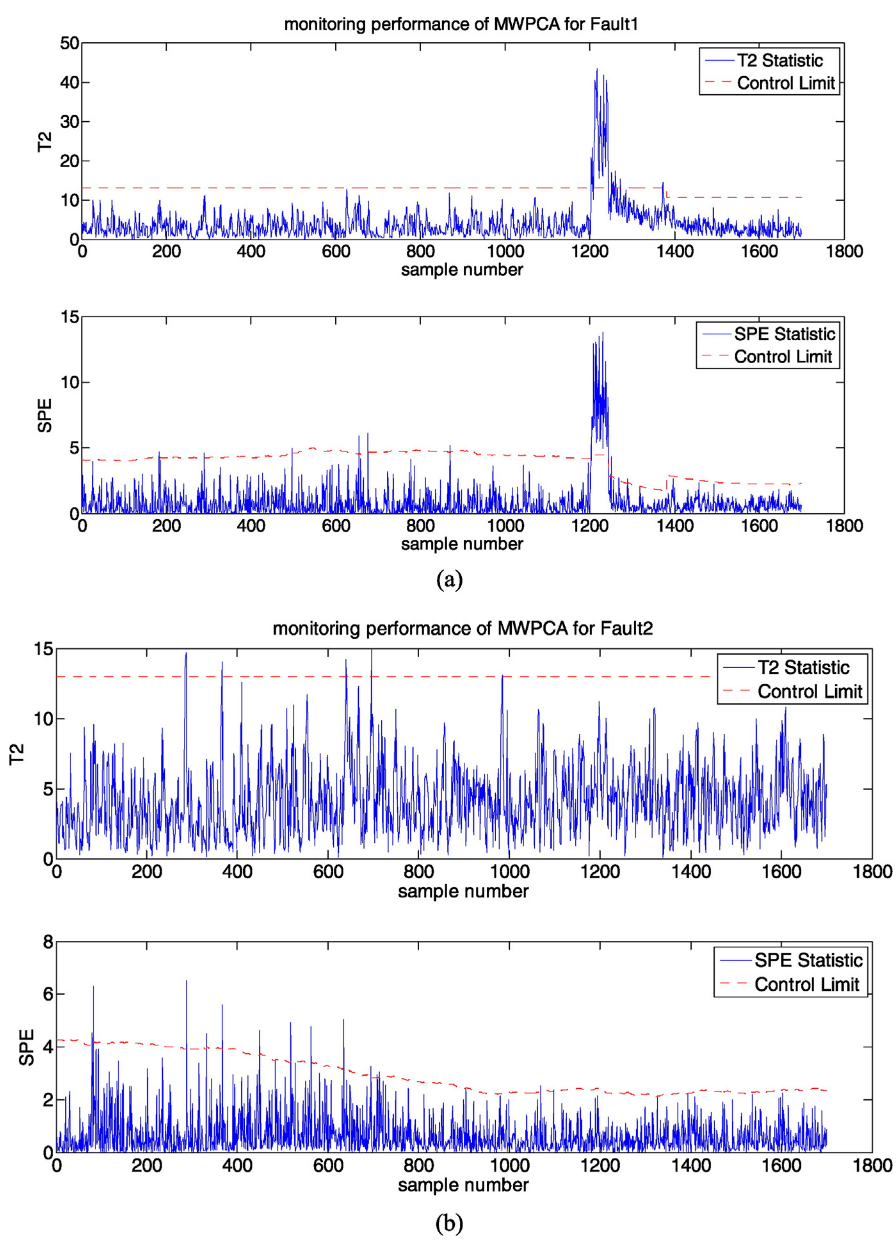


Fig. 2. Monitoring performance of MWPCA for (a) Fault1 (b) Fault2.

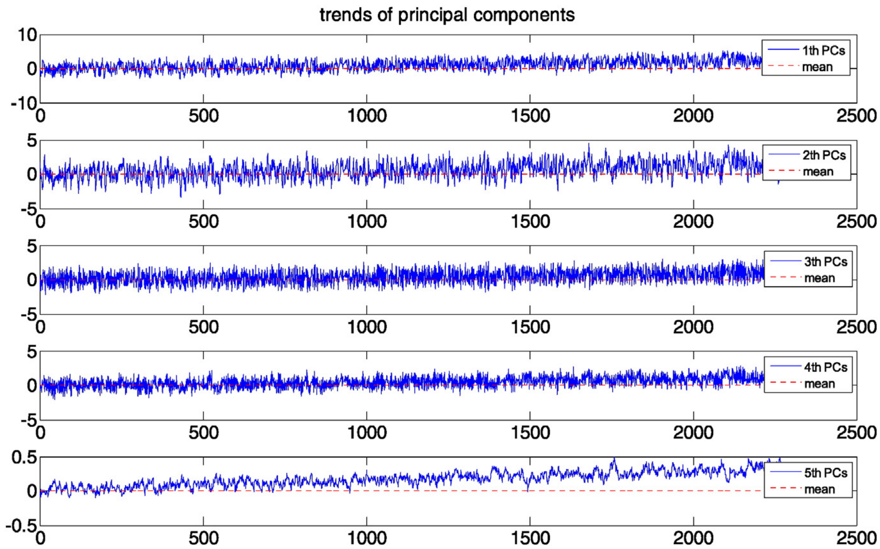


Fig. 3. Trends of PCs for normal process.

3. Fault detection based on IPCA

According to the description above, the IPCs and statistic IT^2 are used to monitor a time-varying process. The IPCA-based fault detection method, which has two stages, can be described as follows.

Offline modeling stage:

(1) Acquire a dataset A with normal operating samples used to acquire the loading matrix and the $IPCs$ dataset to determine the control limit. Normalize dataset A through the mean and variance of each variable;
(2) Build the conventional PCA model with dataset A using the eigenvalue decomposition algorithm without reducing the dimension, and acquire the loading matrix P and PCs;
(3) Calculate the $IPCs$ according to Eq. (7) in dataset A, and then calculate the corresponding IT^2 of each sample with the $IPCs$ according to Eq. (8); and
(4) Use the KDE algorithm to determine the control limit CL_j and CL of IT^2 as well as the threshold CL_j of IPC_j with the 99% confidence level.

Online fault detection stage:

(1) Collect N_1 normal online samples, where $N_1 > W + 2 * L$. Normalize the samples through the means and variances of the training dataset A, and calculate the PCs with the loading matrix P;
(2) Collect a new online sample. Normalize the sample with the means and variances of the training dataset A and calculate the PCs with the loading matrix P of the PCA model;
(3) Calculate the IPC_i of the i^{th} sample with the PCs of the forward $W + 2 * L$ samples through Eq. (7);
(4) Calculate the statistic IT_i^2 of the i^{th} sample with the IPC_i of the i^{th} sample through Eq. (8);
(5) If $IT_i^2 > CL$, then a fault may be present in the process. Otherwise, the sample is a normal one. In addition, when $IPC_{i,j} > CL_j$, the corresponding $PC_{i,j}$ is replaced by $PC_{i-1,j}$; and
(6) Repeat (2)–(5).

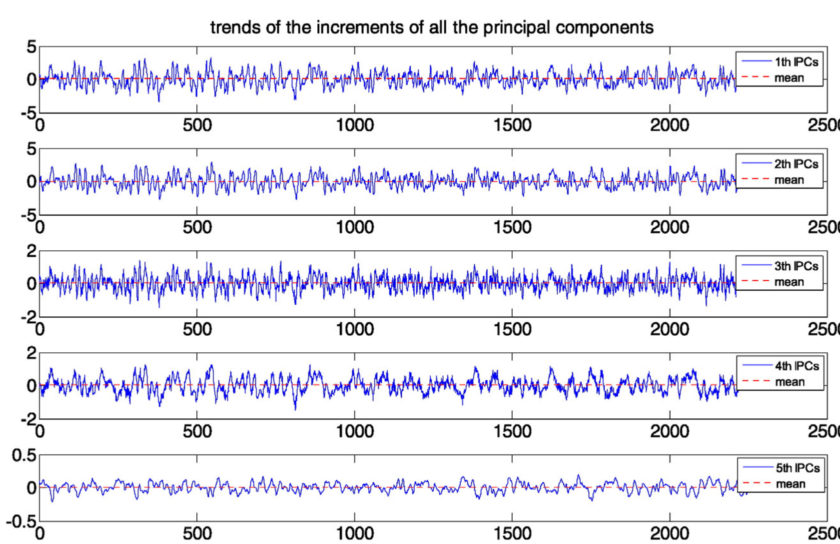


Fig. 4. Trends of the increments of all the PCs for normal process.

4. Case study on a numeric system

In this section, a simulation of a numeric multivariate process suggested by [22] and modified by [23] is employed, and the time-varying processes is added. The process can be described as follows.

$$\mathbf{z}(i)=\begin{bmatrix}0.018 & -0.191 & 0.287\\ 0.847 & 0.264 & 0.943\\ -0.333 & 0.514 & -0.217\end{bmatrix}\mathbf{z}(i-1)+\begin{bmatrix}1 & 2\\ 3 & -4\\ -2 & 1\end{bmatrix}\mathbf{u}(i-1) \quad (9)$$

$$\mathbf{y}(i)=\mathbf{z}(i)+\mathbf{v}(i) \quad (10)$$

$$\mathbf{u}(i)=\begin{bmatrix}0.811 & -0.226\\ 0.477 & 0.415\end{bmatrix}\mathbf{u}(i-1)+\begin{bmatrix}0.193 & 0.689\\ -0.320 & -0.749\end{bmatrix}\times[\mathbf{h}(i-1)+0.002\times(i-1)] \quad (11)$$

In this numeric system, variable $\mathbf{h}$ is a random vector with each element uniformly distributed over the interval $(-2, 2)$. The output $\mathbf{y}$ is the status variable $\mathbf{z}$ plus a random noise vector $\mathbf{v}$, and each element of $\mathbf{v}$ is normally distributed with 0 mean and 0.1 variance. The input $\mathbf{u}$ and output $\mathbf{y}$ are measured for process monitoring. A total of 2500 samples represented as $\mathbf{x}(i)=[\mathbf{y}^T(i)\mathbf{u}^T(i)]^T$are generated by this system. Two faults occur in the process.

Fault 1: step change of u_1 by 2.0 is introduced at the 2001th sample.
Fault 2: linear ramp with 0.002 increment of u_1 is introduced starting from the 1001th sample.

The conventional PCA method is employed to monitor the normal process. Among the 2500 samples, 200 samples are employed to build the PCA method. The confidence limit is set to 99.5%. The monitoring performance is shown in Fig. 1.

Fig. 1 shows that the statistics increase slowly because the process is time-varying. However, the conventional PCA model is changeless. Therefore, the fault alarm rate is excessively high when the model does not match the process. For the time-varying process, the traditional methods usually update the monitoring model online, and the moving window strategy is often employed to update the model. The moving window PCA method is employed to monitor the process. Window size is empirically selected to be 800, and the monitoring performances are shown in Fig. 2.

The step change is introduced at the 2001th sample, and the first 800 samples are used to build the PCA model. As shown in Fig. 2(a), the monitoring performance of MWPCA for Fault 1 is acceptable. The statistics can give timely alarms, and the PCA model and control limits can be updated for the succeeding 1200 normal samples. The fault alarm rate is low for the time-varying process. However, for the slow ramp fault such as Fault 2, the statistics should provide alarms that start from the 201th monitored sample. However, the monitoring model and control limits are updated aimlessly, as shown in Fig. 2(b). The MWPCA method easily tracks the slow ramp process and has no ability to distinguish the normal time-varying and slow ramp fault process.

A direct method is employed to keep the monitoring model changeless and to distinguish the normal between the time-varying and slow ramp fault processes. However, the changeless model will yield a high fault alarm rate for the time-varying process, and requires the addition of the normal time-varying information to the changeless monitoring model. All PCs for each monitored sample of the normal time-varying process are calculated to investigate the time-varying information of the normal process. The trends of all PCs are shown in Fig. 3.

As shown in Fig. 3, the red lines denote the means of all PCs of the PCA model. For the normal time-varying process, the PCs deviate from the means slowly, causing the conventional PCA statistics to shift as well. In acquiring the time-varying information and adding information to the monitoring model, the increments of all PCs or IPCs are calculated through Eq. (7), and the parameters L and W are shown in Fig. 3. Set $L=6$ and $W=40$; then, the trends of the increments of all PCs are shown in Fig. 4.

Fig. 4 shows that the increments of all PCs remained around their means, and thus, a statistic constructed by the IPCs will not increase slowly for the normal time-varying process. The time-varying information is extracted by the IPCs. The trends of all IPCs for Fault 2 are shown in Fig. 5.

As shown in Fig. 5, when Fault 2 occurred, the fifth IPC obviously deviated from the mean. This finding indicates that IPCs can be used to construct a statistic to monitor the process. For the time-varying process, the conventional changeless PCA model and moving window PCA with the updated model cannot yield a desirable monitoring performance. However, with IPCs, the normal time-varying and slow ramp fault processes can be distinguished, and thus, time-varying information extraction and modeling with IPCs can address the monitoring problem of the time-varying process.

With the IPCA, the monitoring performances of the normal numeric process, Fault 1 process, and Fault 2 process discussed above are shown in Fig. 6.

Fig. 6 shows that although the IPCA method does not update the monitoring model online, the time-varying information is added to

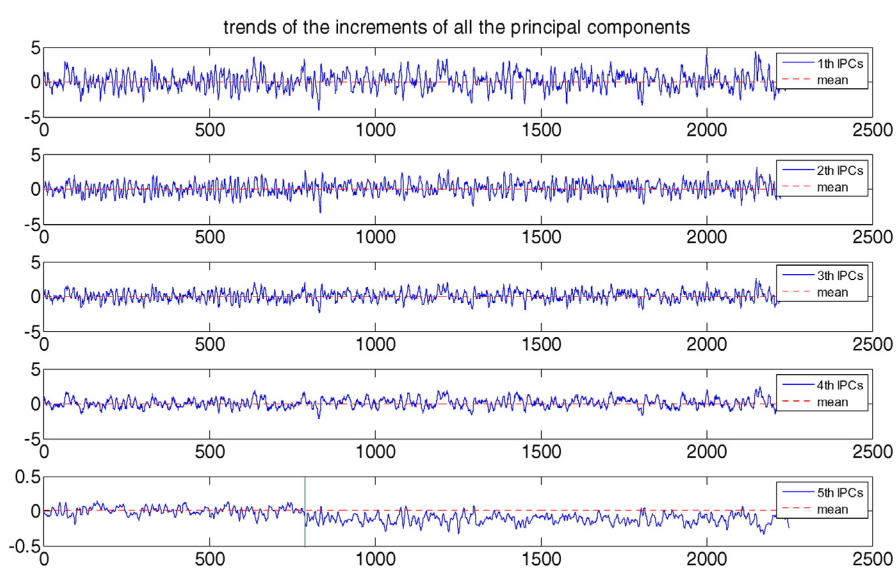
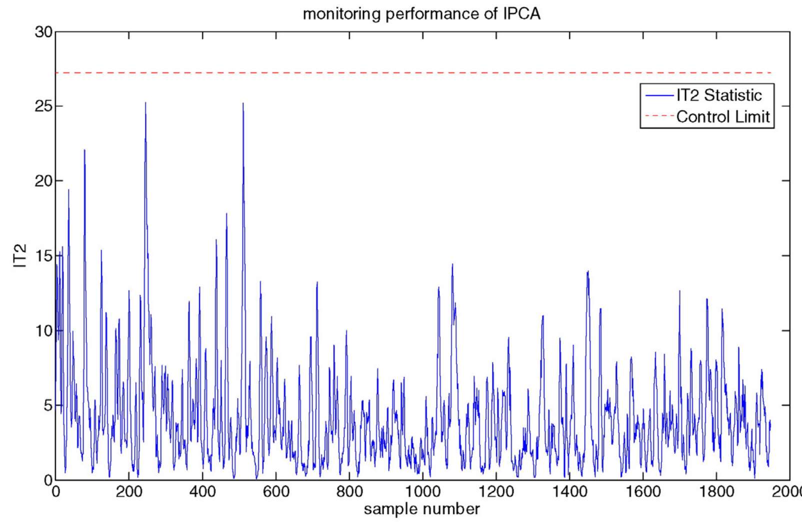
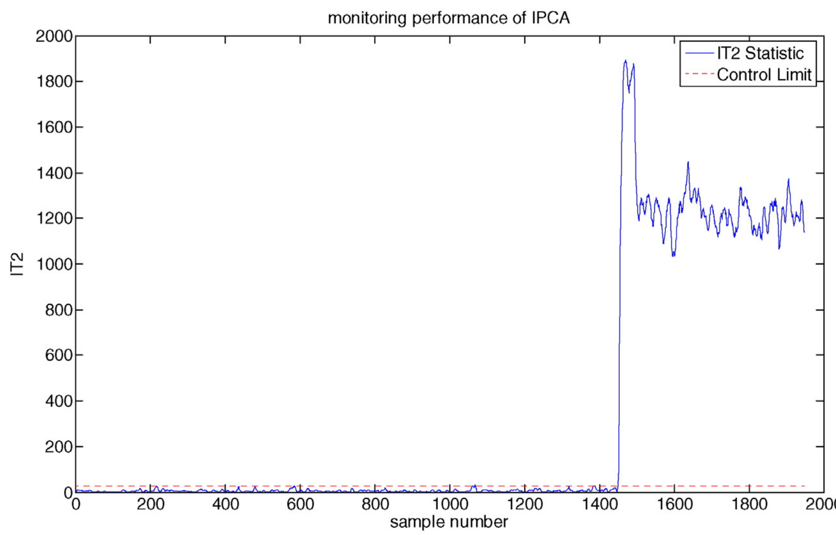


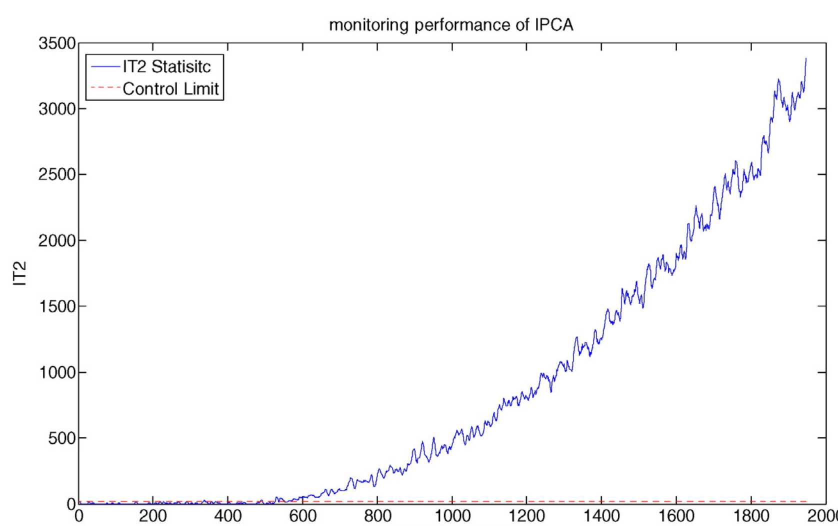
Fig. 5. Trends of the increments of all the PCs for Fault 2.



(a)



(b)



(c)

Fig. 6. Monitoring performance of IPCA for numeric process ((a) Normal, (b) Fault1, and (c) Fault2).

Table 1
Effects of L on monitoring performance.

L	False alarm rate (%)	Missing alarm rate (%)
2	0.000	88.933
4	0.154	38.267
6	0.000	5.733
8	0.257	6.067
10	56.186	6.800

Table 2
Effects of W on monitoring performance.

W	False Alarm rate (%)	Missing Alarm rate (%)
20	1.677	88.067
30	1.430	9.400
40	0.000	5.733
50	1.032	6.667
60	1.608	15.20

the monitoring model. Therefore, for the normal process, the fault alarm rate is kept distinctly low. This method can provide timely and accurate alarms for the step change fault and slow ramp fault.

In this method, the effects of parameters L and W are reflected in Table 1, in which $W = 40$, and Table 2, in which $L = 6$.

Table 1 shows that when parameter L increases, the missing alarm rate decreases quickly first and then becomes smooth and steady. However, when L is excessively large, the false alarm rate increases rapidly, and thus, parameter L should be less than a certain value according to the experiment. Table 2 shows that parameter W affects the missing alarm rate significantly. In this method, parameters L and W can be determined through some experiments.

5. Case study on continuous stirred tank reactor (CSTR)

In this section, a simulation of the non-isothermal CSTR process is given. The non-isothermal CSTR process is first proposed in [24] and further studied in process monitoring [25,26]. This is a first-order reaction. The process flow chart is shown in Fig. 7.

In this process, reactant Pure A is premixed with solvent flow, thereby producing B. This reaction is carried out under the following assumptions: perfect mixing, constant physical properties, and negligible shaft work. The dynamic characteristics of this reaction can be described by the material balance and energy balance equation as shown in Eqs. (12) and (13):

$$\frac{dC_a}{dt} = \frac{F}{V}C_{a0} - \frac{F}{V}C_a - k_0 e^{-\frac{E}{RT}C_a} \tag{12}$$

$$V\rho C_p \frac{dT}{dt} = \rho C_p F(T_0 - T) - \frac{a_2 a F_c^{b+1}}{F_c + a_2 a F_c^b / 2\rho_c C_{pc}}(T - T_C) + (-\Delta H_{r\times n}) V a_1 k_0 e^{-\frac{E}{RT}C_a} \quad , \tag{13}$$

where $C_{a0} = (C_{aa}F_a + C_{as}F_s)/(F_a + F_s)$, and the empirical formula between heat transfer coefficient and the flow of cooling water can be given as $UA = aF_c^b$. The impure reactants and dirty cooling silk can be described by the addition of a random coefficient to the reaction constant k and heat transfer coefficient UA, which are $k = a_1 k_0 e^{-\frac{E}{RT}}$ and $UA = a_2 a F_c^b$. The flows of reactant A and cooling water control the outlet concentration and outlet temperature, respectively. However, in this simulation, only the outlet temperature PI controller is active. Disturbances are also added to the inlet concentration (C_{aa} and C_{as}), inlet temperature T_0, flow of the solvent F_s and the cooling water temperature T_c. The disturbances are first-order processes:

$$\Delta x_t = \phi \Delta x_{t-1} + \sigma_e e_t \quad , \tag{14}$$

where σ_e is the standard variance, $e_t \sim N(0,1)$, and ϕ is the regression coefficient. The measurements of the process variables can be described as $x(t) = x^0(t) + \sigma_m e(t)$, where σ_m is the standard variance of the measured noise.

In this simulation, the nine process variables selected for monitoring are listed in Table 3.

In this study, the sample interval is 1 min. The time variations caused by the deactivation of the catalyst is simulated through the adjustment of the pre-exponential factor k_0 as $k_0 = (1 - t^* 10^{-4})^* k_{0initial}$. Moreover, the normal process and two types of faults are simulated, and 2500 samples are collected for each scene. The two types of faults are shown as follows:

Fault 1: A step change of 2.0 °C is applied to the outlet temperature T_0 from the 2001th sample.
Fault 2: The outlet temperature T_0 increases with 0.005 °C increments from the 1001th sample.

Fig. 8 shows the trends of the nine monitored variables in the normal process.

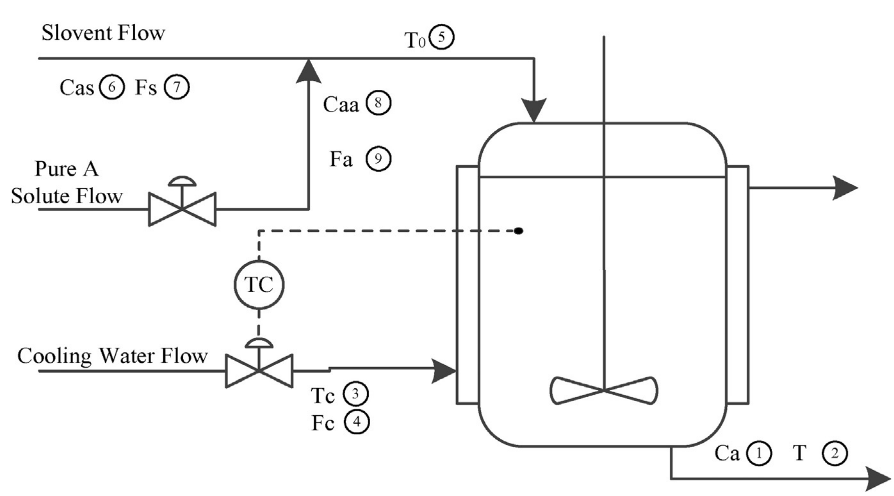


Fig. 7. Flow chart of the non-isothermal CSTR process.

Table 3
Monitored process variables of non-isothermal CSTR.

Variable	Type	Description
T	Output	Outlet temperature
C_a	Output	Outlet concentration
F_c	Manipulated variable	Cooling water flow
F_a	Input	Pure A flow
F_s	Input	Solvent flow
T_c	Input	Cooling water temperature
T_0	Input	Inlet temperature
C_{aa}	Input	Concentration of pure A
C_{as}	Input	Concentration of solvent

For the three scenes, 500 normal samples are used to build the conventional PCA model, obtain the IPCA model, and determine the control limit with 99% confidence level. Set parameter $L=6$ and $W=40$. The monitoring performances of the IPCA method for the three scenes are shown in Fig. 9.

For Fault 1, the fault was introduced from the 2001th sample. A total of 500 samples are used to build the model and (2^*L+W) samples are used to calculate the IPCs in the online monitoring stage. The fault occurs at the 1448th monitored sample. Similarly, for Fault 2, the fault should occur at the 448th monitored sample. As shown in Fig. 9, for the normal time-varying scene, the IPCA method yields a low fault alarm rate, whereas for Fault 1, this method gives off fault alarms on time. An alarm is generated from the 697th monitored sample; however, this method detects Fault 2 effectively because it is very small in the initial period and is not sufficient to generate the alarm. The superiority of the IPCA method is verified by comparing it with the moving window PCA (MWPCA) method, which is commonly used to monitor the time-varying process, is compared. In the MWPCA method, the window size is set as 800, the CPV is set as 85%. The confidence levels of T^2 and *SPE* statistics are set as 99%. The monitoring results of the MWPCA method for the three scenes are shown in Fig. 10.

Figs. 9 and 10 show that for the normal scene, the IPCA method can provide desirable monitoring performance without updating the monitoring model because the time-varying information is added to the monitoring model. MWPCA method can also provide good monitoring performance by updating the monitoring model online. For Fault 1, the step change fault, the IPCA method can yield a high fault detection rate. In the MWPCA method, the T^2 statistic provide good monitoring performance, and the *SPE* statistic yields a high missing alarm rate. This finding can be attributed to the step change that can only affect the PCs in the PC space. The effect is weak for PCs in the residual space. Finally, for Fault 2, the slow ramp fault, the IPCA method can detect this kind of fault effectively without updating the monitoring model, as shown in Fig. 9(c). However, when the newly detected sample is judged as a normal one in the MWPCA method, the monitoring model is updated blindly, and thus, the model can easily track the slow ramp fault. Fig. 10(c) shows that Fault 2 cannot be detected because the monitoring model tracks the slow ramp fault. In conclusion, the IPCA method is effective in distinguishing the slow ramp fault from the normal time-varying process.

6. Application on acetylene hydrogenation reactor

In this section, the IPCA method is used to monitor an acetylene hydrogenation reactor. In the ethylene production process, most of the acetylene is removed through catalytic hydrogenation in the reactor, and the concentration of the reactor outlet acetylene must be maintained below the set impurity level. Therefore, a steady acetylene hydrogenation system is crucial in guaranteeing quality ethylene product. The process flow chart is shown in Fig. 11.

The key process equipment includes an acetylene catalytic hydrogenation reactor (DC401A/B/C) and a green oil tower (FA401). One reactor is waiting on standby, and the two other reactors are connected in series, marked as the first-level and second-level reactors, respectively. Material from the top of the deethanizer is mixed and condensed with H_2, and then flows into the first-level reactor and carries out the catalyst hydrogenation. The product passes through the water cooler and the green oil tower, which removes the green oil. More H_2 is added before the material flows into the second-level reactor. The reactions of the acetylene hydrogenation process are as follows:

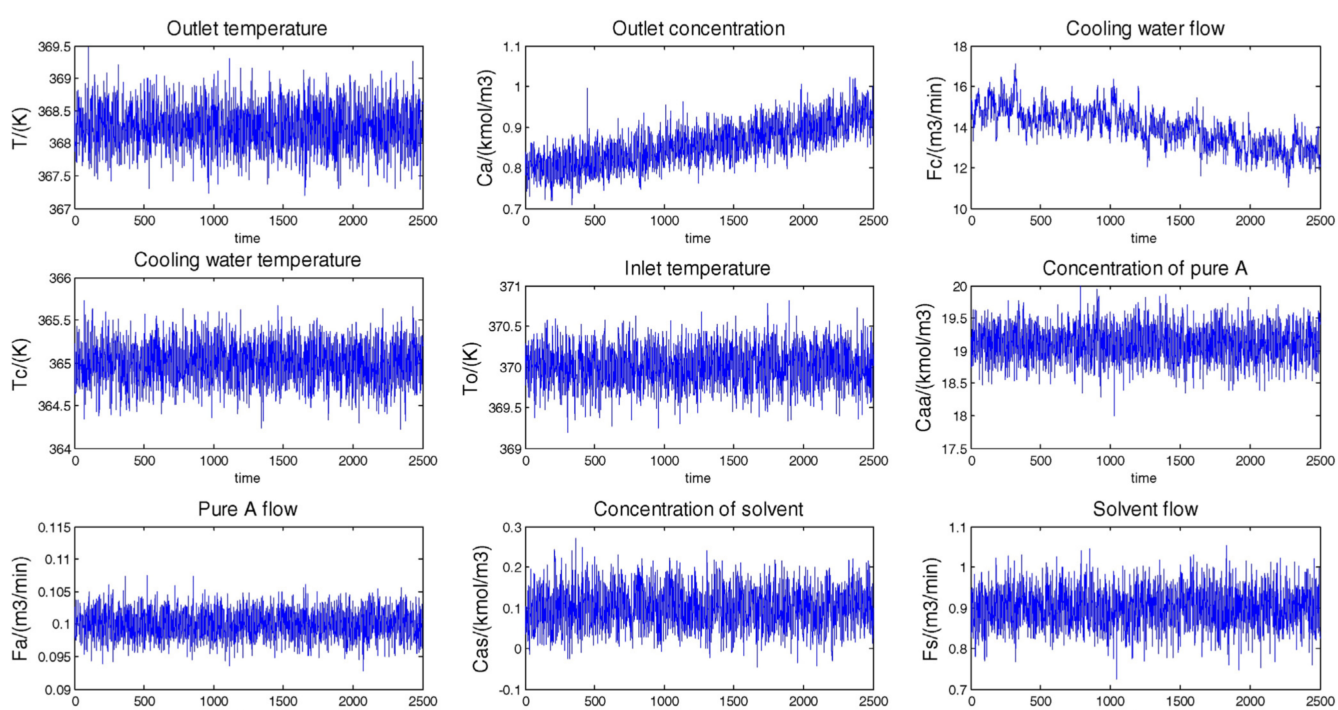


Fig. 8. Trends of monitored variables in the normal process.

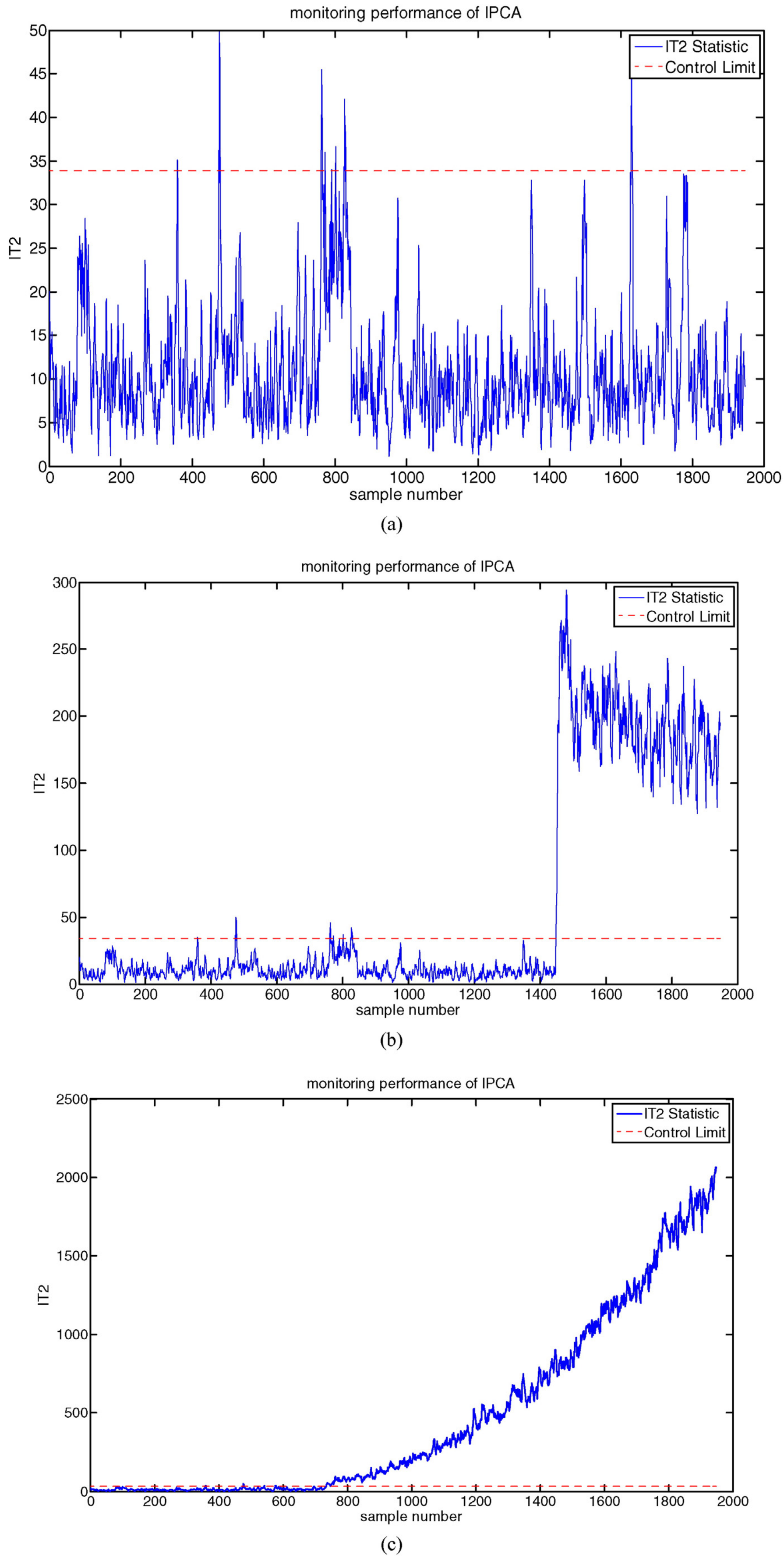


Fig. 9. Monitoring performance of IPCA for CSTR process ((a) Normal, (b) Fault 1, and (c) Fault 2).

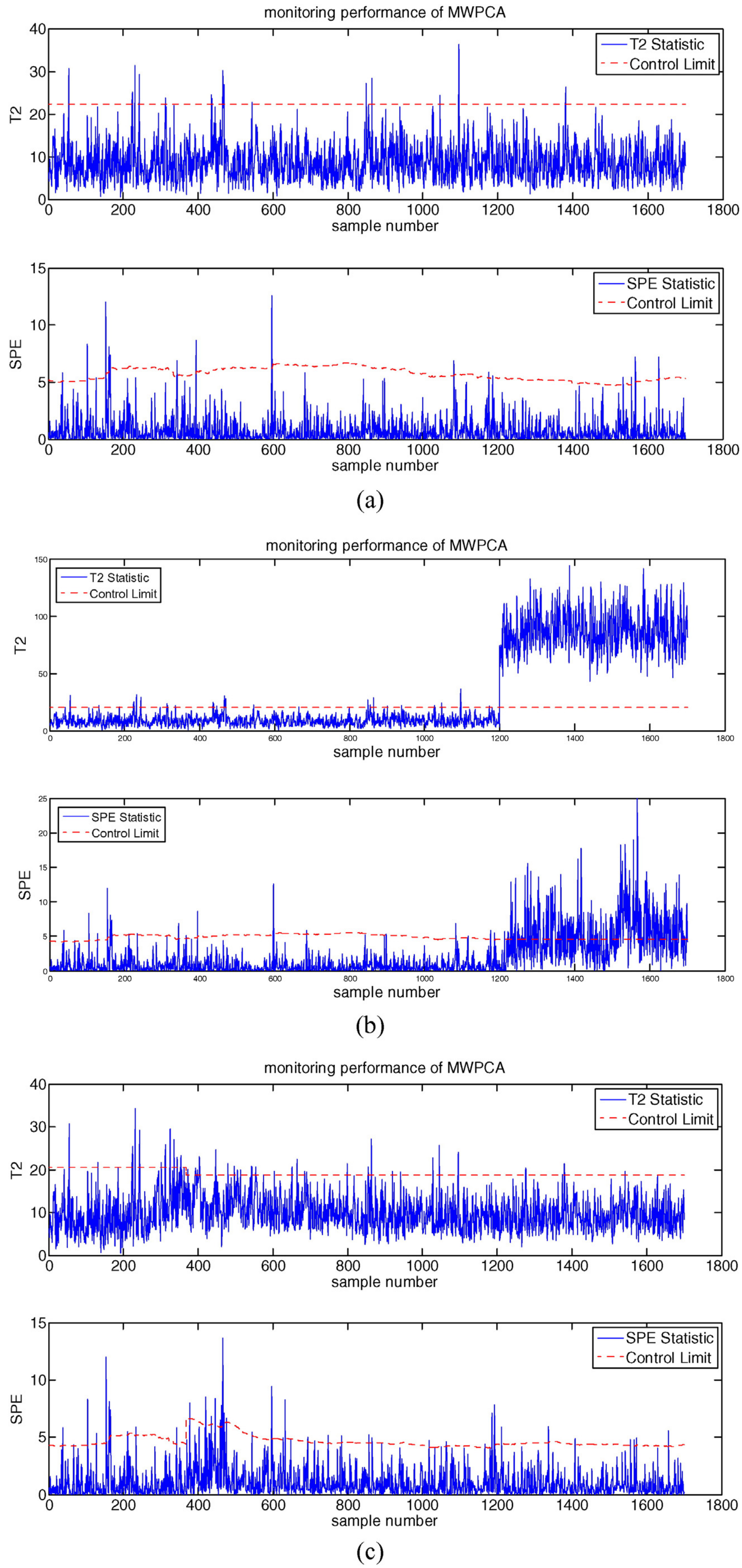


Fig. 10. Monitoring performance of MWPCA for CSTR process ((a) Normal (b) Fault1 (c) Fault2).

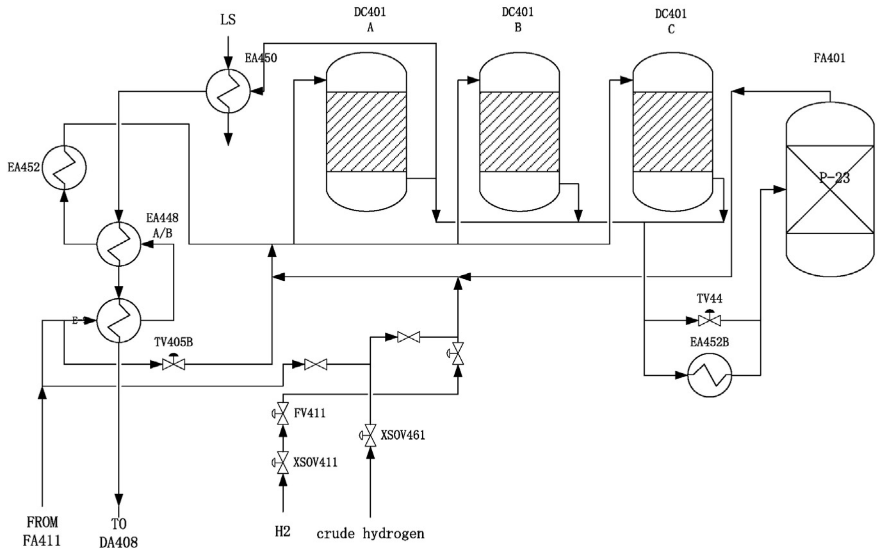


Fig. 11. Flow chart of acetylene hydrogenation process.

Main reaction:

$$C_2H_2 + H_2 \rightarrow C_2H_4 + 174.3kJ/mol \tag{15}$$

Side reaction:

$$C_2H_2 + 2H_2 \rightarrow C_2H_6 + 311.0kJ/mol \tag{16}$$

$$C_2H_4 + H_2 \rightarrow C_2H_6 + 136.7kJ/mol \tag{17}$$

$$mC_2H_2 + nC_2H_4 \rightarrow Green\ \ oil \tag{18}$$

In the practical procedure, the catalyst of the acetylene hydrogenation is deactivated on a daily basis. The acetylene catalytic hydrogenation process is a time-varying process.

A mechanism model of the first-level reactor is built to simulate the practical process in the chemical plant. The model describes the mass and energy balances in the reactor through a group of nonlinear ordinary differential equations coupled with the kinetic model as shown in Eqs. (19) and (20):

$$\frac{dF_i}{dz} = \sum \rho_B \times r_j \times S \tag{19}$$

$$\frac{dT}{dz} = \frac{-S \times \rho_B \times \sum \Delta H_i \times r_i}{\sum F_i \times C_{p,kmol,i}}, \tag{20}$$

where $C_{p,kmol,i}$ is the heat capacity of the i_{th} gas component; F_i is the flow of the i_{th} gas component; ΔH_i is the reaction heat of the i_{th} reaction; r_i is the rate of the i_{th} reaction; T is the reactor bed temperature; z is the reactor length; ρ_B is the density of the catalyst; and S is the reactor cross-sectional area. The kinetic model [27] in this study can be denoted as follows:

$$r_{C_2H_2} = \frac{k_3 p_{C_2H_2} p_{H_2}}{1 + K^A_{C_2H_2} p_{C_2H_2}} Da_3 + \frac{k_4 p_{C_2H_2} p_{H_2}}{\left(1 + K^A_{C_2H_2} p_{C_2H_2}\right)^2} Da_4 + \frac{k_1 p_{C_2H_2} p^2_{H_2}}{\left(1 + K^A_{C_2H_2} p_{C_2H_2}\right)^3} Da_1 \tag{21}$$

$$r_{C_2H6} = \frac{k_2 p_{C_2H4} p_{H_2}}{\left(1 + K^E_{C_2H_2} p_{C_2H_2} + K^E_{C_2H_4} p_{C_2H_4}\right)^3} Da_2 + \frac{k_1 p_{C_2H_2} p^2_{H_2}}{\left(1 + K^A_{C_2H_2} p_{C_2H_2}\right)^3} Da_1 \tag{22}$$

Table 4
Parameters of the kinetic model.

Parameter	Value
A_1	0.8941470e + 004
A_2	1.9047503e + 004
A_3	1.5998785e + 004
A_4	1.2607347e + 004
E_1	7.5850983e + 001
E_2	4.0000000e + 001
E_3	4.3172498e + 004
E_4	1.3908516e + 004
$K^A_{C_2H_2}$	2.4962464e + 002
$K^E_{C_2H_2}$	1.0980532e + 003
$K^E_{C_2H_4}$	1.0566711e + 000
n_1	4.9121400e + 000
n_2	2.1877288e + 000
n_3	1.0000616e + 000
n_4	1.0000085e + 000
ka_1	6.1431896e + 004
ka_2	0.0000000e + 000
ka_3	2.0000000e + 005
ka_4	2.0000000e + 005
Ea_1	1.7549850e + 005
Ea_2	9.4054132e + 004
Ea_3	5.2620097e + 004
Ea_4	4.9808363e + 004

Table 5
Monitored process variables of acetylene hydrogenation reactor.

Variable	Type	Description
$F_{C_2H_6,in}$	Input	Inlet C2H6 flow
$F_{C_2H_4,in}$	Input	Inlet C2H4 flow
$F_{C_2H_2,in}$	Input	Inlet C2H2 flow
$F_{H_2,in}$	Input	Inlet H2 flow
T_{in}	Input	Inlet temperature
$F_{C_2H_6,out}$	Output	Outlet C2H6 flow
$F_{C_2H_4,out}$	Output	Outlet C2H4 flow
$F_{C_2H_2,out}$	Output	Outlet C2H2 flow
$F_{H_2,out}$	Output	Outlet H2 flow

$$k_i = A_i * e^{(-E_i/(RT))}(i = 1, 2, 3, 4) \tag{23}$$

$$Da_i = \frac{1}{(1 + ka_i * (n_i - 1) * e^{(-Ea_i/8.314/T)} * Day)^{(1/(n_i - 1))}}, \tag{24}$$

where R is the perfect gas constant, T is the reactor temperature, and $p_{C_2H_2}$, $p_{C_2H_4}$, and $p_{H_2}$are the C_2H_2, C_2H_4, and H_2 partial pressures, respectively. Da_i reflects the deactivated rate of the catalyst. In obtaining the kinetic model parameters, the genetic algorithm is employed to optimize the parameters and minimize the errors between the predicted values and the practical measurement values of the model. The optimized parameters are listed in Table 4.

The mechanism model is used to simulate the first level reactor. In the model, reactor length is 3.16 m, reactor cross-sectional radius is 1.65 m, and catalyst density is 700 kg/m^3. Moreover, the reactor has already been running for 10 days. White noise is added to the model inputs. The sample interval is 5 min. A total of 1000 samples are generated by this model, and nine process variables are sampled to monitor the first-level reactor. The process variables are listed in Table 5.

In this study, two types of faults are added as follows:

Fault 1: A step change of 5.0 °C is added to the inlet temperature starting from the 800th sample.

Fault 2: The inlet C2H4 flow increases by 0.03 kmol/h starting from the 800th sample.

The trends of the nine process variables under the normal process are shown in Fig. 12.

For the three scenes, 400 normal samples are used to build the conventional PCA model, obtain the IPCA model, and determine the control limit with 99% confidence level. Parameters are set as follows: $L = 5$ and $W = 30$. The monitoring results of the IPCA method for the three scenes are shown in Fig. 13.

As shown in Fig. 13, for Fault 1, the fault introduced from the 801th sample, 400 samples are used to build the model, and $(2^*L + W)$ samples are used to calculate the IPCs in the online monitoring stage. The fault occurs at the 361th monitored sample. Similarly, Fault 2 should occur at the 361th monitored sample. An alarm is generated from the 380th monitored sample because Fault 2 is a very small fault occurring in the initial period and is not sufficient to generate the alarm. The MWPCA method is also compared with the ICPA. In the MWPCA method, the window size is set as 400, the CPV is as 85%, and the confidence levels of the T^2 and SPE statistics are set as 99%. The monitoring curves of the MWPCA method are shown in Fig. 14.

Figs. 13 and 14 show that for the normal scene, the IPCA method can provide desirable monitoring performance without updating the monitoring model by adding the time-varying information to the monitoring model, which is almost the same as the MWPCA method updating the monitoring model online. For Fault 1, the step change fault, the IPCA method can yield high fault detection rate. For the MWPCA method, the T^2 statistic also provides good monitoring results. For the slow ramp fault (Fault 2), as shown in Fig. 13(c), the IPCA method can detect this fault successfully without updating the monitoring model. However, the MWPCA method updates the monitoring model blindly when the newly detected sample is adjudged as a normal one. As shown in Fig. 14(c), because the monitoring model tracks the slow ramp fault, Fault 2 cannot be detected obviously. In conclusion, the IPCA method is effective for the time-varying process.

7. Conclusion

In this study, IPCA is proposed to distinguish the slow ramp fault from a normal time-variant chemical process. In contrast with conventional adaptive time-variant process monitoring methods, such as recursive method, moving window strategy, and others that require constant updating of the monitoring model and controlling the limit online blindly, the IPCA method builds the monitoring model using the time-varying information provided by IPCs and keeps the monitoring model changeless. This method can distinguish the slow ramp fault from the normal time-varying process. Finally, a case study on a numeric system, the non-isothermal CSTR process and the application on the acetylene hydrogenation reactor, demonstrated the effectivity of this method.

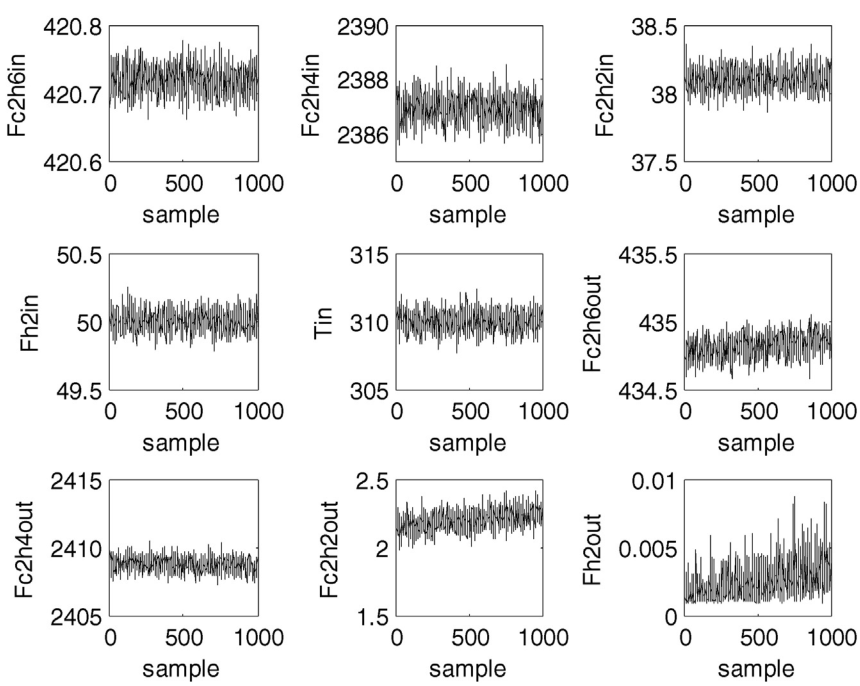


Fig. 12. Trends of nine process variables in the normal process of acetylene hydrogenation reactor.

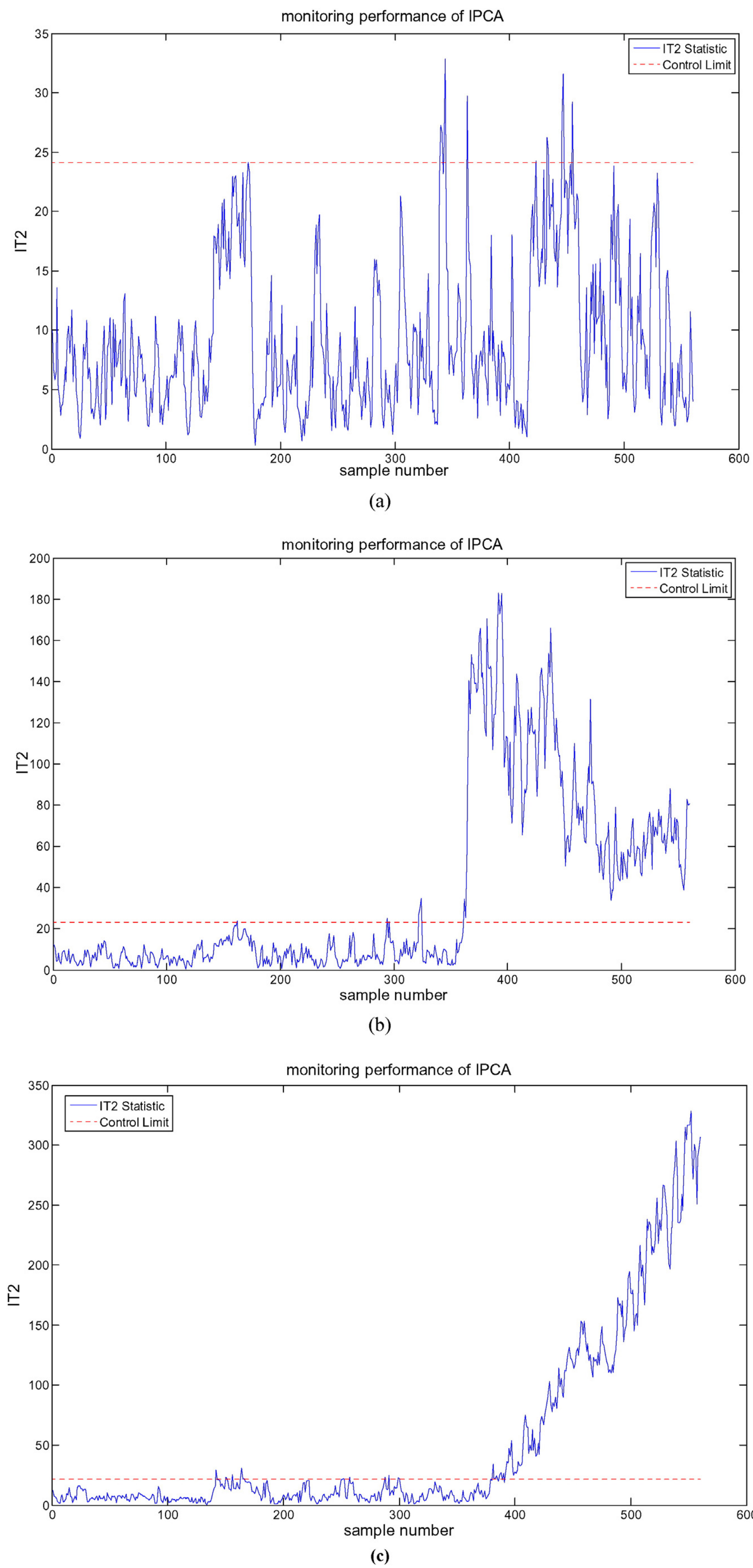


Fig. 13. Monitoring performance of IPCA for acetylene hydrogenation process ((a) Normal, (b) Fault1, and (c) Fault2).

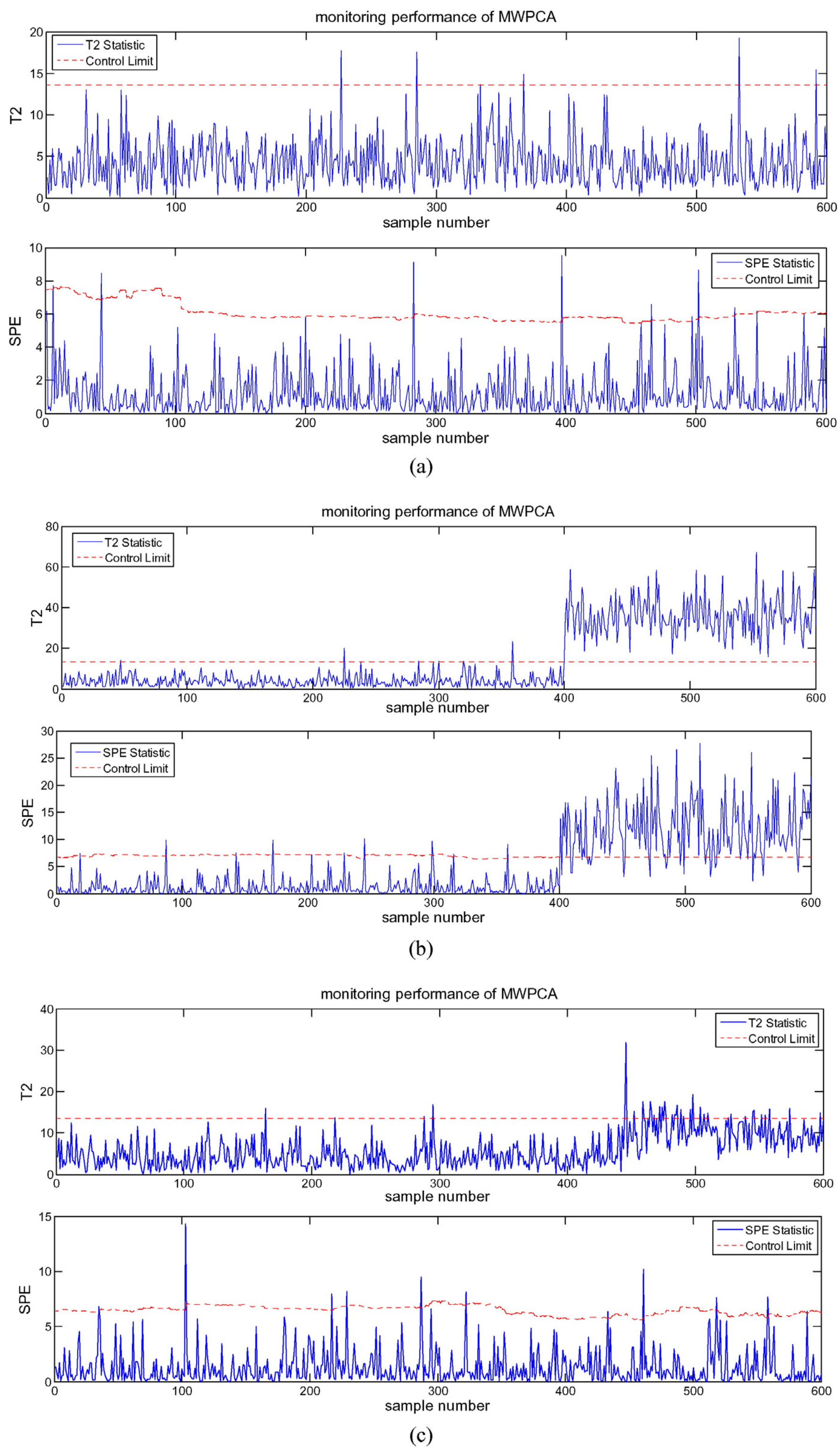


Fig. 14. Monitoring performance of MWPCA for acetylene hydrogenation process ((a) Normal, (b) Fault 1, and (c) Fault 2).

Acknowledgement

This work is supported by the Major State Basic Research Development Program of China (2012CB720500), National Natural Science Foundation of China (61174118), Shanghai Natural Science Foundation (14ZR1421800, 16ZR1407300) and the State Key Laboratory of Synthetical Automation for Process Industries (PAL-N201404).

References

[1] C. Zhao, F. Gao, Fault-relevant principal component analysis (Fpca) method for multivariate statistical modeling and process Monitoring, Chemom. Intell. Lab. Syst. 133 (2014) 1–16.

[2] J. Yu, A particle filter driven dynamic Gaussian mixture model approach for complex process monitoring and fault diagnosis, J. Process Control 22 (2012) 778–788.

[3] S. Stubbs, J. Zhang, J. Morris, Fault detection in dynamic processes using a simplified monitoring-specific CVA state space modelling approach, Comput. Chem. Eng. 41 (2012) 77–87.

[4] Y. Zhang, S. Li, Z. Hu, Improved multi-scale kernel principal component analysis and its application for fault detection, Chem. Eng. Res. Des. 90 (2012) 1271–1280.
[5] M.M. Rashid, J. Yu, A new dissimilarity method integrating multidimensional mutual information and independent component analysis for non-Gaussian dynamic process monitoring, Chemom. Intell. Lab. Syst. 115 (2012) 44–58.
[6] Y. Zhang, J. An, Z. Li, et al., Modeling and monitoring for handling nonlinear dynamic processes, Inf. Sci. 235 (2013) 97–105.
[7] W. Li, H.H. Yue, S. Valle-Cervantes, et al., Recursive PCA for adaptive process monitoring, J. Process Control 10 (2000) 471–486.
[8] L.M. Elshenawy, H.A. Awad, Recursive fault detection and isolation approaches of time-varying processes, Ind. Eng. Chem. Res. 51 (2012) 9812–9824.
[9] X. Wang, U. Kruger, B. Lennox, Recursive partial least squares algorithms for monitoring complex industrial processes, Control. Eng. Pract. 11 (2003) 613–632.
[10] Y. Zhang, S. Li, Y. Teng, Dynamic processes monitoring using recursive kernel principal component analysis, Chem. Eng. Sci. 72 (2012) 78–86.
[11] S. Wold, Exponentially weighted moving principal components analysis and projections to latent structures, Chemom. Intell. Lab. Syst. 23 (1994) 149–161.
[12] S. Lane, E.B. Martin, A.J. Morris, P. Gower, Application of exponentially weighted principal component analysis for the monitoring of a polymer film manufacturing process, Trans. Inst. Meas. Control. 25 (2003) 17–35.
[13] S.J. Qin, Recursive PLS algorithms for adaptive data modeling, Comput. Chem. Eng. 22 (1998) 503–514.
[14] X. Wang, U. Kruger, G.W. Irwin, Process monitoring approach using fast moving window PCA, Ind. Eng. Chem. Res. 44 (2005) 5691–5702.
[15] X.B. He, Y.P. Yang, Variable MWPCA for adaptive process monitoring, Ind. Eng. Chem. Res. 47 (2007) 419–427.
[16] Y. Ma, H. Shi, H. Ma, et al., Dynamic process monitoring using adaptive local outlier factor, Chemom. Intell. Lab. Syst. 127 (2013) 89–101.
[17] Y. Ma, H. Shi, M. Wang, Adaptive local outlier probability for dynamic process monitoring, Chin. J. Chem. Eng. 22 (2014) 820–827.
[18] X. Xie, H. Shi, Dynamic multimode process modeling and monitoring using adaptive Gaussian mixture models, Ind. Eng. Chem. Res. 51 (2012) 5497–5505.
[19] Z. Ge, Z. Song, Online monitoring of nonlinear multiple mode processes based on adaptive local model approach, Control. Eng. Pract. 16 (2008) 1427–1437.
[20] D.W. Scott, Multivariate Density Estimation, Wiley Online Library, New York, 1992.
[21] A.R. Webb, K.D. Copsey, G. Cawley, Statistical Pattern Recognition, Wiley, New York, 2011.
[22] W. Ku, R.H. Storer, C. Georgakis, Disturbance detection and isolation by dynamic principal component analysis, Chemom. Intell. Lab. Syst. 30 (1995) 179–196.
[23] J. Lee, C. Yoo, I. Lee, Statistical process monitoring with independent component analysis, J. Process Control 14 (2004) 467–485.
[24] S. Yoon, J.F. MacGregor, Fault diagnosis with multivariate statistical models part I: using steady state fault signatures, J. Process Control 11 (2001) 387–400.
[25] S.W. Choi, E.B. Martin, A.J. Morris, et al., Fault detection based on a maximum-likelihood principal component analysis (PCA) mixture, Ind. Eng. Chem. Res. 44 (2005) 2316–2327.
[26] C.F. Alcala, S.J. Qin, Reconstruction-based contribution for process monitoring with kernel principal component analysis, Ind. Eng. Chem. Res. 49 (2010) 7849–7857.
[27] A. Borodziński, A. Cybulski, The kinetic model of hydrogenation of acetylene-ethylene mixtures over palladium surface covered by carbonaceous deposits, Appl. Catal. A Gen. 198 (2000) 51–66.