Batch Process Monitoring and Fault Diagnosis Based on Multi-Time-Scale Dynamic PCA Models

Yuan Yao* and Furong Gao**

** Dept. of Chemical and Biomolecular Engineering, Hong Kong University of Science and Technology,*
Clear water bay, Kowloon, Hong Kong SAR, P. R. China,
*** Dept. of Chemical and Biomolecular Engineering, Hong Kong University of Science and Technology,*
Clear water bay, Kowloon, Hong Kong SAR, P. R. China
(Tel: +852-2358-7139, Fax:+852-2358-0054, e-mail: kefgao@ust.hk).

Abstract: Dynamics are inherent characteristics of batch processes, which can be divided into short time-scale dynamics within a batch duration and long time-scale dynamics across several batches. The interactions between process variables make different types of dynamics confounded. Under such situations, it is difficult to perform efficient fault diagnosis. In this paper, a batch process monitoring scheme is proposed to separate different types of process variations for modeling and perform monitoring and fault diagnosis with multi-time-scale dynamic principal component analysis (PCA) models. Simulation results show that the fault diagnosis efficiency is enhanced.

Keywords: batch process, monitoring, fault diagnosis, principal component analysis, dynamics.

1. INTRODUCTION

In today's industrial manufacturing, batch processes are widely applied to manufacture high-value-added products. To ensure operation safety and product quality, the multivariate statistical monitoring methods, such as multiway principal component analysis (MPCA) (Nomikos and MacGregor, 1994; Nomikos and MacGregor, 1995) which is an extension of principal component analysis (PCA), have been utilized in batch process monitoring and fault diagnosis.

Dynamics are inherent characteristics of batch processes, including short time-scale dynamics within a batch duration and long time-scale dynamics across several batches. Different types of batch dynamics are usually caused by different values of variable response time which measures the time process variables take to react to given inputs. Fast-response variables have small response time constant, while slow-response variables have large values which may be longer than a batch duration. To model batch process dynamics better, several multivariate statistical monitoring methods have been proposed. Batch dynamic principal component analysis (BDPCA) (Chen and Liu, 2002) captures within-batch dynamic information, while two-dimensional dynamic principal component analysis (2-D-DPCA) (Lu, et al., 2005) can model both long and short time-scale dynamics in a two-dimensional (2-D) model structure.

In batch processes, variable correlations always exist. Especially, changes in slow-response variables can also affect fast-response variable trajectories. This makes different types of variable dynamics confounded, and causes difficulties in process fault diagnosis, as shown later. Therefore, it is desirable to have a method which can decouple process variation information according to dynamic time scales and monitor different types of variations separately. Thus, the fault diagnosis efficiency and accuracy can be enhanced.

Several existing multivariate statistical methods can divide process variations into blocks, scales or levels, but none of them can be utilized directly to handle the situation mentioned above. Multiblock PCA or partial least squares (PLS) methods (Westerhuis et al., 1998) group process variables into meaningful blocks and concern both the inner relationship within each block and the inter relationship among blocks. Although the variables with different response time can be divided into different blocks, two kinds of dynamics information are not separated due to variable correlations. Multiscale PCA (Bakshi, 1998) makes use of wavelet analysis techniques to transform each variable signal from time domain to frequency domain, and performs PCA on wavelet coefficients at each scale. However, the different dynamics characteristics of each variable are not taken into consideration. Multilevel component analysis (MLCA) and multilevel simultaneous component analysis (MLSCA) (Timmerman, 2006) separate within-batch variations and between-batch variations. But only the static variations are extracted, while process dynamics are not modeled. Besides, none of the methods reviewed in this paragraph can deal with long time-scale dynamics across several batches.

In this paper, a batch process monitoring scheme is developed. This scheme makes use of variable response time information which can be easily achieved, and separate process variations into different levels corresponding to dynamics time scales. 2-D-DPCA method is adopted to build multi-time-scale models. Thus, faults occurring to a certain level can be accordingly detected with the level model. Then, diagnosis can also be performed in the corresponding level, indicating the causing of the fault more clearly.

The article is organized as following. In section 2, the 2-D-DPCA method is reviewed. Then, a multi-time-scale batch process monitoring scheme is proposed and described detail in section 3. Simulation results are given in section 4. A batch process with both long time-scale and short time-scale dynamics is simulated to compare the monitoring and diagnosis efficiencies between the conventional 2-D-DPCA method and the proposed scheme. Finally, a conclusion is given in section 5 to summarize the paper.

2. TWO-DIMENSIONAL DYNAMIC PCA (2-D-DPCA)

2-D-DPCA method proposed by the authors can model both long and short time-scale batch process dynamics with a parsimonious two-dimensional (2-D) time series model structure together with PCA technique (Lu, et al., 2005).

Process dynamics can be indicated by the correlations between current measurements and lagged measurements. Long time-scale dynamics often behave as a kind of two-dimensional (2-D) dynamics, which means the current measurements are dependent not only on lagged measurements in the past time direction in the same batch, but also on lagged measurements in some past batches. These lagged variables form a region called the support region or the region of support (ROS). In 2-D-DPCA, an expanded data matrix $\tilde{\mathbf{X}}$ is formed by including all the lagged measurements in ROS, together with current measurements. For more details about ROS determination, please refer to Yao et al.'s work (2008).

Suppose $\tilde{\mathbf{X}}$ has been normalized to have unit variances and zero means. PCA algorithm is performed on it:

$$\tilde{\mathbf{X}} = TP^T + E \ . \qquad (1)$$

where T and P are score matrix and loading matrix respectively, and E is the residual matrix. The number of scores retained in the score space can be determined using cross-validation (Wold, 1978). Thus, the original process data are divided into two subspaces. Score space extracts systematic variation information, including both 2-D dynamics and cross-correlation information among variables, while normal distributed noises are retained in residual space. Therefore, *SPE* statistic and corresponding control limits can be calculated for process monitoring in residual space. After a fault is detected by the *SPE* control plot, contribution plots with control limits (Westerhuis et al., 2000) are used in fault diagnosis to find the causes of the faults.

When a batch process only has short time-scale dynamics, its ROS is selected as a region containing several steps of lagged measurements in current batch. In such a case, 2-D-DPCA model is similar to BDPCA model (Chen and Liu, 2002).

3. MULTI-TIME-SCALE MONITORING SCHEME

3.1 Motivations

As mentioned in introduction section, in batch processes, fast-response variable trajectories are often affected by disturbances in slow-response variables. Take injection molding process as an example. In that process, temperature variables' response time constants are often longer than a batch duration, while pressure variables response fast. Suppose a disturbance occurs to barrel temperature. It takes a long time for barrel temperature to recover. During this period, the material properties, such as viscosity and density, change gradually due to the temperature change. This further causes slow drifts in pressure variable trajectories, although pressures are fast-response variables. From this example, it can be seen that both short and long time-scale dynamics are confounded in fast-response variable trajectories.

As shown in the simulation example in section 4, such confounding leads to difficulties in fault diagnosis results. Therefore, it is desirable to decouple process variation information into several levels according to dynamic time scales. Then, level models can be built and different types of variations can be monitored and diagnosed separately, so that the fault diagnosis efficiency and accuracy can be enhanced.

3.2 Variable classification

As a kind of external information, variable response time is easy to be estimated from process open-loop tests which are regular steps in controller designs. Such information is used to classify variables into groups. It is the first step of multi-time-scale modeling and monitoring.

In many cases, the variables can be simply divided into two groups. One contains fast-response variables, while the other contains slow-response variables which can cause long time-scale dynamics beyond a batch. In some other situations, it may be desired to further divide the above two groups into sub-groups. Suppose there are *M* number of variable divided into the fast-response variable group. Take each variable's response time constant as a pattern. The k-means clustering algorithm (Jain et al., 1999) is adopted for partitioning the *M* number of patterns. The final cluster number is determined automatically with a specified threshold of the minimal distance between two cluster centers or the maximal radius of a cluster. A larger threshold results in fewer variable groups; vice versa. The slow-response variable group can also be further divided in the same way. By doing so, the process variables with similar response time constants are clustered into the same group.

3.3 Multi-time-scale level separation

Without losing generality, first, suppose the process variables are divided into two groups. As discussed in section 3.1, two types of dynamics may confound in the trajectories of the variable in the fast-response variable group. To solve this problem, the operation data in this group should be decomposed into two parts: one part can be explained by the variable measurements in the slow-response variable group, and the other part can not be explained by them and only contains short time-scale dynamics. The level separation is based on the idea of external analysis, which was originally proposed by Takane and Shibayama (1991) and further discussed by Yoon and MacGregor (2001). Kano et al. (2004) made use of this idea to distinguish faults from normal changes in operating conditions.

Consider a batch process data matrix $\hat{X}(I \times J \times K)$, where I, J, K are the number of batches, variables and time intervals respectively. Unfold this three-way data matrix into a two-way matrix $X(IK \times J)$ by keeping the variable dimension and merging the other two dimensions. Suppose X have been normalized. After variable classification, X can be described as $X = [F \quad S]$, where F consists of J_F number of fast-response variables and S consists of $J_S = J - J_F$ number of slow-response variables. To decompose F, regression analysis is performed by regarding S and F as inputs and outputs respectively. If variables in S are independent of each other, the ordinary least square (OLS) regression can be used:

$$\Phi = (S^T S)^{-1} S^T F \, , \tag{2}$$

where Φ is the regression coefficient matrix. The significance of regression can be tested (Montgomery, 2005) to show whether there are correlations between S and F. If there is no correlation, the levels are naturally seperated. The short time-scale level consists of $D^S = F$, while the long time-scale level consists of $D^L = S$. Otherwise, calculate (3).

$$E = F - S\Phi \, , \tag{3}$$

where $S\Phi$ contains a part of information in F which is explained by slow-response variable, while the filtered data matrix E dose not contains long time-scale dynamics. When the slow-response variables are not independent, PLS or principal component regression (PCR) can be utilized to avoid the collinearity problem. Thus, the process variation information is separated into two levels according to different time scales of dynamics: $D^S = E$ and $D^L = [S\Phi \quad S]$.

When there are more than two groups, the time-scale level separation is performed in an iterative way. Unfolded data matrix X is described as $X = \begin{bmatrix} X_1^0 & X_2^0 & \cdots & X_C^0 \end{bmatrix}$, where X_i^j is the filtered data matrix of the ith variable group after the jth iteration run in time-scale level separation, consisting of J_i number of variables. When $j = 0$, X_i^j represents the data before performing iteration steps. C is the total number of variable groups, and the variables in X_i^j response faster than the variables in X_{i+1}^j. In the jth run, let $S^j = X_{C-j+1}^{j-1}$ and $F^j = \begin{bmatrix} X_1^{j-1} & X_2^{j-1} & \cdots & X_{C-j}^{j-1} \end{bmatrix}$. Φ^j is then calculated in the similar way as (2), and the data are filtered as

$$\begin{aligned} E^j &= F^j - S^j \Phi^j = \begin{bmatrix} X_1^{j-1} & X_2^{j-1} & \cdots & X_{C-j}^{j-1} \end{bmatrix} - X_{C-j+1}^{j-1} \Phi^j \\ &= \begin{bmatrix} X_1^j & X_2^j & \cdots & X_{C-j}^j \end{bmatrix} \end{aligned} \, . \tag{4}$$

After C-1 cycles of iteration, all levels are separated. The shortest time-scale level consists of $D^1 = E^{C\text{-}1}$. The second shortest time-scale level consists of $D^2 = [S^{C\text{-}1}\Phi^{C\text{-}1} \quad S^{C\text{-}1}]$. … The longest time-scale level consists of $D^C = [S^1\Phi^1 \quad S^1]$.

3.4 *Multi-time-scale dynamic PCA modeling, monitoring and fault diagnosis*

After level separation, 2-D-DPCA is adopted to construct level models for online monitoring and fault diagnosis.

Take a C level separation as an example. In level j (j>1), $D^j = [S^{C\text{-}j+1}\Phi^{C\text{-}j+1} \quad S^{C\text{-}j+1}]$. Since $S^{C-j+1}\Phi^{C-j+1}$ is completely dependent on S^{C-j+1}, it only represents redundant information in a process monitoring context. Therefore, the variation information in each level is reorganized as $G^1 = E^{C-1}$, $G^2 = S^{C-1}$, …, $G^C = S^1$ with matrix dimensions of ($IK \times J_1$), ($IK \times J_2$), …, ($IK \times J_C$) respectively. These matrices are rearranged into three-dimensional arrays with dimensions of ($I \times J_1 \times K$), ($I \times J_2 \times K$), …, ($I \times J_C \times K$). Then, following ordinary procedures, 2-D-DPCA models can be established for each level. The *SPE* control limits are calculated for online monitoring. For a level belonging to short time-scale dynamics, the 2-D-DPCA model reduces to a BDPCA model. For these levels, the T^2 control limits can also be calculated, since there is no batch-wise dynamics.

In online monitoring, the new data are firstly filtered based on (4) using coefficient matrices Φ^1, Φ^2, …, $\Phi^{C\text{-}1}$ in turns. Thus, the variations contained in the new data are separated into different time-scale levels. The corresponding 2-D-DPCA model is utilized to monitor each level. After faults are detected in some levels, the contribution plots can be used for fault diagnosis in these levels accordingly.

4. SIMULATION EXAMPLE

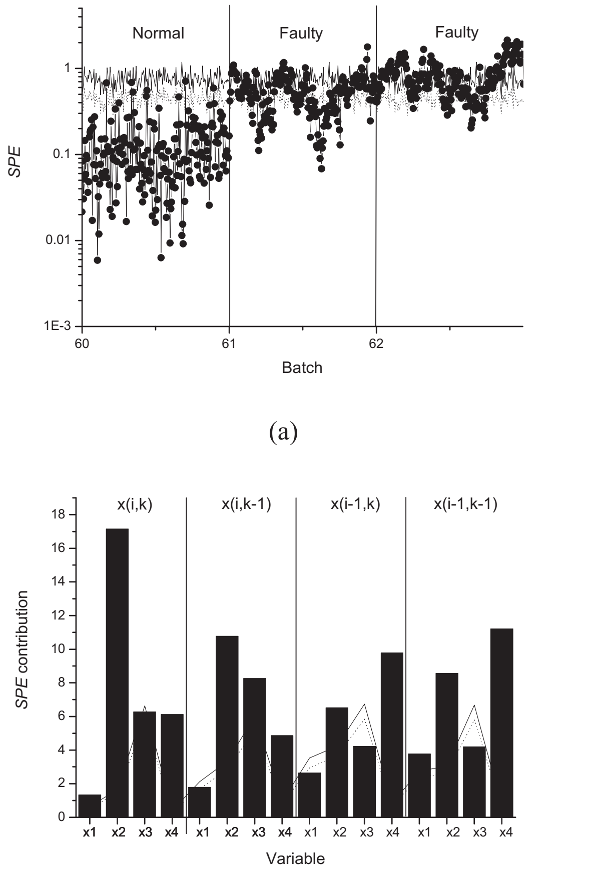
4.1 *Batch process modeling*

In this section, a simulated batch process with both long and short time-scale dynamics is utilized to compare the monitoring and fault diagnosis efficiency of the proposed multi-time-scale dynamic PCA models with the conventional 2-D-DPCA model. The process model is given as below,

$$\begin{aligned} x_1(i,k) &= 0.5 * x_1(i,k-1) + 0.8 * x_1(i-1,k) - 0.3 * x_1(i-1,k-1) \\ x_2(i,k) &= 0.44 * x_2(i-1,k) + 0.67 * x_2(i,k-1) - 0.11 * x_2(i-1,k-1) \, , \\ x_3(i,k) &= 0.4 * x_3(i,k-1) + 0.25 * x_1(i,k) + 0.35 * x_2(i,k) \\ x_4(i,k) &= 0.8 * x_4(i,k-1) + 0.53 * x_1(i,k) - 0.33 * x_2(i,k) \end{aligned} \tag{5}$$

where i is the batch index; k is the time index; x_1 and x_2 are two independent slow-response variables with long time-scale dynamics described in a 2-D structure; x_3 and x_4 are fast-response variables correlated to their own values at one step before in the current batch, which are also affected by x_1, x_2. Gaussian noises with variance 0.01 are added into the data.

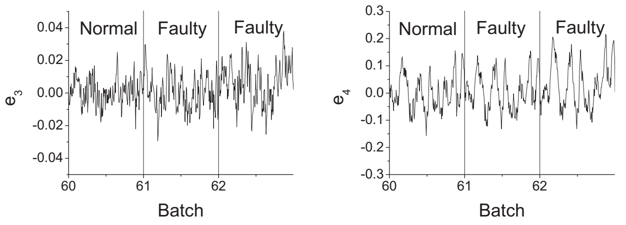
For conventional 2-D-DPCA modeling, the ROS is determined as $\mathbf{x}(i,k-1), \mathbf{x}(i-1,k), \mathbf{x}(i-1,k-1)$, where $\mathbf{x}(i,k-1) = \begin{bmatrix} x_1(i,k) & x_2(i,k) & x_3(i,k) & x_4(i,k) \end{bmatrix}$. So that, there are totally 16 variables in the augmented data matrix $\tilde{\mathbf{X}}$, including 4 current variables and 12 lagged variables in the ROS.



(a)

(b)

Fig. 1. Monitoring and diagnosis results of fault 1 based on 2-D-DPCA: (a) monitoring result; (b) fault diagnosis result.

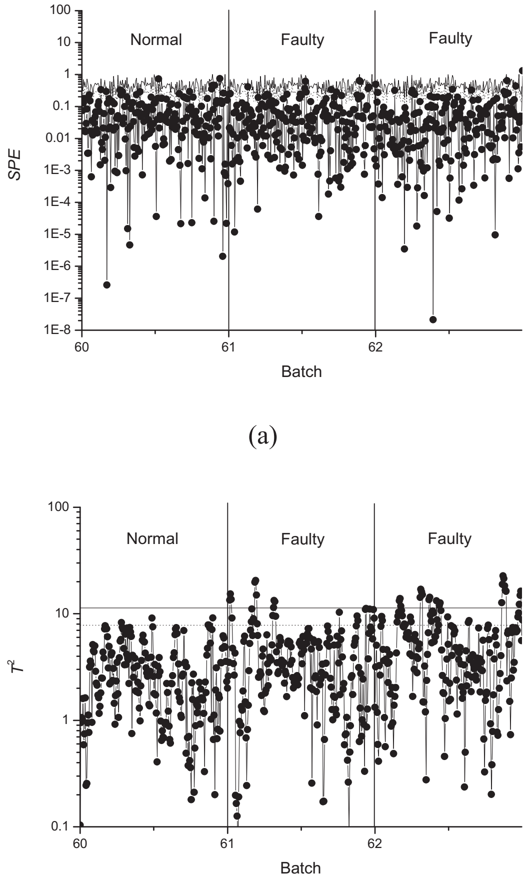


(a)

(b)

Fig. 2. Filtered variable trajectories in fault 1: (a) e_3; (b) e_4.

For multi-time-scale dynamic PCA modeling, x_1 and x_2 belong to the slow-response variable group *S*, while x_3 and x_4 are divided into the fast-response variable group *F*. The regression model between *F* and *S* is built to remove the effects of x_1 and x_2 from x_3 and x_4, as described in (2) and (3). Supposing e_3 and e_4 are the filtered values of x_3 and x_4, the variation information is separated into $G^S = \left[e_3 \quad e_4\right]$ as the short time-scale level and $G^L = \left[x_1 \quad x_2\right]$ as the long time-scale level. Then, 2-D-DPCA is performed on each level to model the two different types of dynamics. Let $\hat{\mathbf{x}}(i,k-1) = \left[x_1(i,k) \quad x_2(i,k)\right]$. In the long time-scale level, the ROS is selected as $\hat{\mathbf{x}}(i,k-1), \hat{\mathbf{x}}(i-1,k), \hat{\mathbf{x}}(i-1,k-1)$.



(a)

(b)

Fig. 3. Monitoring results of fault 1 based on short time-scale level model: (a) *SPE* plot; (b) T^2 plot.

The 2-D-DPCA model is calculated based on 2 current variables in $\hat{\mathbf{x}}(i,k)$ and 6 lagged variables in the ROS. In the short time-scale level, the algorithm is performed on 4 variables including $e_3(i,k), e_4(i,k), e_3(i,k-1), e_4(i,k-1)$.

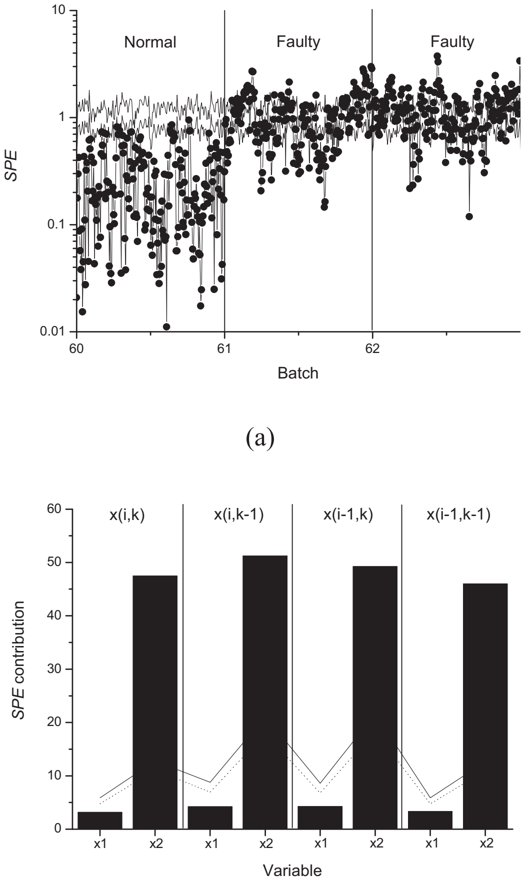
4.2 Online modeling and fault diagnosis

Two faults are introduced into the process. Fault 1 occurs to the slow-response variable x_2. From batch 61, x_2 is formulated as (6) to simulate a fault:

$$x_2(i,k) = 0.6 * x_2(i-1,k) + 0.3 * x_2(i,k-1) + 0.2 * x_2(i-1,k-1) \,. \quad (6)$$

Fig. 1 shows the monitoring and the fault diagnosis results based on conventional 2-D-DPCA, respectively. The *SPE* control chart shows that the fault can be detected from the beginning of batch 61. However, from the contribution plot of batch 61, Fig. 1(b), it is hard to say which variable is faulty. Due to the variable correlations, many variables (including the lagged variables) are outside the control limits.

In multi-time-scale monitoring, variable x_1 and x_2 are filtered to get short time-scale dynamic signals e_3 and e_4. Since the fault occurs to the slow-response variable x_2, and the effects



(a)

(b)

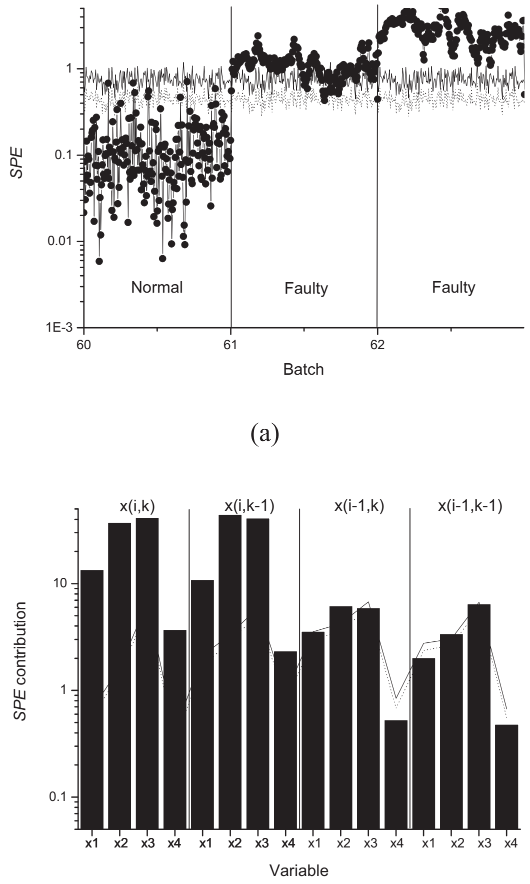
Fig. 4. Monitoring results of fault 1 based on long time-scale level model: (a) monitoring; (b) diagnosis.

of x_1 and x_2 have been removed from the short time-scale level, there is no significant difference between the trajectories of e_3 and e_4 in a normal cycle and those in the faulty cycles, as shown in Fig. 2. The monitoring results in Fig. 3 confirm this. Neither *SPE* nor T^2 plot in this level is affected by the fault significantly. At the same time, the *SPE* control plot in the other level detects the fault efficiently, as Fig. 4(a) shows. This points out that the fault happens in the long time-scale level. Then, contribution plot in this level is plotted to find out the reason of the fault. From Fig. 4(b), it is very easy to conclude that x_2 is the faulty variable.

Fault 2 is about the fast-response variable x_3. From batch 61, the formulation of x_3 becomes:

$$x_3(i,k) = 0.5 * x_3(i-1,k) + 0.25 * x_1(i,k) + 0.35 * x_2(i,k) . \qquad (7)$$

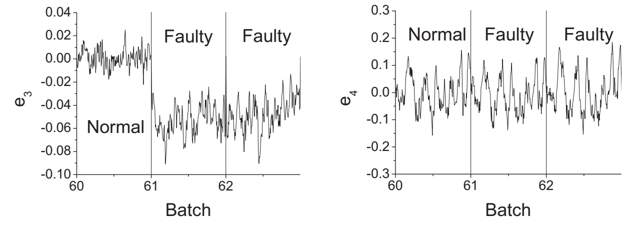
As shown in Fig. 5, again, the conventional 2-D-DPCA detects the fault very quickly, but the contribution plot can not give a clear indication about the reason of the fault. Fig. 6 shows the trajectories of e_3 and e_4. Obviously, significant magnitude differences exist between the trajectory of e_3 in a normal batch and that in faulty batches. So that, this fault is hopefully to be detected by the T^2 control chart in the short time-scale level, which is confirmed by Fig. 7(a). The



(a)

(b)

Fig. 5. Monitoring and diagnosis results of fault 2 based on 2-D-DPCA: (a) monitoring; (b) diagnosis.



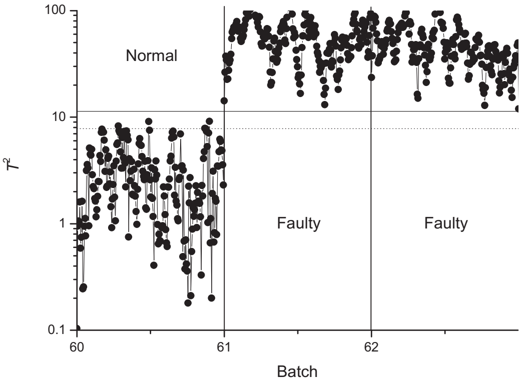
(a) (b)

Fig. 6. Filtered variable trajectories in fault 2: (a) e_3; (b) e_4.

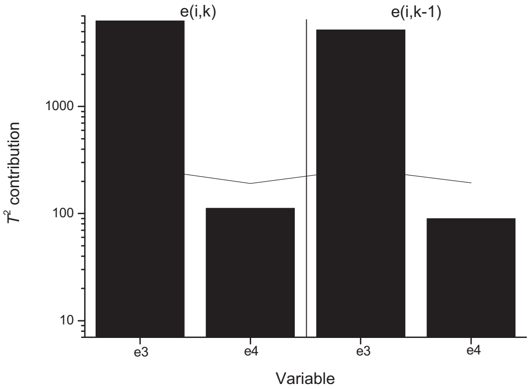
monitoring in the other level, as shown in Fig. 8, dose not show the fault, as it only occurs to a fast-response variable and dose not affect the long time-scale dynamics. The fault diagnosis is only needed to be performed in the short time-scale level. The contribution plot diagnoses the reason of the fault clearly and correctly, as Fig. 7(b) shows.

5. CONCLUSIONS

Batch process variables have various response time constants, causing dynamics with different time scales. The trajectories of the fast-response variables are often affected by the slow-



(a)



(b)

Fig. 7. Monitoring and diagnosis results of fault 2 based on short time-scale level model: (a) monitoring; (b) diagnosis.

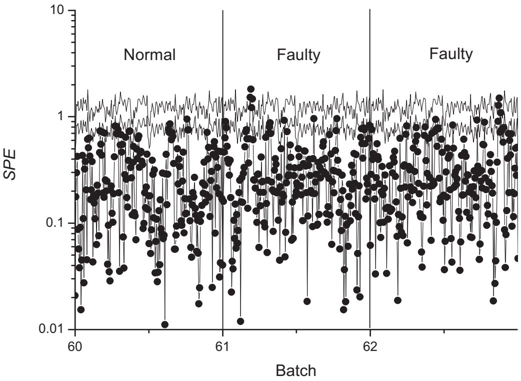


Fig. 8. Monitoring results of fault 2 based on long time-scale level model

response variables, confounding different types of dynamics and causing trouble in fault diagnosis.

A multi-time-scale dynamic PCA monitoring scheme is proposed in this paper. The process variations are separated into different levels according to the dynamics time scales. Then 2-D-DPCA method is adopted to model each level for online monitoring. The simulation results show that the fault diagnosis accuracy is largely improved.

In this paper, variable response time constants are assumed to be known as a kind of external information. It is better if such information can be achieved from the analysis of the operation data. This issue will be studied in the future researches to make the method completely data-based.

REFERENCES

Bakshi, B.R. (1998). Multiscale PCA with application to multivariate statistical process monitoring. *AIChE Journal*, 44, 1596-1610.

Chen, J. and Liu K.C. (2002). On-line batch process monitoring using dynamic PCA and dynamic PLS models. *Chemical Engineering Science*, 57, 63-75.

Jain, A.K., Murty, M.N., and Flynn, R.J. (1999). Data clustering: a review. *ACM Computing Surveys*, 31, 264-323.

Kano, M., Hasebe, S., Hashimoto, I. and Ohno, H. (2004). Evolution of multivariate statistical process control: application of independent component analysis and external analysis. *Computers and Chemical Engineering*, 28, 1157-1166.

Lu, N., Yao Y., and Gao F. (2005). Two-dimensional dynamic PCA for batch process monitoring. *AIChE Journal*, 51, 3300-3304.

Montgomery, D.C. (2005). *Design and analysis of experiments*, *6th ed*, New York: Wiley.

Nomikos, P. and MacCregor J.F. (1994). Monitoring batch processes using multiway principal component analysis. *AIChE Journal,* 40, 1361-1375.

Nomikos, P. and MacCregor J.F. (1995). Multivariate SPC charts for monitoring batch processes. *Technometrics,* 37**,** 41-59.

Takane, Y. and Shibayama, T. (1991). Principal component analysis with external information on both subjects and variables. *Psychometrika*, 56, 97–120.

Timmerman, M.E. (2006). Multilevel component analysis. *British Journal of Mathematical and Statistical Psychology*, 59, 301-320.

Westerhuis, J.A., T. Kourti, and J.F. MacGregor (1998). Analysis of multiblock and hierarchical PCA and PLS models. *Journal of Chemomet*rics, 12, 301-321.

Westerhuis, J.A., S.P. Gurden and A.K. Smilde (2000). Generalized contribution plots in multivariate statistical process monitoring. *Chemometrics and Intelligent Laboratory Systems*, 51, 95-114.

Wold, S. (1978). Cross-validatory estimation of the number of components in factor and principal components models. *Technometrics*, 20, 397-405.

Yao, Y., Diao, Y., Lu, N., Lu, J., and Gao, F. (2008). Two-dimensional dynamic principal component analysis wieht auto-determined suppert region. *Industrial & Engineering Chemistry Research*, in print.

Yoon, S. and MacGregor, J.F. (2001). Incorporation of external information into multivariate PCA/PLS models. *Proc. of 4th IFAC Workshop on On-line Fault Detection and Supervision in the Chemical Industries*, 121-126.