Korean J. Chem. Eng., **20**(6), 1000-1011 (2003)

Three-Way Data Analysis with Time Lagged Window for On-Line Batch Process Monitoring

Junghui Chen† and Jen-Hung Yen

Department of Chemical Engineering, Chung-Yuan Christian University, Chung-Li, Taiwan 320, Republic of China
(*Received 13 March 2003 • accepted 17 June 2003*)

Abstract−In this paper, on-line batch process monitoring is developed on the basis of the three-way data structure and the time-lagged window of process dynamic behavior. Two methods, DPARAFAC (dynamic parallel factor analysis) and DTri-PLS (dynamic trilinear partial least squares), are used here depending on the process variables only or on the process variables and quality indices, respectively. Although multivariate analysis using such PARAFAC (parallel factor analysis) and Tri-PLS (trilinear partial least squares) models has been reported elsewhere, they are not suited for practicing on-line batch monitoring owing to the constraints of their data structures. A simple modification of the data structure provides a framework wherein the moving window based model can be incorporated in the existing three-way data structure to enhance the detectability of the on-line batch monitoring. By a sequence of time window of each batch, the proposed methodology is geared toward giving meaningful results that can be easily connected to the current measurements without the extra computation for the estimation of unmeasured process variables. The proposed method is supported by using two sets of benchmark fault detection problems. Comparisons with the existing two-way and three-way multiway statistical process control methods are also included.

Key words: Batch Process, Process Monitoring, Statistical Process Control

INTRODUCTION

Due to the strong demand for high-value-added specialty chemicals, the batch chemical process has become more and more important. This is especially used for the products whose volume is small, such as pharmaceuticals, semiconductors, polymer, biochemicals etc. The batch process, as the name indicates, is characterized by prescribed processing of raw materials into products within finite duration. Batch processes are transient in nature with their state changing with time so that operation process variables can have significant effects on the product characteristics. These quality variables are usually difficult to measure online; they are examined off-line in a laboratory. However, inspecting a finished product does not improve its quality; it only indicates the performance of the produced product. By the time when the poor quality product has been made, it is too late to fix the process. If the off-specification product is found far upstream of the point where inspection takes place, the problem of the batch run can be detected and exposed. The appropriate procedures can also be worked out to produce an accepted product or an unaccepted product can be removed without waiting for finishing this batch. Therefore, in order to operate batch processes safely and profitably, better on-line process monitoring is strongly required.

To ensure that the operating batches match design specifications, regular collection and analysis of large quantities of process raw data are required in order to check if the incoming operating condition is still good enough. Since the final quality variable cannot be easily measured on-line, several easily measurable variables, typically following a trajectory or a profile, such as temperature, pressure and flow, are taken online. The profiles as the fingerprint of the batch operations provide vital information characteristics of the operation of the batch. However, thousands of observations are usually collected from the on-line sensor every few seconds. It is not feasible for an operator to manually monitor all process variables. It is also very tedious to extract the status of the current process based on the knowledge of the domain specialists. Ironically, some of the process specialists may be unable to inspect the measurements they collect. In fact, a poorly performed operation system may go unnoticed for a long period of time. Therefore, a systematic and automatic scheme for monitoring processes is crucial to batch process operation.

Recently, some techniques based on chemometrics have been applied to the field of batch process monitoring. Those techniques are used to extract the state of the system via applications of mathematical and statistical methods from the big volume of the historical database. The batch operation data are usually arranged in a three-way matrix with batches, measurements and their time profiles. Nomikos and MacGregor [1994] first used multiway principal component analysis (MPCA) to analyze three-way batch data. In MPCA, the three-way matrix is unfolded into a two-way matrix. Then the huge unfolded matrix is analyzed by using standard principal component analysis (PCA) and finally the result is folded back again to obtain a three-way model for the representation of the batch data. On the other hand, parallel factor analysis (PARAFAC) without any unfolding processes directly decomposes the three-way matrix into factors in each of the original dimensions [Carroll and Chang, 1970; Harshman, 1970]. It generally gets the simplest possible model since the number of parameters is significantly reduced when compared with MPCA [Dahl et al., 1999].

Several on-line batch-monitoring methods based on chemometrics were proposed [Nomikos and MacGrego, 1995; Boqué and Smilde, 1999]. They were built based on the predicted measure-

†To whom correspondence should be addressed.
E-mail: jason@wavenet.cycu.edu.tw

ments of the on-going batches to monitor the current operating condition. These methods were concerned with static rather than dynamic relationships. They implicitly assumed that the measured variable at one time instant had serial independence within each variable series at past time instances. They also indicated statistical inter-independence between different measured variable series at past time instances. Besides, extra computation was needed to estimate the future unmeasured data in order to fill up the predicted data from the current point to the end of the batch. These anticipating observations might cause false detection. To alleviate these problems, another on-line piecewise monitoring was proposed. It relied on the proper number of the monitoring points during this batch run [Chen and Liu, 2000; Louwerse and Smilde, 2000]. However, the size of time intervals could not be easily determined if the characteristics of the batch profile were unknown in advance. Our previous work [Chen and Liu, 2002] constructed a dynamic principal component analysis (BDPCA) and a batch dynamic partial least squares (BDPLS) for online batch process monitoring. Without the future data, BDPCA and BDPLS based on the current and the previous measured data incorporated the series-correlation to enhance the detectability of the on-line batch monitoring. However, these approaches were focused only on the improvement of traditional MPCA and MPLS (multiway partial least squares). With the advantage of three-way data analysis (like PARAFAC and Tri-PLS) on good compressed data, PARAFAC and Tri-PLS will be further extended to on-line monitoring in this study to construct better batch monitoring methods.

There are three goals in this paper. First, it is shown how the data structures of the PARAFAC model only focus on the relationship among all batches without considering the relationship among the process variables at different times. Second, to improve the conventional technique of the on-line monitoring, the time window frame of DPARAFAC and of Tri-DPLS for multivariable statistical process control is separately developed. Third, the effectiveness of the proposed methods in dealing with unfinished batches in on-line monitoring is supported by two benchmark problems. Comparisons with the existing MSPC methods, including MPCA, PARAFAC, MPLS and Tri-PLS, will also be done.

PAST RELATED RESEARCH REVIEW: PARAFAC AND Tri-PLS

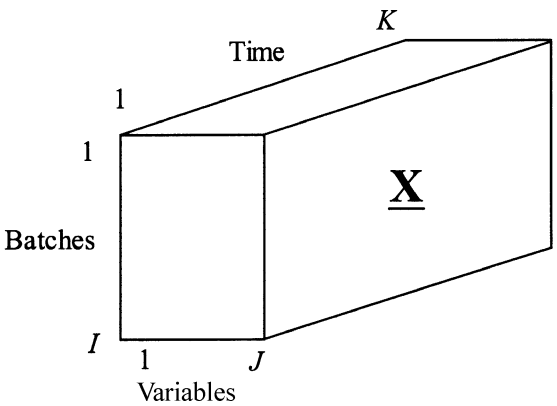

Fig. 1. The representation of batch measurements in a three-way array with I batches, J variables and K times.

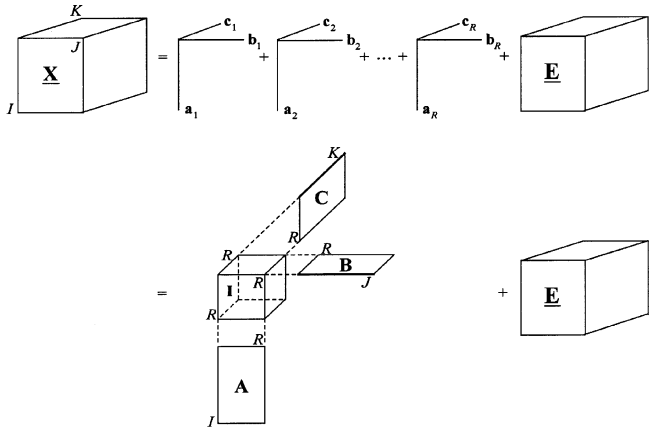

Fig. 2. The PARAFAC model is decomposed into a set of triads. Each triad contains three vectors (a_r, b_r and c_r) (top). The combination of these vectors (a_r, b_r and c_r) forms A, B and C loading matrices of the row, column and layer-direction of $\underline{\mathbf{X}}$, respectively (bottom). The core matrix I is a R×R×R three-way identity matrix.

The three-way experimental data structure is composed of quality measures, various batches and times of observations as shown in Fig. 1. A data matrix $\underline{\mathbf{X}}$ (I×J×K) with I batches, J variables, and K time points is defined here. In this section, a brief overview of the three-way data analysis techniques, PARAFAC and Tri-PLS, is presented. For a more comprehensive reading and discussion of practical applications, extensive literature can be found [Bro, 1997; Wise et al., 1999; Louwerse and Smilde, 2000].

1. PARAFAC

The PARAFAC model is the decomposition of a three-way data $\underline{\mathbf{X}}$ (I×J×K) in terms of the sums of the Kronecker products of three vectors $\mathbf{a}_r$, $\mathbf{b}_r$ and $\mathbf{c}_r$ [Harshman, 1970; Geladi, 1989] (Fig. 2)

$$\underline{\mathbf{X}} = \sum_{r=1}^{R} \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r + \underline{\mathbf{E}} \tag{1}$$

where $\mathbf{a}_r$, $\mathbf{b}_r$ and $\mathbf{c}_r$ are a set of independent loading vectors of the *r*-th component.

With the given Gaussian noise ($\underline{\mathbf{E}}$), the solution of Eq. (1) is considered as the minimization of the following problem:

$$\min_{\substack{a_r, b_r, c_r \\ r=1,\ldots,R}} \left\| \underline{\mathbf{X}} - \sum_{r=1}^{R} \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r \right\| \tag{2}$$

Note that the above equation is wrong. The norm operation cannot be applied to the three-way array. An alternative method slices the array from different directions and puts each slice side by side to form two-way matrices. The three two-way matrices of the PARAFAC model can be represented by

$$\begin{aligned} \mathbf{X}^{(1)} &= \mathbf{A}(\mathbf{C} \circ \mathbf{B})^T + \mathbf{E}^{(1)} \\ \mathbf{X}^{(2)} &= \mathbf{B}(\mathbf{C} \circ \mathbf{A})^T + \mathbf{E}^{(2)} \\ \mathbf{X}^{(3)} &= \mathbf{C}(\mathbf{B} \circ \mathbf{A})^T + \mathbf{E}^{(3)} \end{aligned} \tag{3}$$

where o is the Khatri-Rao product of two matrices partitioned in the column [Rao and Mitra, 1971]. **A**, **B** and **C** with the column vectors $\mathbf{a}_r$, $\mathbf{b}_r$ and $\mathbf{c}_r$, r=1, 2, ⋯, R. $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ and $\mathbf{X}^{(3)}$ are unfolded matrices with respect to the frontal, horizontal and lateral slices of the

three-way array, respectively,

$$\mathbf{X}^{(1)}=\begin{bmatrix} x^1_{1,1} & x^1_{2,1}\cdots x^1_{J,1} & x^1_{1,2} & x^1_{2,2}\cdots x^1_{J,2}\cdots x^1_{1,K} & x^1_{2,K}\cdots x^1_{J,K} \\ x^2_{1,1} & x^2_{2,1}\cdots x^2_{J,1} & x^2_{1,2} & x^2_{2,2}\cdots x^2_{J,2}\cdots x^2_{1,K} & x^2_{2,K}\cdots x^2_{J,K} \\ \vdots & \vdots\ \ddots\ \vdots & \vdots & \vdots\ \ddots\ \vdots\ \cdots\ \vdots & \vdots\ \ddots\ \vdots \\ x^I_{1,1} & x^I_{2,1}\cdots x^I_{J,1} & x^I_{1,2} & x^I_{2,2}\cdots x^I_{J,2}\cdots x^I_{1,K} & x^I_{2,K}\cdots x^I_{J,K} \end{bmatrix}_{I\times JK} \tag{4}$$

$$\mathbf{X}^{(2)}=\begin{bmatrix} x^1_{1,1} & x^1_{1,2}\cdots x^1_{1,K} & x^2_{1,1} & x^2_{1,2}\cdots x^2_{1,K}\cdots x^I_{1,1} & x^I_{1,2}\cdots x^I_{1,K} \\ x^1_{2,1} & x^1_{2,2}\ldots x^1_{2,K} & x^2_{2,1} & x^2_{2,2}\cdots x^2_{2,K}\cdots x^I_{2,1} & x^I_{2,2}\cdots x^I_{2,K} \\ \vdots & \vdots\ \ddots\ \vdots & \vdots & \vdots\ \ddots\ \vdots\ \cdots\ \vdots & \vdots\ \ddots\ \vdots \\ x^1_{J,1} & x^1_{J,2}\cdots x^1_{J,K} & x^2_{J,1} & x^2_{J,2}\cdots x^2_{J,K}\cdots x^I_{J,1} & x^I_{J,2}\cdots x^I_{J,K} \end{bmatrix}_{J\times IK} \tag{5}$$

and

$$\mathbf{X}^{(3)}=\begin{bmatrix} x^1_{1,1} & x^1_{2,1}\cdots x^1_{J,K} & x^2_{1,1} & x^2_{2,1}\cdots x^2_{J,1}\cdots x^I_{1,1} & x^I_{2,1}\cdots x^I_{J,1} \\ x^1_{1,2} & x^1_{2,2}\cdots x^1_{J,2} & x^2_{1,2} & x^2_{2,2}\cdots x^2_{J,2}\cdots x^I_{1,2} & x^I_{2,2}\cdots x^I_{J,2} \\ \vdots & \vdots\ \ddots\ \vdots & \vdots & \vdots\ \ddots\ \vdots\ \cdots\ \vdots & \vdots\ \ddots\ \vdots \\ x^1_{1,K} & x^1_{2,K}\cdots x^1_{J,K} & x^2_{1,K} & x^2_{2,K}\cdots x^2_{J,K}\cdots x^I_{1,K} & x^I_{2,K}\cdots x^I_{J,K} \end{bmatrix}_{K\times IJ} \tag{6}$$

where $x^i_{j,k}$ is the observation of variable j at the time point k in batch run i. The solution of the PARAFAC model can be found by alternating least squares (ALS), which successively uses the two-given loading models, and then estimating the last loading model parameters [Kiers and Krijnen, 1991]. The procedure is described as follows:

Step 1: Select the number of components, R.

Step 2: Estimate $\mathbf{A}$ from $\mathbf{X}^{(1)}$ by the least squares method,

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \left\| \mathbf{X}^{(1)} - \mathbf{A}(\mathbf{C}\circ\mathbf{B}) \right\| \tag{7}$$

The least squares method updates $\mathbf{A}$ with the given $\mathbf{B}$ and $\mathbf{C}$,

$$\mathbf{A}=\mathbf{X}^{(1)}\mathbf{Z}^T(\mathbf{Z}\mathbf{Z}^T)^{-1} \tag{8}$$

where $\mathbf{Z}=(\mathbf{C}\circ\mathbf{B})^T$ and $\mathbf{Z}=\{z_{ij}\}_{R\times JK}$

Step 3: Estimate $\mathbf{B}$ from $\mathbf{X}^{(2)}$ by the same method of Eq. (7),

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \left\| \mathbf{X}^{(2)} - \mathbf{B}(\mathbf{C}\circ\mathbf{A}) \right\| \tag{9}$$

with the given $\mathbf{A}$ and $\mathbf{C}$, $\mathbf{B}$ can be computed by

$$\mathbf{B}=\mathbf{X}^{(2)}\mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1} \tag{10}$$

where $\mathbf{D}=(\mathbf{C}\circ\mathbf{A})^T$ and $\mathbf{D}=\{d_{ij}\}_{R\times IK}$

Step 4: Estimate $\mathbf{C}$ from $\mathbf{X}^{(3)}$ by minimizing the following problem

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \left\| \mathbf{X}^{(3)} - \mathbf{C}(\mathbf{B}\circ\mathbf{A}) \right\| \tag{11}$$

with the given $\mathbf{A}$ and $\mathbf{B}$, $\mathbf{C}$ can be solved by

$$\mathbf{C}=\mathbf{X}^{(3)}\mathbf{V}^T(\mathbf{V}\mathbf{V}^T)^{-1} \tag{12}$$

where $\mathbf{V}=(\mathbf{B}\circ\mathbf{A})^T$ and $\mathbf{V}=\{v_{ij}\}_{R\times IJ}$

Step 5: Repeat Step 2-4 until $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are converged. The three-way matrix is decomposed into three loading matrices $\mathbf{A}$(I×R), $\mathbf{B}$ (J×R), and $\mathbf{C}$(K×R).

The PARAFAC model takes advantage of the unique solution [Kruskal, 1977] and eliminates the need for additional factor rotation processes [Harshman and Lundy, 1984]. Like the representation of the conventional PCA model, the two-way matrix representation of PARAFAC is given by

$$\mathbf{X}^{(1)}=\mathbf{A}(\mathbf{C}\circ\mathbf{B})^T+\mathbf{E}^{(1)}=\mathbf{T}\mathbf{P}^T+\mathbf{E}^{(1)} \tag{13}$$

where $\mathbf{T}=\mathbf{A}$ is the score vector and $\mathbf{P}=\mathbf{C}\circ\mathbf{B}$ is the loading vector. Based on Eqs. (10) and (12) the loading matrices $\mathbf{B}$ and $\mathbf{C}$ are partially expanded:

$$\mathbf{B}=\begin{bmatrix} \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{1,k}d_{(i-1)K+k,1} & \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{1,k}d_{(i-1)K+k,2} & \cdots & \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{1,k}d_{(i-1)K+k,R} \\ \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{2,k}d_{(i-1)K+k,1} & \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{2,k}d_{(i-1)K+k,2} & \cdots & \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{2,k}d_{(i-1)K+k,R} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{J,k}d_{(i-1)K+k,1} & \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{J,k}d_{(i-1)K+k,2} & \cdots & \sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{J,k}d_{(i-1)K+k,R} \end{bmatrix}(\mathbf{D}\mathbf{D}^T)^{-1} \tag{14}$$

and

$$\mathbf{C}=\begin{bmatrix} \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,1}v_{(i-1)J+j,1} & \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,1}v_{(i-1)J+j,2} & \cdots & \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,1}v_{(i-1)J+j,R} \\ \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,2}v_{(i-1)J+j,1} & \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,2}v_{(i-1)J+j,2} & \cdots & \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,2}v_{(i-1)J+j,R} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,K}v_{(i-1)J+j,1} & \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,K}v_{(i-1)J+j,2} & \cdots & \sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,K}v_{(i-1)J+j,R} \end{bmatrix}(\mathbf{V}\mathbf{V}^T)^{-1} \tag{15}$$

Note that the elements of the first matrices of $\mathbf{B}$ and $\mathbf{C}$, $\sum_{i=1}^{I}\sum_{k=1}^{K} x^i_{j,k}$ $d_{(i-1)K+k,r}$ and $\sum_{i=1}^{I}\sum_{j=1}^{J} x^i_{j,k}v_{(i-1)J+j,r}$ are only the combinational relationships between all batches and the time, and between all batches and the variables, respectively. This implies that the serial correlations among the batch process variables at two different times in this model are not considered. This will limit the PARAFAC model to the off-line batch process monitoring.

If the PARAFAC model is applied to on-line monitoring, estimated future data are needed in order to fill up the unmeasured data from the current time to the end of the batch (Fig. 3), because PARAFAC models require complete history of the batch data. The extra computational load for anticipating future observations is needed. Besides, the estimated values may not exactly follow the actual dy-

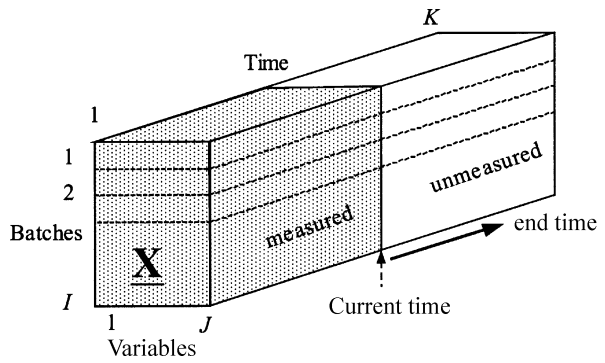

Fig. 3. Anticipating the future unmeasured variables in the on-line PARAFAC model.

namic process behavior and they may lead to false detection.

2. Tri-PLS

Tri-PLS is used to predict properties of batch processes based on those variables only indirectly related to the properties under three-way partial least squares. This extends the traditional MPLS to three-orders. The given experimental data are subdivided into two blocks, a dependent block (**Y**) and an independent block ($\underline{\mathbf{X}}$). $\underline{\mathbf{X}}$ block I×J×K is decomposed into a set of triads. Each triad is a rank-one model of the array. It consists of one score vector and two weight vectors. These two weight vectors separately represent variable ($\mathbf{w}^J$) and time ($\mathbf{w}^K$) directions. **Y** block organizes a two-way array (I×M) that summarizes the final quality variables (m=1, 2, ⋯, M) in each batch run (Fig. 4). The goal of Tri-PLS is to determine a set of triads that is correlated with **Y** while describing a large amount of the variation in $\underline{\mathbf{X}}$ [Bro, 1996]. It can be formulated as

$$[\mathbf{w}^J, \mathbf{w}^K] = \arg \max_{\mathbf{w}^J, \mathbf{w}^K} [(\mathbf{w}^J)^T \mathbf{S} \mathbf{w}^K] \tag{16}$$

where $\mathbf{S} = \left\{ \sum_{i=1}^{I} y_i x_{jk}^i \right\}$. Thus, **Y** and $\underline{\mathbf{X}}$ matrices are decomposed into the summation of the product of score vectors $\mathbf{t}_r$ and loading vectors $\mathbf{p}_r (= \mathbf{b}_r \otimes \mathbf{c}_r)$ and $\mathbf{q}_r$ plus some residual matrices **E** and **F**:

$$\mathbf{T} = \mathbf{X}^{(1)} \mathbf{W}$$

$$\mathbf{X} = \sum_{r=1}^{R} \mathbf{t}_r \mathbf{p}_r^T + \mathbf{E} = \mathbf{T}\mathbf{P}^T + \mathbf{E}$$

$$\mathbf{Y} = \sum_{r=1}^{R} \mathbf{t}_r \mathbf{q}_r^T + \mathbf{F} = \mathbf{T}\mathbf{Q}^T + \mathbf{F} \tag{17}$$

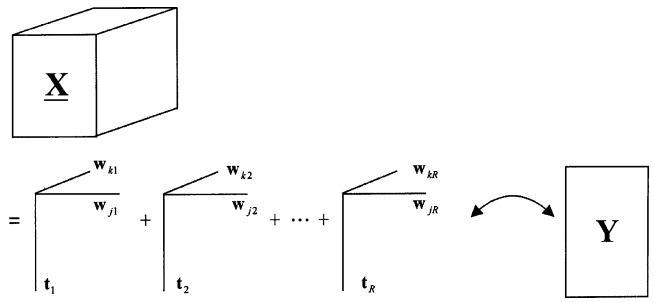

Fig. 4. The representation of the Tri-PLS model, where $\underline{\mathbf{X}}$ is decomposed into a set of triads and the triads are significantly correlated with Y.

where $\mathbf{W} = [\mathbf{w}_1^K \otimes \mathbf{w}_1^J \ \mathbf{w}_2^K \otimes \mathbf{w}_2^J \ \cdots \ \mathbf{w}_R^K \otimes \mathbf{w}_R^J]$. R is the number of principal components retained in Tri-PLS. Like PARAFAC, Tri-PLS still needs computation of the future unmeasured batch data when on-line monitoring is applied.

ON-LINE BATCH MONITORING USING DYNAMIC PARAFAC

The PARAFAC or Tri-PLS models previously discussed are good for offline batch monitoring because they are concerned with statistic rather than dynamic relationships of the batch. A simple method that includes the serial correlation in the process variables makes use of the concept of the moving window. An auto-regressive model structure for batch i at the time point k can be represented as,

$$\mathbf{X}_d^i(\mathrm{k}) = [\mathbf{x}^i(\mathrm{k}) \ \mathbf{x}^i(\mathrm{k}-1) \ \cdots \ \mathbf{x}^i(\mathrm{k}-\mathrm{d})] \tag{18}$$

where $\mathbf{x}^i(\mathrm{k}) = [x_{1,k}^i \ x_{2,k}^i \ \ldots \ x_{J,k}^i]^T$ is the J-dimensional observation vector at time point k. d is the window length of the dynamic process. $\mathbf{X}_d^i(\mathrm{k})$ can be defined as a data window at time point k. A similar concept was also applied to the PCA model to capture the dynamic behavior of process variables [Ku et al., 1995; Wachs and Lewin, 1999]. Thus, K-d data windows are obtained by cutting along the time dimension of the data series. The three-way data array is developed by stacking K-d slices with the size (d+1)×J together shown in Fig. 5. The three-way data array $\underline{\mathbf{X}}_d^i$ is defined by (K-d)×J×(d+1). The data set has the same structure as shown in Fig. 1. PARAFAC can be applied to analyzing this data array, $\underline{\mathbf{X}}_d^i$.

According to the definition of $\mathbf{X}^{(1)}$ in Eq. (3) $\underline{\mathbf{X}}_d^i$ can be represented by

$$(\mathbf{X}_d^i)^{(1)} = \mathbf{A}^i (\mathbf{C}^i \mathrm{o} \mathbf{B}^i)^T + \mathbf{E}^i \tag{19}$$

where

$$(\mathbf{X}_d^i)^{(1)} = \begin{bmatrix} (vec\mathbf{X}_d^i(\mathrm{d}+1))^T \\ (vec\mathbf{X}_d^i(\mathrm{d}+2))^T \\ \vdots \\ \vdots \\ (vec\mathbf{X}_d^i(\mathrm{K}))^T \end{bmatrix}$$

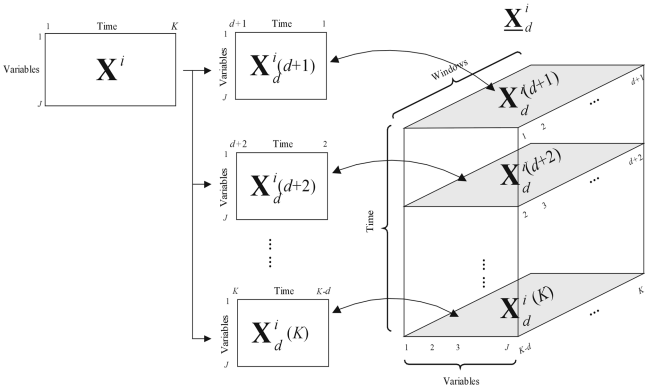

Fig. 5. The representation of the three-way array with the time window for the measurements of each batch.

$$= \begin{bmatrix} (\mathbf{x}^i(\mathrm{d}+1))^T & (\mathbf{x}^i(\mathrm{d}))^T & \cdots & (\mathbf{x}^i(1))^T \\ (\mathbf{x}^i(\mathrm{d}+2))^T & (\mathbf{x}^i(\mathrm{d}+1))^T & \cdots & (\mathbf{x}^i(2))^T \\ & \vdots & & \vdots \\ & \vdots & & \vdots \\ (\mathbf{x}^i(\mathrm{K}))^T & (\mathbf{x}^i(\mathrm{K}-1))^T & \cdots & (\mathbf{x}^i(\mathrm{K}-\mathrm{d}))^T \end{bmatrix} \quad (20)$$

vec is an operator that transforms a matrix into one long column vector by staking one column of the matrix after another one. Like the representation of PARAFAC in Eq. (10) and (12), $\mathbf{B}^i$ and $\mathbf{C}^i$ are expanded into

$$\mathbf{B}^i = \begin{bmatrix} \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{1,(m+n)} v_{(m+n),1} & \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{1,(m+n)} v_{(m+n),2} \\ \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{2,(m+n)} v_{(m+n),1} & \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{2,(m+n)} v_{(m+n),2} \\ \vdots & \vdots \\ \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{J,(m+n)} v_{(m+n),1} & \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{J,(m+n)} v_{(m+n),2} \end{bmatrix}$$

$$\begin{bmatrix} \cdots \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{1,(m+n)} v_{(m+n),F} \\ \cdots \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{2,(m+n)} v_{(m+n),F} \\ \ddots \quad \vdots \\ \cdots \sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{J,(m+n)} v_{(m+n),F} \end{bmatrix} (\mathbf{D}^i(\mathbf{D}^i)^T)^{-1} \quad (21)$$

$$\mathbf{C}^i = \begin{bmatrix} \sum_{j=1}^{J}\sum_{m=1}^{K-d} x^i_{j,m} l_{(j-1)(K-d)+m,1} & \sum_{j=1}^{J}\sum_{m=1}^{K-d} x^i_{j,m} l_{(j-1)(K-d)+m,2} \\ \sum_{j=1}^{J}\sum_{m=d}^{K-1} x^i_{j,m} l_{(j-1)(K-d)+m,1} & \sum_{j=1}^{J}\sum_{m=d}^{K-1} x^i_{j,m} l_{(j-1)(K-d)+m,2} \\ \vdots & \vdots \\ \sum_{j=1}^{J}\sum_{m=d+1}^{K} x^i_{j,m} l_{(j-1)(K-d)+m,1} & \sum_{j=1}^{J}\sum_{m=d+1}^{K} x^i_{j,m} l_{(j-1)(K-d)+m,2} \end{bmatrix}$$

$$\begin{bmatrix} \cdots \sum_{j=1}^{J}\sum_{m=1}^{K-d} x^i_{j,m} l_{(j-1)(K-d)+m,F} \\ \cdots \sum_{j=1}^{J}\sum_{m=d}^{K-1} x^i_{j,m} l_{(j-1)(K-d)+m,F} \\ \ddots \quad \vdots \\ \cdots \sum_{j=1}^{J}\sum_{m=d+1}^{K} x^i_{j,m} l_{(j-1)(K-d)+m,F} \end{bmatrix} (\mathbf{V}^i(\mathbf{V}^i)^T) \quad (22)$$

where $\mathbf{D}^i=\{d_{ij}\}_{R\times(K-d)(d+1)}$ and $\mathbf{V}^i=\{v_{ij}\}_{R\times J(K-d)}$. Each element of the first matrix of $\mathbf{B}^i$, $\sum_{n=0}^{K-d-1}\sum_{m=1}^{d+1} x^i_{j,(m+n)} v_{(m+n),r}$, implicitly contains the dynamic relationship for variable j at different time in batch i. Also each element of $\mathbf{C}^i$, $\sum_{j=1}^{J}\sum_{m=d}^{K-1} x^i_{j,m} l_{(j-1)(K-d)+m,2}$ has implicitly dynamic relationships between all variables and the time windows. This means that the PARAFAC model with the moving window term is involved with the serial correlations among the batch process variables.

If samples of I batch runs are available, the average directions of all batch runs can be estimated by pooling each batch loading matrices $\mathbf{B}^i$ and $\mathbf{C}^i$, i.e. $\mathbf{B}^{avg}=\frac{1}{I}\sum_{i=1}^{I}\mathbf{B}^i$ and $\mathbf{C}^{avg}=\frac{1}{I}\sum_{i=1}^{I}\mathbf{C}^i$. When the operating batch is measured at the time point k, a data window using the data information up to the current time point, $\mathbf{X}^i_d(k)$, is projected onto the lower dimensional space of the PARAFAC model,

$$\mathbf{a}^i(k)=[(\mathbf{P}^{avg})^T\mathbf{P}^{avg}]^{-1}(\mathbf{P}^{avg})^T vec(\mathbf{X}^i_d(k)) \quad (23)$$

where $\mathbf{a}^i(k)=(\mathbf{t}^i(k))^T$ and $\mathbf{P}^{avg}=\mathbf{B}^{avg}\mathrm{o}\mathbf{C}^{avg}$. The residuals of the current measurement are,

$$vec(\mathbf{E}^i(k))=vec(\mathbf{X}^i_d(k))-\mathbf{P}^{avg}\mathbf{a}^i(k) \quad (24)$$

where

$$\mathbf{E}^i(k)=[\mathbf{e}^i(k)\ \ \mathbf{e}^i(k-1)\ \ \cdots\ \ \mathbf{e}^i(k-d)] \quad (25)$$

To determine whether the current operating condition falls in the accepted region, like the conventional MPSC approach, two statistical monitoring charts for on-line monitoring are derived from two spaces, (i) principal component space, and (ii) residual space, based on the *residuals statistics* (Q-chart) and the *Hotelling statistic* (T^2-chart), respectively [Jackson, 1991].

Q-chart: the sum of squared residuals of batch i can detect the type of abnormal event when the PARAFAC model defined by the first R principal components is inadequate to describe the upset event.

$$Q^i_k=(\mathbf{e}^i(k))^T(\mathbf{e}^i(k))=\sum_{j=1}^{J}(e^i_{jk})^2 \quad (26)$$

where Q^i_k represents Q value of batch i at time point k. e^i_{jk} is the residual j of the batch i at time point k. The distribution of Q_k value follows a chi-square distribution [Box, 1954]. The control limit on the Q-chart at significance level α for time interval k can be computed, where α is typical 0.99 or 0.95.

T^2-chart: a measure of the major variation of the sample $\mathbf{x}^i(k)$ to the lower dimensional space model is defined as:

$$(T^i_k)^2=(\mathbf{t}(k))^T\mathbf{\Lambda}^{-1}\mathbf{t}(k)\frac{(K-d+1)(K-d+1-R)}{R((K-d+1)^2-1)} \quad (27)$$

where $(T^i_k)^2$ is T^2 value of batch i at time point k and $\mathbf{\Lambda}^{-1}$ represents the estimated covariance matrix of the t-scores of all batch runs. It is the diagonal matrix containing the inverse of eigenvalues associated with the R eigenvectors (or principal directions) retained in the model. The control limit T^2_α can be approximated by means of the F-distribution, $T^2_\alpha \sim F_{r,K-d+1-r,\alpha}$.

To sum up, the DPARAFAC approach is schematically given hereafter:

DPARAFAC Modeling Procedures:

1. Collect the historical batch data sets indicative of normal operations. The data should cover the range of the batch operating patterns and the conditions that yield desired product quality.

2. Pre-treat the collected data before the model is developed. Auto-

scaling each measurement variable that centers and scales the variance to unit one is applied. This will avoid the important variables of small magnitudes from being taken over by less important but larger magnitude variables.

3. Select an appropriate number of time intervals throughout a data window (d) of all batches in order to capture the dynamic relations among batch data and sum them to get the average time lags.

4. Construct the three-dimensional data array for each batch by stacking the windows together, as shown in Fig. 5, and the data windows may have overlap.

5. Apply PARAFAC to analyzing each batch array ($\mathbf{X}_d^i$, i=1, 2, ⋯, I). The average loading matrices ($\mathbf{P}^{avg}=\mathbf{B}^{avg}\otimes\mathbf{C}^{avg}$) among all batches are produced.

6. Compute the score for all batches at each time point to set up the control limits Q_k and T_k^2 for each time point k individually.

On-line Monitoring Procedures:

7. Record the operating data at a new sampling time of a new batch run. Arrange the data window with previous data and the current data [Eq. (18)].

8. Project the data window onto the previously selected two subspace dominant feature directions of the DPARAFAC model. Compute the *residuals statistics* (Q) and the *Hotelling statistic* (T^2) [Eqs. (26) and (27)].

9. Check if the current time point is in the control limits. If either Q or T^2 is above its control limit, the operator is alerted and one further analyzes what has caused the abnormal situation. Otherwise, keep monitoring the next new time point and go to Step 7.

ON-LINE BATCH MONITORING WITH Tri-DYNAMIC PLS

To extend three-way array of DPARAFAC, the dynamic tri-dynamic PLS (DTri-DPLS) is developed. It consists of two data arrays with respect to process variables and quality variables. Process variables ($\underline{\mathbf{X}}$) are rearranged into a dynamic data matrix with a time lagged data window for each batch. Because the quality variables ($\mathbf{Y}$) are only measured at the end of each batch run, not available throughout the duration of each batch run. In order to incorporate the quality variables into the dynamic data matrix of process variables, the number of rows in the quality variables should be properly arranged to make a consistent size with that of the process variable data matrix. In Fig. 6, the values of the quality variables are duplicated at each row. The data is structured because quality at the end of the batch run is accepted if the operating condition of the data window ($\mathbf{X}_d^i(k)$) at the current time point *k* is normal.

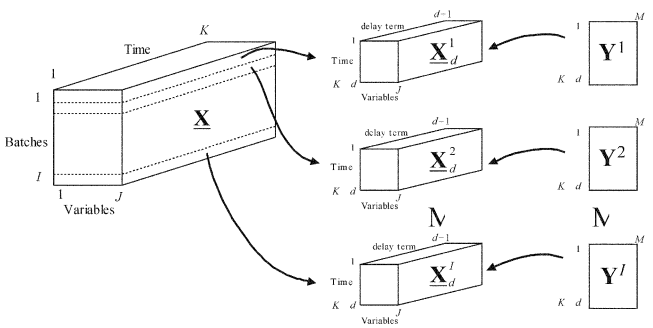

Fig. 6. Arrangement of a three-way array into I three-way dynamic arrays and the corresponding quality matrices.

The goal of DTri-PLS is to make a decomposition of the three-way array $\underline{\mathbf{X}}_d^i$ into a set of triads to maximize covariance with $\mathbf{Y}^i$ for each batch. Each triad consists of one score vector ($\mathbf{t}_r^i$) and two weight-direction vectors (($\mathbf{w}_r^J$)^i and ($\mathbf{w}_r^d$)^i). Like the previous discussion of the Tri-PLS algorithm, the algorithm for constructing the DTri-PLS model to compute the parameters is derived as follows. Let $\underline{\mathbf{H}}_0=\underline{\mathbf{X}}_d^i$ and $\mathbf{F}_0=\mathbf{Y}^i$. For simplifying the representation, the batch index i is removed from Eq. (28) to Eq. (34).

$$[\mathbf{w}_r^J, \mathbf{w}_r^d] = \arg \max_{\mathbf{w}_r^J, \mathbf{w}_r^K} [(\mathbf{w}_r^J)^T \mathbf{S}_r \mathbf{w}_r^d] \tag{28}$$

$$\mathbf{t}_r = \mathbf{H}_r \mathbf{w}_r \tag{29}$$

$$\mathbf{q}_r = \frac{\mathbf{F}_r \mathbf{t}_r}{|\mathbf{F}_r \mathbf{t}_r|} \tag{30}$$

$$\mathbf{u}_r = \mathbf{F}_r \mathbf{q}_r \tag{31}$$

where $\mathbf{w}_r=\mathbf{w}_r^d\otimes\mathbf{w}_r^J$, $\mathbf{S}_r=\{(s_{jk})_r\}=\left\{\sum_{ii=1}^{K-d}(u_{ii})_r(h_{jk}^{ii})_r\right\}$, $\mathbf{F}_r=\{(f_{jk})_r\}$ and $\mathbf{H}_r=\{(h_{jk}^{ii})_r\}$. Repeat the above procedures until convergence. The regression coefficient related $\underline{\mathbf{X}}$ and $\mathbf{Y}$ can be computed from

$$\mathbf{m}_r = (\mathbf{T}_r^T \mathbf{T}_r)^{-1} \mathbf{T}_r^T \mathbf{u}_r \tag{32}$$

where $\mathbf{T}_r=[\mathbf{t}_1\ \mathbf{t}_2\ \cdots\ \mathbf{t}_r]$. With the given $\mathbf{t}_r$ and $\mathbf{H}_r$, $\mathbf{b}_r$ and $\mathbf{c}_r$ are easily determined by the least-squares solution. The residual array for the next iteration is calculated from

$$\underline{\mathbf{H}}_r = \underline{\mathbf{H}}_{r-1} - \mathbf{t}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r \tag{33}$$

and

$$\mathbf{F}_r = \mathbf{F}_{r-1} - \mathbf{T}_r \mathbf{m}_r \mathbf{q}_r^T \tag{34}$$

This removes the variance associated with the already calculated r-th directions of $\mathbf{b}_r\otimes\mathbf{c}_r$ and $\mathbf{q}_r$ in the variance of process variables and quality variables, respectively. The entire procedure is repeated by using Eqs. (28)-(34) for the next component r+1 until the description of $\mathbf{Y}$ is properly obtained. For all batches, the average of cross correlation between $\underline{\mathbf{X}}_d^i$ and $\mathbf{Y}^i$, $\mathbf{S}_r^{avg}=\frac{(K-d+1)}{J(k-d)}\sum_{i=1}^{I}\mathbf{S}_r^i$, is applied here. Thus, the DTri-PLS model is given by the equation,

$$\underline{\mathbf{X}}_d^i = \sum_{r=1}^{R} \mathbf{t}_r^i \otimes \mathbf{b}_r^{avg} \otimes \mathbf{c}_r^{avg} + \underline{\mathbf{E}}^i \tag{35}$$

If the operating batch at the time point k is the data window using the data information up to the current time point, $\mathbf{X}_d^i(k)$, is projected onto the lower dimensional space of the DTri-PLS model,

$$\begin{aligned}
t_1^i(k) &= (vec\,\mathbf{X}_d^i(k))^T \mathbf{w}_1^{avg} \\
t_2^i(k) &= ((vec\,\mathbf{X}_d^i(k))^T - t_1^i(k)(\mathbf{w}_1^{avg})^T)\mathbf{w}_2^{avg} \\
&= vec(\mathbf{X}_d^i(k))^T (\mathbf{I} - \mathbf{w}_1^{avg}(\mathbf{w}_1^{avg})^T)\mathbf{w}_2^{avg} \\
&\ \ \cdots \qquad \cdots \\
t_R^i(k) &= (vec\,\mathbf{X}_d^i(k))^T (\mathbf{I} - \mathbf{w}_1^{avg}(\mathbf{w}_1^{avg})^T) \cdots\cdots (\mathbf{I} - \mathbf{w}_{R-1}^{avg}(\mathbf{w}_{R-1}^{avg})^T)\mathbf{w}_R^{avg}
\end{aligned} \tag{36}$$

where $\mathbf{w}_i^{avg}=(\mathbf{w}_i^d)^{avg}\otimes(\mathbf{w}_i^J)^{avg}$, i=1, 2, ⋯, R. The residuals of the current measurement is,

$$vec(\mathbf{E}^i(k)) = vec(\mathbf{X}_d^i(k)) - \mathbf{P}^{avg}\mathbf{a}^i(k) \tag{37}$$

where $\mathbf{P}^{avg}=[\mathbf{b}_1^{avg}\otimes\mathbf{c}_1^{avg}\cdots\mathbf{b}_R^{avg}\otimes\mathbf{c}_R^{avg}]$ and $\mathbf{a}^i(k)=[t_1(k)\cdots t_R(k)]^T$.

As mentioned previously, even if the quality variable is measured at the end of the batch run, DTri-PLS still has control limits in the process variables: the *residuals statistics* (Q-chart) and the *Hotelling statistic* (T^2-chart). The current operating condition of Q-chart and T^2-chart values of the DTri-PLS model is still only based on the current and the previous data without the future estimated measurements. Furthermore, DTri-PLS focuses on quality variables and it can better determine the relationship between process variables and the quality variables (i.e., the maximum covariance between process variables and quality variables). The modeling and monitoring procedures of DTri-PLS for on-line monitoring are the same as that of DPARAFAC, except for the loading $\mathbf{p}_i^{avg}$ vector in DTri-PLS indirectly obtained from the weighted vector ($\mathbf{w}_i^{avg}$).

EXAMPLES

The proposed algorithm is applied to two benchmark data sets: a DuPont industrial batch polymerization reactor and an exothermic batch reactor. Each example will be discussed in the subsections as follows.

1. Example 1: DuePont Benchmark Data

Data from an industrial batch polymerization [Nomikos and MacGregor, 1995a] are used here to demonstrate the application of the proposed method. The batch reactor consists of two stages. Approximately two hours is needed to finish one batch run. The critical property measurements are usually taken twelve hours or more each batch run is finished. First, the model is built upon a historical data set of the normal operating condition batches. A total of successful 36 batches of data sets are screened and collected from 55 batches. Each successful batch run has a duration of 100 time intervals. Ten variables are measured during the batch run, including temperatures, pressures and flow rate. The data set is arranged in a three-way array $\mathbf{X}(36\times10\times100)$.

In on-line DPARFAC, the average time lagged window of all batches is 7. The data set for each batch is arranged in 93 time-lagged window matrices [Eq. (14)]. Then these matrices are built into a three-way array and are decomposed by using PARAFAC for each batch. It is found, via cross-validation, that three principal components are needed to describe the data set. This model captures 74%

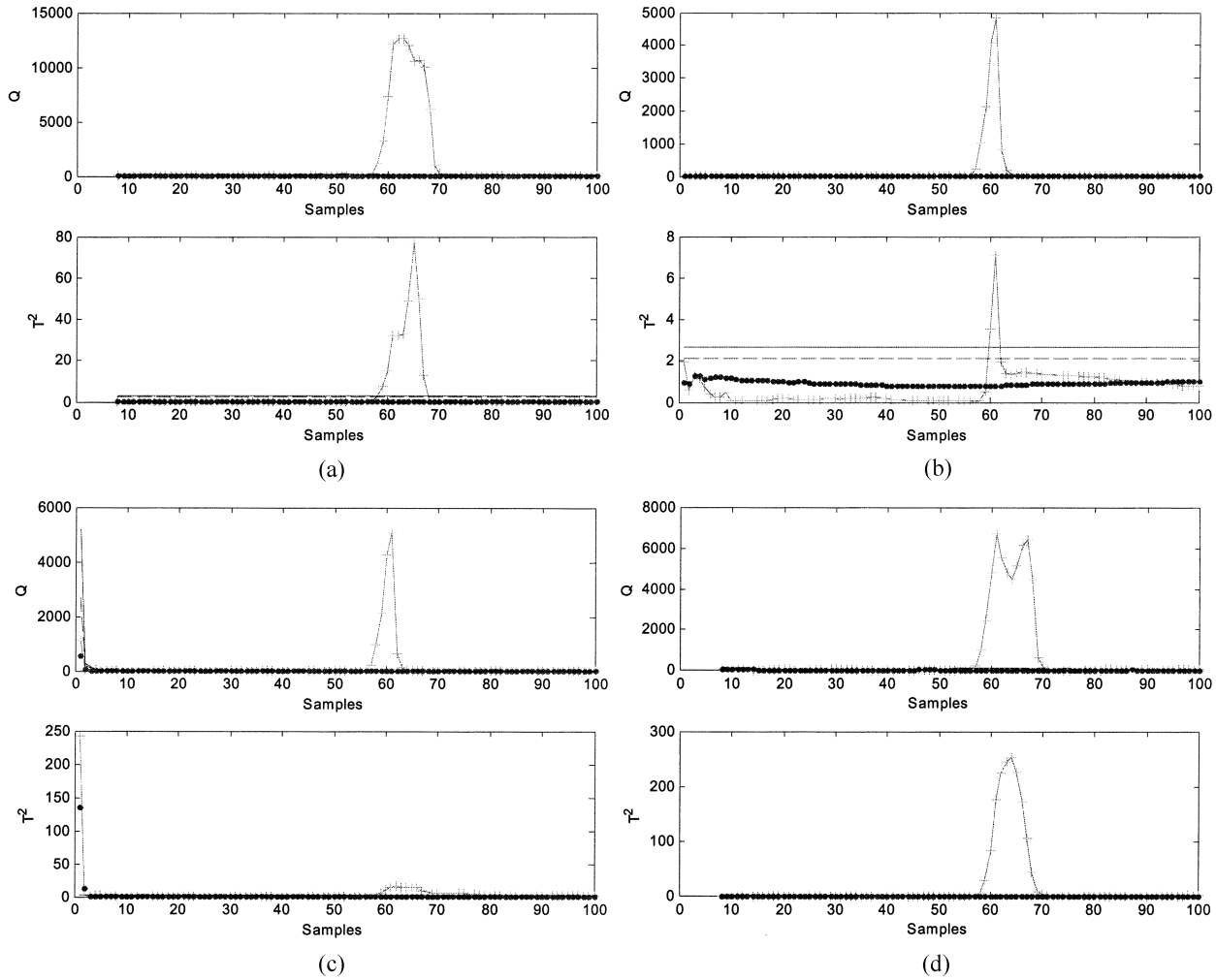

Fig. 7. Q and T^2 control charts for on-line monitoring in Example 1: (a) DPARAFAC; (b) MPCA; (c) PARAFAC; (d) BDPCA. Each chart for the monitoring model contains 95% (dash line) and 99% (solid line) control limits. The solid line with positive signs represents the abnormal batch; the dotted points, the normal one.

of the variation in the process data set. With these 36 batches, the upper control limits (95% and 99%) for the Q-chart and the T^2-chart values are computed for every time point.

The effectiveness of this DPARAFAC monitoring model is tested through abnormal and normal batchs that are not used to construct the model. The abnormal batch is monitored for every time point with the Q-chart and the T^2-chart values. The results are shown in Fig. 7(a). For clear illustration, data around the control limits are zoomed in [Fig. 8(a)]. This abnormal batch immediately drifts away from the normal operating region from the 57th time point. This indicates a special variation occurs from the 57-th time point of the polymerization. With the variation, the batch operating behavior in the model projection plane exceeds the control limits (Q-chart and T^2-chart control limits). Therefore, this batch is assigned as being "out-of-control" or "abnormal". When the normal batch is applied, Q-chart and T^2-chart values of the monitoring point for every time point are always under the control limits. This indicates that it is very unlikely to have any disturbance exist during this batch run; namely, this batch is assigned as being "under-control" or "normal".

The anticipated future observations of the on-line MPCA method are used here. The basic idea is to fill up the empty measurements of the future process variables of the operating batch from the current time to the end with the last deviation from the average trajectories obtained at the current time. This assumes that the future measurements will deviate persistently from their average trajectories at a constant level for the future measured variables during this rest of the batch run. A similar method is also applied to the traditional PARAFAC model, but the score and loading matrices are based on the three-way decomposition. The control charts of the MPCA and PARAFAC models are shown in Figs. 7(b) and 7(c), respectively. Figs. 8(b) and 8(c) are the zoom-in pictures of Figs. 7(b) and 7(c) to clearly distinguish the abnormal condition from the control limits. It is obvious that among DPARAFAC, MPCA and PARAFAC have the same capability to detect the abnormal and normal conditions. This is because some of measurement variables in this abnormal profile are significantly drifted away from the typical normal one around the 57th time point (Fig. 9), particularly measurement variables 5, 6, 9 and 10. Compared with the on-line prediction of the MPCA and the PARAFAC models, DPARAFAC without estimating the future measurements still yields very good results.

2. Example 2: Batch Reactor Benchmark Data

An exothermic chemical batch reactor is used here to make a comparison between the proposed monitoring techniques and the tradi-

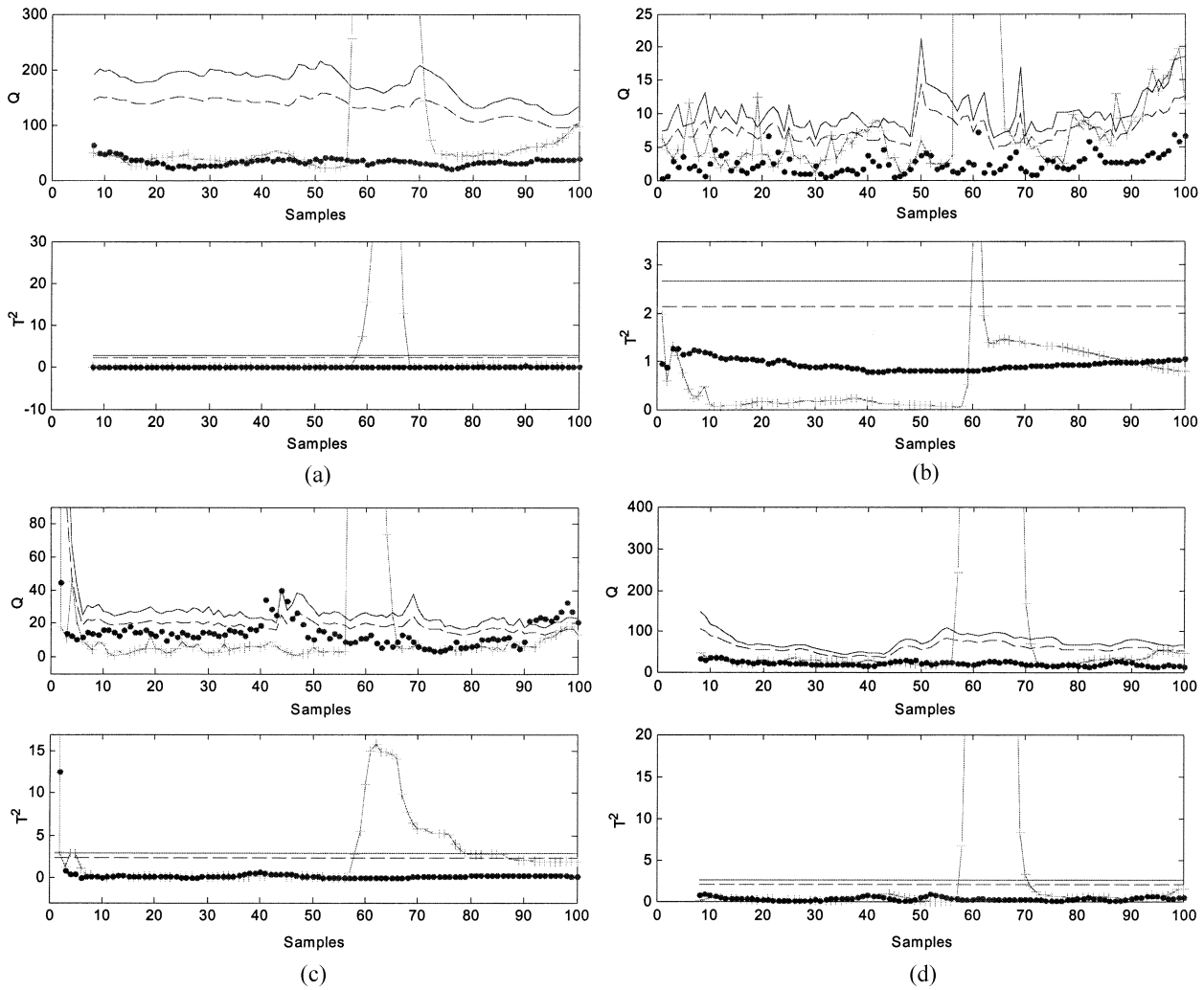

Fig. 8. Zooming in the data of Fig. 7 around the control limits: (a) DPARAFAC; (b) MPCA; (c) PARAFAC; (d) BDPCA.

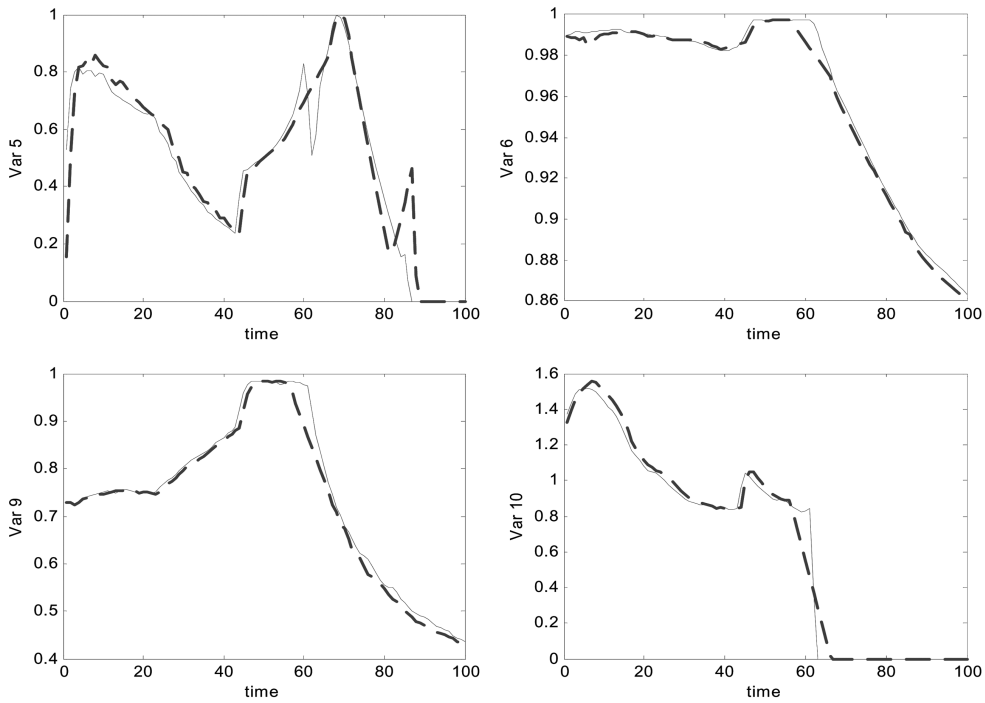

Fig. 9. Trajectories of the measured variables 5, 6, 9 and 10 from a typical normal batch run (solid line) and an abnormal one (dashed line).

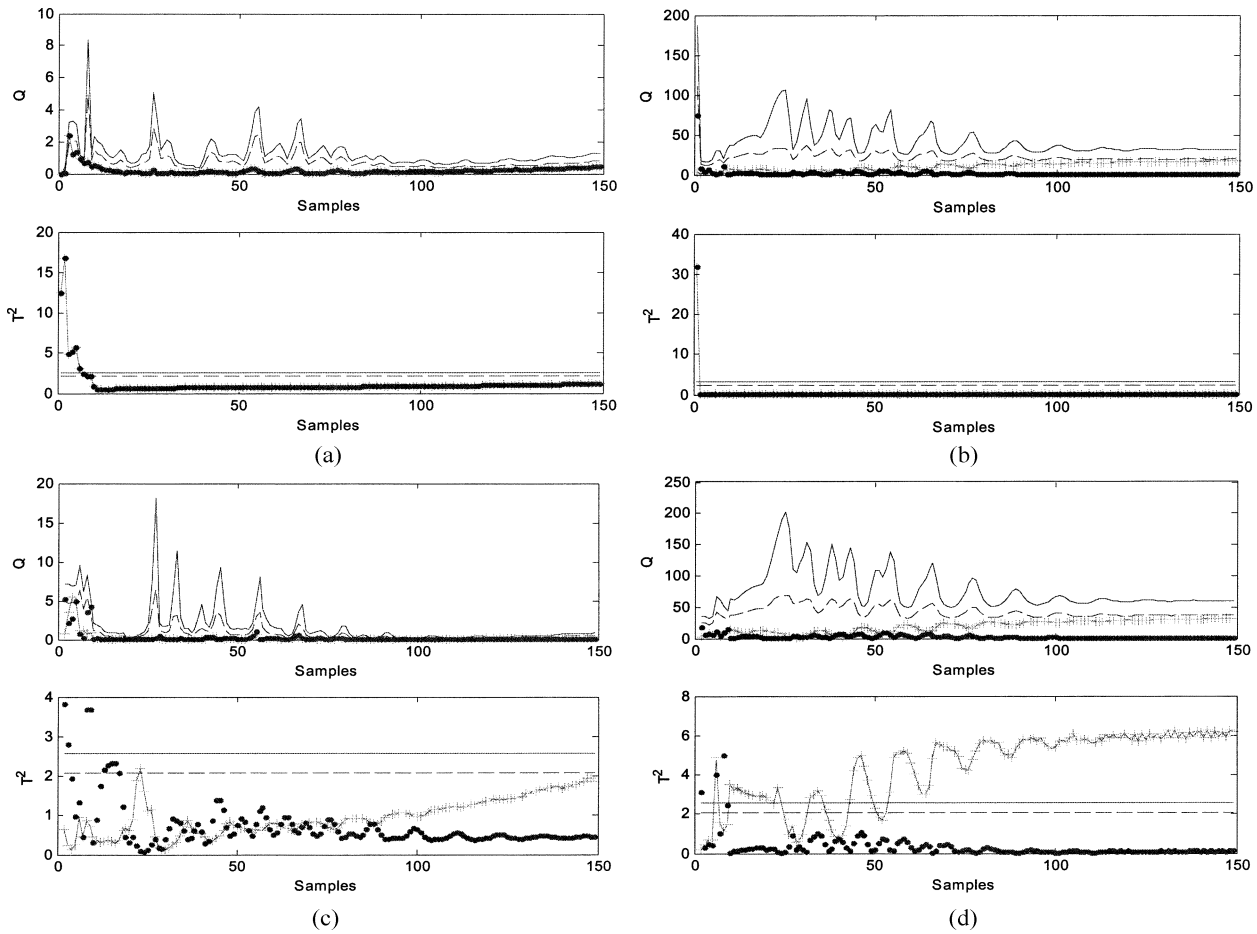

Fig. 10. Q and T^2 control charts for on-line monitoring in Example 2: (a) MPCA; (b) PARAFAC; (c) BDPCA; (d) DPARAFAC. Each chart of the monitoring model contains 95% (dash line) and 99% (solid line) control limits. The solid line with positive signs represents the abnormal condition; the dotted points, the normal one.

tional ones when the quality measurements are, or are not, available. The batch reaction contains two consecutive first-order reactions:

$$A \rightarrow B \rightarrow C$$

Two stages are run in the system. In the first (start-up) stage, the steam in the jacket initially heats the reactor content until the desired operating level. In the second (maintenance) stage, the cooling water in the jacket is used to remove the exothermic heats of reaction. Six process variables are measured in each batch run: the jacket temperature, the temperature of the metal wall between the reactor and the jacket, the reactor temperature, the cooling water flow rate, the position of cooling water control valve and the output signal of the controller that regulates the temperature in the reactor. Two quality variables, concentration of C_B and C_C, are measured at the end of each batch run. The simulation condition and relevant parameters are the same as that of Luyben [1990], except the initial concentration C_A.

A total of 50 batches based on the normal operation are used as the basis analysis. The duration of each batch is 300 minutes: 100 minutes of the start-up time and 200 minutes of the maintenance time. The sampling interval is 2 minutes. In the normal operating condition, there are process variations, such as the initial concentration of A and the inlet cooling water temperature. Two additional batches with good quality and bad quality, respectively, are generated for testing. In the bad quality batch, assume that the initial concentration is shifted a bit to 0.65 lbmol/ft^3 instead of the normal initial concentration, 0.60 lbmol/ft^3.

Different models, depending on the process variables only (MPCA, PARAFAC, BDPCA and DPARFAC), and the combination of the process variables and quality indices (MPLS, Tri-PLS, BDPLS and DTri-PLS), are tested as follows. BDPCA and BDPLS, which are our previous methods [Chen and Liu, 2002] are also the time window-based methods without the future unmeasured batch data, but they are based on a two-way matrix analysis.

(i) Models built with *process variables only*: Four different models, MPCA, PARAFAC, BDPCA and DPARFAC, are used here to make a comparison. The number of components selected via cross-validation for MPCA, PARAFAC, BDPCA and DPARFAC are 4,

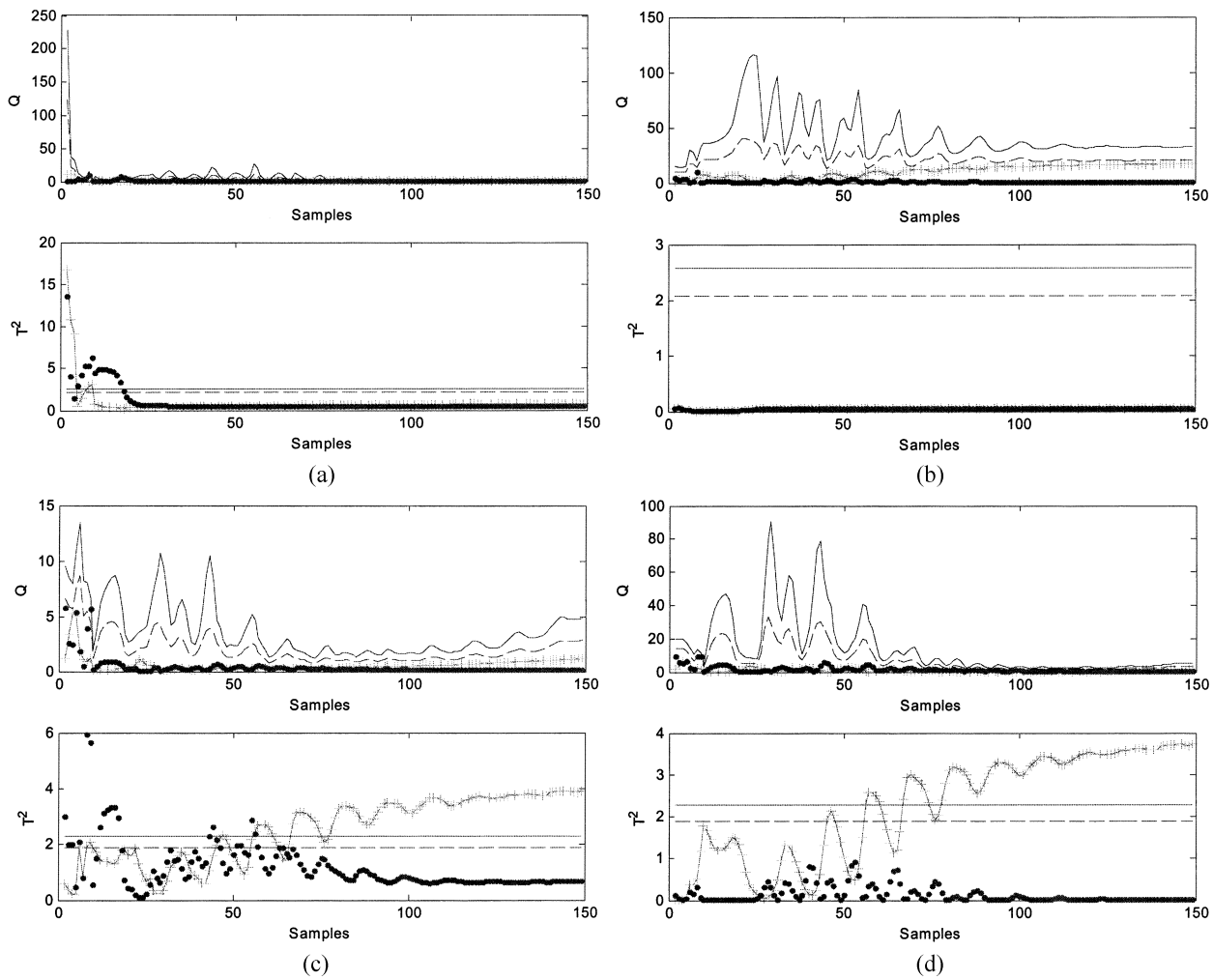

Fig. 11. Q and T^2 control charts for on-line monitoring in Example 2: (a) MPLS; (b) Tri-PLS; (c) BDPLS; (d) Tri-DPLS. Each chart of the monitoring model contains 95% (dash line) and 99% (solid line) control limits. The solid line with positive signs represents the abnormal condition; the dotted points, the normal one.

2, 4 and 4, respectively. The Q and T^2 plots of the four models for the good and bad batches are shown in Fig. 10. No matter whether there are normal or abnormal batches, MPCA and PARAFAC have no apparent outliers during the whole batch run [Figs. 10(a) and 10(b)]. Even if DBPCA falls inside of the confidence limits, the monitoring points of this abnormal batch unusually increase from the 50th time point [Fig. 10(c)]. However, there are some false detections at the normal batch in BDPCA at the initial period of time. Even if the bad batch with the shifted initial concentration is selected for testing, DPARAFAC can significantly detect the occurrence of a small disturbance after the initial period of time [Fig. 10(d)].

(ii) Models built with *process variables* and *quality indices*: Four different models (MPLS, Tri-PLS, BDPLS and DTri-PLS) are used here. The number of components selected via cross-validation for MPLS, Tri-PLS, BDPLS and DTri-PLS is 4, 4, 5 and 4, respectively. The Q and T^2 plots of the four models at the bad batch and the good batch are shown in Fig. 13. Because quality variables are closely correlated with the process variables in the control charts of DBPLS and DTri-PLS at the bad batch [Figs. 11(c) and 11(d)], it is evident that the residuals of the abnormal batches have increased remarkably and fallen outside of the 95% confidence limit after the 60th time point, but there are still some false detections at the normal batch in BDPLS at the initial period of time. For MPLS and Tri-PLS, the monitoring points for the bad batches are not detected during the batch run [Figs. 11(a) and 11(b)]. This indicates that BDPLS and DTri-PLS, which are capable of capturing the dynamic relationship between the process variables and detecting the occurrence of the small disturbance, can perform better than the traditional static methods (MPLS and Tri-PLS). Furthermore, DTri-PLS is more compact than BDPLS because the number of parameters in the three-way mode of DTri-PLS is significantly reduced compared with BDPLS with the two-way mode.

CONCLUSION

In this study, three-way data analysis methods are applied to on-line batch process monitoring through folding time window frames while considering the dynamic characteristics of a batch process in MSPC. The proposed methods involve the serial-correlation effects among the process measurements. They are good for on-line batch monitoring. Based on given process variables or quality indices, two three-way data analysis models, DPARAFAC and DTri-PLS, are separately studied in this paper to improve the performance of the traditional three-way methods, PARAFAC and Tri-PLS.

As the new batch evolves, the measurements for the future time period are unknown until the end of that batch using the existing methods (like MPCA, MPLS, PARAFAC and Tri-PLS) for on-line monitoring. Anticipating the future observations of the rest batches is needed. This may cause false detection because the predicted values without considering the dynamic serial relationship may distort the data information. Therefore, the results of these methods are less favorable when many measurements are unknown in the initial stage of a batch run. Besides, extra computations are needed to estimate the missing measurements from the current time until the end of the batch. In this paper, the proposed methods (DPARAFAC and DTri-PLS) for the on-line monitoring stage do not predict or fulfill the future unmeasured data. They are based on current and previous measured data only. DPARAFAC and DTri-PLS incorporate both series-correlation in one batch and the cross-correlation within the batches. Compared with the existing conventional two-way and three-way models, these proposed methods deliver better results through two benchmark data in monitoring the normal and the abnormal batches.

ACKNOWLEDGMENTS

This work is supported by National Science Council, R.O.C., under the grant NSC-90-2214-E033-010.

REFERENCES

Boqué, R. and Smilde, A. K., "Monitoring and Diagnosing Batch Processes with Multiway Covariance Regression Models," *AIChE J.*, **45**, 1504 (1999).

Box, G. E. P., "Some Theorems on Quadratic Forms Applied in the Study of Analysis of Variance Problems: Effect of Inequality of Variance in One-way Classification," *The Annals of Mathematical Statistics*, **25**, 290 (1954).

Bro, R., "Multiway Calibration Multilinear PLS," *Journal of Chemometrics*, **10**, 47 (1996).

Bro, R., "PARAFAC. Tutorial and Applications," *Chemometrics and Intelligent Laboratory Systems*, **38**, 149 (1997).

Carroll, J. D. and Chang, J. J., "Analysis of Individual Differences in Multidimensional Scaling via a N-way Generalization of Eckart Young Decomposition," *Psychometrika*, **35**, 283 (1970).

Chen, J. and Liu, J., "On-line Piecewise Monitoring for Batch Processes," ADCHEM 2000, Pisa, Italy (2000).

Chen, J. and Liu, K.-C., "On-line Batch Process Monitoring Using Dynamic PCA and Dynamic PLS Models," *Chemical Engineering Science*, **57**, 63 (2002).

Dahl, K. S., Piovoso, M. J. and Kosanovich, K. A., "Translating Third-Order Data Analysis Methods to Chemical Batch Processes," *Chemometrics and Intelligent Laboratory Systems*, **46**, 161 (1999).

Geladi, P., "Analysis of Multi-way (Multi-model) Data," *Chemometrics and Intelligent Laboratory Systems*, **15**, 143 (1989).

Harshman, R. A. and Lundy, M. E., "The PARAFAC Model for Three-way Factor Analysis and Multidimensional Scaling, in Research Method for Multimodel Data Analysis," Law, H. G., Snyder, C. W., Hattie, Jr. J. and McDonald, R. P. Eds., New York, Praeger (1984).

Harshman, R. A., "Foundation of the PARAFAC Procedure: Model and Conditions for an Explanatory Multi-model Factor Analysis," *UCLA Working Paper in Phonetics*, **16**, 1, (1970).

Jackson, J. E., "A Users Guide to Principal Components," Wiley, New York (1991).

Kiers, H. A. L. and Krijnen, W. P., "An Efficient Algorithm for PARAFAC of Three-way Data with Large Numbers of Observation Units," *Psychometrika*, **56**, 147 (1991).

Kruskal, J. B., "Three-way Arrays: Rank and Uniqueness of Trilinear Decomposition with Application to Arithmetic Complexity and Statistics," *Linear Algebra Applicat.*, **18**, 95 (1977).

Ku, W., Storer, R. H. and Georgakis, C., "Disturbance Detection and Isolation by Dynamic Principal Component Analysis," *Chemometrics and Intelligent Laboratory Systems*, **30**, 179 (1995).

Louwerse, D. J. and Smilde, A. K., "Multivariate Statistical Process Con-

trol of Batch Processes Based on Three-way Models," *Chemical Engineering Science*, **55**, 1225 (2000).
Luyben, W. L., "Process Modeling, Simulation and Control for Chemical Engineers," McGraw-Hill, New York (1990).
Nomikos, P. and MacGregor, J. F., "Monitoring Batch Processes Using Multiway Principal Component Analysis," *AIChE J.*, **40**, 1361 (1994).
Nomikos, P. and MacGregor, J. F., "Multivariate SPC Charts for Monitoring Batch Processes," *Technometrics*, **37**, 41 (1995).
Rao, C. R. and Mitra, S., "Generalized Inverse of Matrices and Its Applications," New York, Wiley (1971).
Wachs, A. and Lewin, D. R., "Improved PCA Methods for Process Disturbance and Failure Identification," *AIChE. J.*, **45**, 1688 (1999).
Wise, B. M., Gallagher, N. B., Butler, S. W., White JR., D. D. and Barna, G. G., "A Comparison of Principal Component Analysis Multiway Principal Component Analysis, Trilinear Decomposition and Parallel Factor Analysis for Fault Detection in Semiconductor Etch Process," *Journal of Chemometrics*, **13**, 379 (1999).