Quality-Relevant Fault Diagnosis with Concurrent Phase Partition and Analysis of Relative Changes for Multiphase Batch Processes

Chunhui Zhao
State Key Laboratory of Industrial Control Technology, Dept. of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China

DOI 10.1002/aic.14400
Published online February 18, 2014 in Wiley Online Library (wileyonlinelibrary.com)

Multiplicity of phases as indicated by changes of process characteristics is an inherent nature of many batch processes for both normal and fault cases. To more efficiently perform online fault diagnosis via reconstruction for multiphase batch processes, the phase nature and the relationship between normal and fault cases within each phase should be deeply addressed. This article proposes a quality-relevant fault diagnosis strategy with concurrent phase partition and analysis of relative changes for multiphase batch processes. First, a concurrent phase partition algorithm is developed. The basic idea is to track the changes of process characteristics at normal and fault statuses jointly so that multiple sequential modeling phases are identified simultaneously for both normal and fault cases. Then, the relative changes from the normal status to each fault case are analyzed in each phase to reveal the specific fault effects more efficiently. The fault effects are decomposed in two different monitoring subspaces, principal subspace, and residual subspace, by capturing their different roles in removing out-of-control signals. The significant increases relative to the normal case are judged to be responsible for the concerned alarm monitoring statistics in each phase. The others are composed of general variations that are deemed to still follow normal rules and thus insignificant to remove alarm monitoring statistics. Those alarm-responsible fault deviations are then used to develop reconstruction models which can more efficiently recover the fault-free part for online fault diagnosis. The proposed algorithm is illustrated with a typical multiphase batch process with one normal case and three fault cases. © 2014 American Institute of Chemical Engineers *AIChE J*, 60: 2048–2062, 2014

Keywords: analysis of relative change, concurrent phase partition, multiphase batch processes, reconstruction modeling, fault diagnosis

Introduction

A lot of research work has been done in recent years for online process monitoring and fault diagnosis[1] which have become significant for industrial processes. Chemical process plant safety, quality improvement, operational constraints and plant economics and so forth are some of the main drives. The related methodologies have attracted more and more attentions from engineers and researchers. Typically, data driven models from principal component analysis (PCA)[2] or partial least squares (PLS)[3–5] are built to decompose process variation information and develop monitoring charts. The tasks involved in statistical process monitoring (SPM)[6] typically include fault detection, fault estimation, fault identification and diagnosis and so forth. There has been tremendous interest in diagnosing the possible root causes of a fault situation. Different methods[7–15] have been reported for fault diagnosis based on historical measurement data. In the field of multivariate SPM, the method of contribution plots[16–18] has been widely used for years to isolate the root faulty variables based on the assumption that the out-of-control monitoring statistics are contributed significantly by those faulty variables. This approach does not need *a priori* fault knowledge except for a normal statistical monitoring model, which, however, may result in misleading results because of the "smearing" effect.[13,16,19] Although there is no fundamental analysis on the conditions of correct fault identification using contributions, Westerhuis et al.[16] have discussed that a fault in one variable can smear to contributions of other variables in traditional contribution plots. If the actual fault directions can be obtained, the fault can be diagnosed without ambiguity, including recovering fault-free data and estimating the fault magnitude. In the context of PCA models, the concept of fault reconstruction was defined in the work of Dunia and Qin[20] where faults were characterized by fault directions or subspaces to estimate the fault-free part of the measurement data. The procedure to restore normal conditions by applying a corrective action in the data is called data reconstruction, and the procedure for identifying a fault by reconstruction for a given type of faults is called fault diagnosis via reconstruction. In their work, the fault detectability, identifiability, and reconstructability were

Correspondence concerning this article should be addressed to C. Zhao at chhzhao@zju.edu.cn.

addressed regarding the applicability of reconstruction method. Also, the method for extraction of fault feature subspace or directions was proposed by performing singular value decomposition on the averaged historical fault data.[14] In general, by modeling on fault data, more specific fault characteristics can be captured than the case with no fault data, providing more information for fault diagnosis.

Since 1994, most batch process monitoring methods are based on multiway principle component analysis (MPCA) and multiway PLS.[21–31] However, although they have been well developed and applied widely over the last two decades, there are still some problems. Batch process operation is in general carried out in a sequence of discrete steps, such as polymer production, fermentation, semiconductor etching and so forth. Events taking place in different steps have different characteristics and impose different impacts on the final product yield and quality. However, conventional multiway methods are not easily able to reveal the changes of process characteristics across phases because they take the entire batch data as a single object by unfolding the data throughout the process. Also, it may be difficult for online application as the new batch data are not complete up to the concerned time and the unknown future values have to be estimated. As defined by Undey and Cinar,[30] steps occurring in a single processing unit as a succession of events caused by operational or phenomenological (chemical reactions, microbial activities, etc.) regimes are called phases; steps occurring in different processing units and performing different unit operations are called stages. Moreover, Lu et al.[32] defined modeling phases according to the changes of underlying process characteristics. In the present work, the phase definition by Lu et al. is used. A batch process may be divided into several modeling phases where despite that the process trajectories may be time-varying, the variable correlations stay largely similar within the same phase,[32] revealing the same impact on product quality. Therefore, changes of variable correlations may be used to indicate changes of phases. In fact, the multiplicity of phases is an inherent nature of many batch processes for both normal and fault cases. Considering different phases exhibit significantly different underlying behaviors, it is desirable to develop multiple phase-based models. Then, each model represents a specific phase and explains the local process behaviors, which can effectively enhance process understanding and improve monitoring reliability.[30] Kosanovich et al.[33] and Dong and McAvoy[34] developed two MPCA/nonlinear MPCA models to analyze the phase-specific nature of a two-phase jacketed exothermic batch chemical reactor, where monitoring results show that the two phase-based models are more powerful than a single model. Their phase models, however, inherit the common weakness of the conventional MPCA model that the unavailable future data in an evolving batch should be estimated for online monitoring. Subphase modeling methods[32,35] have been well developed and applied, which can preserve the dynamic process characteristics and do not require fulfilling missing process observations for online application. The process time is used as an indicator to adopt the corresponding phase model for online monitoring. To get the phase information, many phase partition methods[18,32–37] have been reported for batch processes, which separate phases based on different principles. Assuming that no *a priori* process knowledge is available, the phase information can be identified based on the changes of underlying characteristics. A clustering algorithm[32,35] was developed to collect similar process characteristics based on evaluation of distance similarity. However, the clustering algorithm does not consider the time sequence of operation phases. Time-slices from different time regions may be mixed together and cause confusion for phase understanding, which thus, requires a heavy postprocessing. To overcome this problem, a sequential step-wise phase partition (SSPP) algorithm[38] was proposed which can get multiple phases in time sequence and guarantee similar characteristics within the same phase. However, previous phase partition strategies only consider a single mode (i.e., only normal processes). Changes of process characteristics along time direction in fault cases (i.e., phase nature of fault processes) have not been addressed yet. As mentioned before, multiplicity of phases as indicated by changes of process characteristics is an inherent nature of many batch processes for both normal and fault cases. In comparison with the phases in normal processes, the process characteristics in fault cases may be more complex. For multiphase batch processes, after some abnormalities happen, the similar process characteristics within the same phase in the normal case may be changed under the influences of disturbances. For example, one phase in the normal case may be further separated into several modeling phases in some fault case so that each modeling phase only covers similar fault characteristics while different phases represent different fault characteristics. For the specific purpose of fault diagnosis, the fault process information should be effectively analyzed. By analysis of phase nature in fault processes, it can reveal how process characteristics change under the influences of some disturbances. How to consider multiplicity of fault characteristics and how to analyze the fault effects relative to the normal case are important questions, which, however, have been addressed yet.

For more efficient online quality-relevant fault diagnosis in multiphase batch processes, this article proposes a phase-based fault reconstruction modeling strategy. It includes two key features: automatic concurrent phase partition from an integrated and parallel perspective for the normal case and fault cases; and comprehensive fault subspace decomposition by analyzing relative changes from normal to fault and checking the faults influences on monitoring results. The objective of concurrent phase partition is to simultaneously identify different phases in time sequence for both normal and fault cases, which can guarantee similar characteristics within the same phase and also provide the basis for the following phase-based analysis of relative changes. The objective of analysis of relative changes is to extract those significant quality-relevant fault effects that have caused out-of-control monitoring statistics. The fault variations with significant increases relative to the normal case are deemed to be responsible for alarms and should be used for the development of reconstruction models. The feasibility and performance of the proposed algorithm is illustrated by a typical multiphase batch process with one normal case and three fault cases.

The rest of this article is organized as follows. First, PLS-based fault detection strategy is preliminarily revisited. Then, the proposed fault diagnosis methodology is presented for multiphase batch processes on the basis of PLS monitoring system. It develops the algorithms of concurrent phase division and analysis of relative changes for reconstruction modeling and online fault diagnosis. In Illustrations Section, its applications to injection molding (IM), a multiphase batch process, are presented, suggesting its feasibility and efficacy for online fault diagnosis. Finally, conclusions are drawn in the last section.

Review of PLS Monitoring Method

Regression modeling methods,[3–5,39,40] such as PLS, canonical correlation analysis, and some variations on these methods, have been used in multivariate SPM in a way similar to PCA-based methods. It is deemed that PCA-related algorithm can be used to decompose the general process variations and detect abnormal situations in process variables (**X**). Comparatively, PLS-related algorithm can supervise the variations in process variables that are more influential on quality variables (**Y**) by decomposing **X** in a different way.

By PLS, predictor latent variables (LVs) (**T**) which represent quality-relevant process variations are calculated by

$$\mathbf{T}=\mathbf{XR} \tag{1}$$

where **R** is the regression weights to directly calculate LVs from **X**. The detailed calculation procedure can refer to previous work.[3–5]

Call the regression of **X** onto the extracted A predictor LVs (**T**)

$$\begin{gathered}\mathbf{P}^{\mathrm{T}}=\left(\mathbf{T}^{\mathrm{T}}\mathbf{T}\right)^{-1}\mathbf{T}^{\mathrm{T}}\mathbf{X}\\ \mathbf{X}=\mathbf{TP}^{\mathrm{T}}+\mathbf{E}\end{gathered} \tag{2}$$

where $\mathbf{P}(J_x\times A)$ are the loadings for **X**, which tell how **T** contribute to the variations in **X**. $\mathbf{E}(N\times J_x)$ are the residuals of **X** after the explanation of **T**.

By the PLS algorithm, **X** are decomposed into one low-dimensional systematic subspace (**T**) and one residual subspace (RS) (**E**). Based on the decomposition in Eq. 2, one T^2-statistic for the systematic subspace and one squared predicted error (SPE)-statistic for the RS are calculated. Assuming the process variables follow a multivariate normal distribution, control limit for T^2 can be obtained using the F-distribution with α as the significance factor[41]

$$T^2\sim\frac{A(N^2-1)}{N(N-A)}F_{A,N-A,\alpha} \tag{3}$$

Similarly, in the RS, the confidence limit of SPE can be approximated by a weighted Chi-squared distribution (χ)[22]

$$\mathrm{SPE}\sim g\chi_{h,\alpha}{}^2 \tag{4}$$

where $g=v/2m$ and $h=2m^2/v$, in which m is the average of all SPE values calculated in Eq. 4, and v is the corresponding variance.

Methodology

In this section, the proposed fault diagnosis methodology is presented for multiphase batch processes. The flow diagram of the proposed method is illustrated in Figure 1 regarding model development as shown in Figure 1a and online application as shown in Figure 1b. First, the fault processes are associated with normal processes for concurrent phase division which provides a good platform for the following phase-based relative analysis and reconstruction modeling. Then, to better interpret the fault deviations and more efficiently recover the fault-free part, the relative changes from normal to fault are analyzed in each specific phase where the fault effects responsible for the out-of-control monitoring statistics are modeled for fault correction. Based on the above modeling procedure, online fault diagnosis via reconstruction is performed to isolate the fault causes. Two important issues are addressed, referring to checking the changes of quality-relevant process characteristics simultaneously for normal and fault cases (i.e., concurrent phase analysis) and analyzing the difference of process variations between normal and fault cases (i.e., analysis of relative changes). The specific implementation procedure is presented in the following subsections.

Concurrent phase partition

Considering the multiplicity of phases in batch processes, phase division is an important issue for batch process modeling. Automatic clustering-based phase division and phase-based modeling methods[32,35] have been proposed and well applied for online process monitoring. For readability, the clustering-based algorithm is briefly revisited in Appendix A. However, the clustering result requires a heavy and complicated postprocessing[38] to get the final phases. Moreover, the clustering result is greatly influenced by the way to evaluate similarity, which, however, does not consider the specific purpose of process monitoring. To overcome the aforementioned problems, a SSPP[38] algorithm was proposed to detect the phase landmarks in time sequence from the process beginning step by step. It addresses the phase nature by tracking the influences of time-varying process characteristics on the performance of monitoring models. For readability, the SSPP algorithm is briefly revisited in Appendix B. Unfortunately, both algorithms only consider the changes of process characteristics in the normal case for the purpose of process monitoring. Based on previous analyses, as the fault deviations relative to normal case are more important for reconstruction-based fault diagnosis, phases should be analyzed simultaneously for normal and fault cases by checking the changes of process characteristics.

In each batch run (batch index $i=1,2,\ldots,I$), assume that J process variables are measured online at $k=1,2,\ldots,K$ time instances throughout the operation cycle, forming each regular batch dataset, denoted as $\mathbf{X}(K\times J)$. Therefore, for the normal case and each fault case, a three-way data array is available by collecting data from I batches, $\underline{\mathbf{X}}_m(I_m\times J\times K)$ ($m=1,2,\ldots,M$). There are M cases in all, including one normal case and $M-1$ fault cases. Besides, quality data $\mathbf{Y}_m\left(I_m\times J_y\right)$ are correspondingly collected for J_y quality variables by end-of-batch measurement. So for each case, the process and quality data pairs are obtained, $\{\underline{\mathbf{X}}_m,\mathbf{Y}_m\}$.

The basic analysis unit of quality-relevant concurrent phase division is time-slice process-quality data pair in each case. The basic idea is to simultaneously find the phase landmarks for all cases from the quality-relevant perspective to make sure that processes enter and leave each phase at the same time. To achieve this purpose, three rules are considered for automatic phase division, including the similar quality-relevant characteristics along time direction within each case, the time sequence, and the concurrent consideration of all cases. In one word, the phase landmarks are detected by simultaneously and sequentially checking the changes of quality-relevant variable correlations from the process beginning step by step for all cases which is thus termed quality-relevant concurrent step-wise sequential phase partition (QCSSPP) algorithm here.

Prepare the time-slice process-quality data pair $\{\mathbf{X}_{m,k}(I\times J),\ \mathbf{Y}_m\left(I_m\times J_y\right)\}$ ($k=1,2,\ldots,K$) from the process beginning of each case. The basic procedure is described as follows:

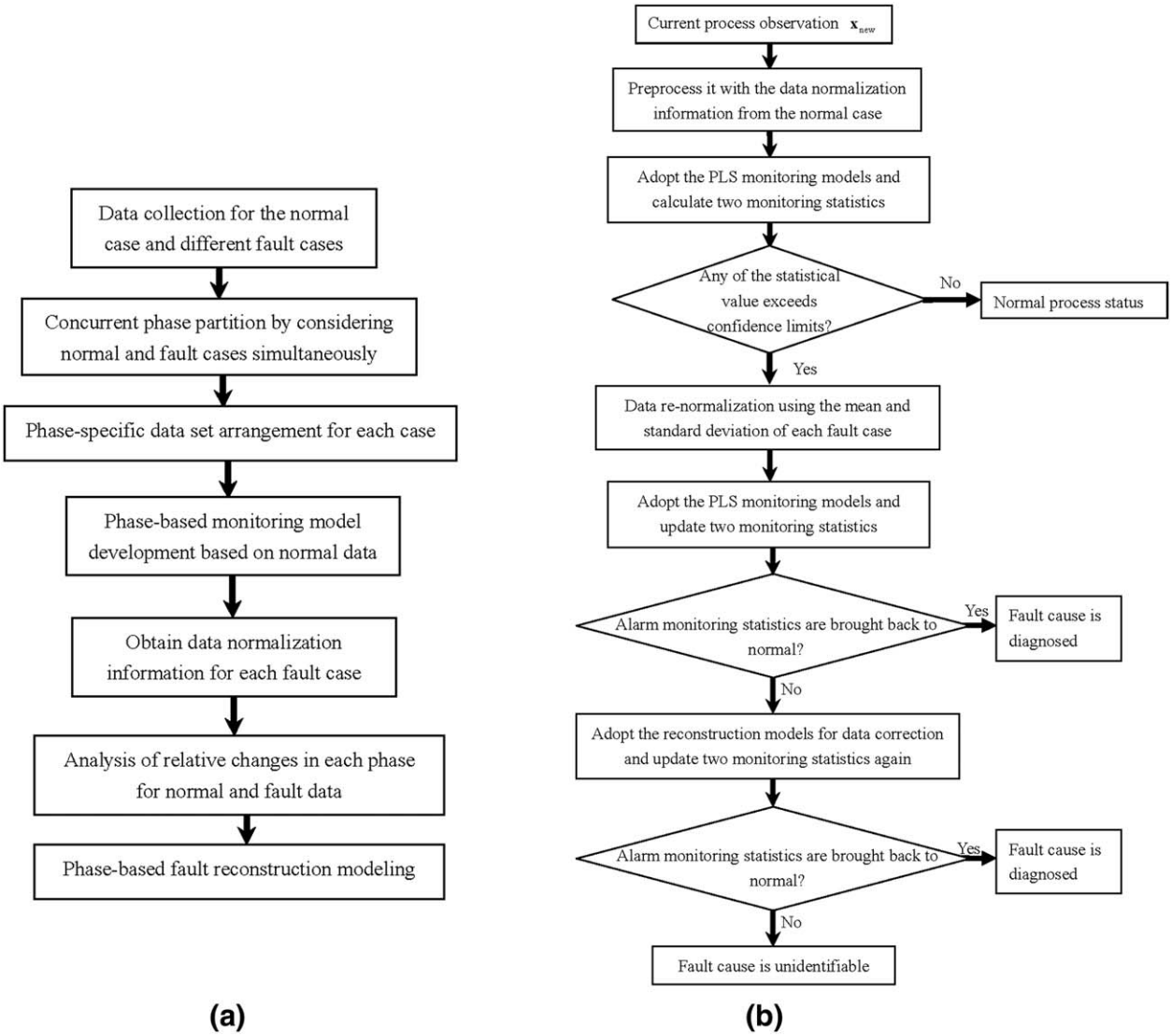

Figure 1. Flow chart of the proposed algorithm (a) model development and (b) online fault diagnosis procedure.

Step 1: Multicase Data Preparation. Input the normalized time-slice regression data pair $\{\bar{\mathbf{X}}_{m,k}, \bar{\mathbf{Y}}_m\}$ ($k = 1, 2, \ldots, K$) from each case which are preprocessed to have zero mean and standard deviation, respectively.

Step 2: Time-Slice-Based PLS Modeling. Perform PLS algorithm on the time-slice regression data pairs and get the initial time-slice PLS models, including weights $\mathbf{R}_{m,k}$ $\left(J \times R_{m,k}\right)$ and loadings $\mathbf{P}_{m,k}\left(J \times R_{m,k}\right)$ ($k = 1, 2, \ldots, K$). The number of LVs at each time, $R_{m,k}$, is determined by prediction residual sum of squares (PRESS)[5] against the number of LVs for good prediction performance. Then, find the number of LVs that occurs most frequently throughout the batch process and accordingly unify it as the dimension of time-slice PLS models for each case, R_m. Use the updated time-slice PLS models with the unified dimension for quality prediction. Re-evaluate the prediction results using mean squared errors (MSE) index at each time k for the concerned quality indices, which is calculated as

$$\mathrm{MSE}_{m,k} = \frac{1}{I}\sum_{i=1}^{I} |\mathbf{y}_{m,i} - \hat{\mathbf{y}}_{m,k,i}|^2 \tag{5}$$

where $\mathbf{y}_{m,i}\left(J_y \times 1\right)$ are the real values of J_y quality indices of the ith batch in Case m, and $\hat{\mathbf{y}}_{m,k,i}$ are the prediction values at the kth time in Case m. The index, $\mathrm{MSE}_{m,k}$, represents the prediction power of time-slice PLS models to J_y quality indices.

Step 3: Time-Segment-Based PLS Modeling. From the beginning of batch processes in each case, add the next time-slice to the existing ones and variable-unfold them, $\mathbf{X}^v_{m,t}(It \times J)$(where subscript t denotes the current time). Also, the quality data $\mathbf{Y}^v_{m,t}(It \times J)$ are arranged in a similar way by duplicating $\bar{\mathbf{Y}}_m$ t times so as to have the same row dimension with that of $\mathbf{X}^v_{m,t}(It \times J)$. They prepare the variable-unfolding regression data pair $\{\mathbf{X}^v_{m,t}(It \times J), \mathbf{Y}^v_{m,t}(It \times J)\}$. Perform PLS on the rearranged data matrix and get the new PLS weight model, $\mathbf{R}^v_{m,t}(J \times R_m)$, which is called time-segment PLS model. Evaluate the prediction results using MSE at each time k up to t

$$\mathrm{MSE}^v_{m,k} = \frac{1}{I}\sum_{i=1}^{I} |\mathbf{y}_{m,i} - \hat{\mathbf{y}}^v_{m,k,i}|^2 \tag{6}$$

where $\hat{\mathbf{y}}^v_{m,k,i}$ are the prediction values at the kth time in Case m by the current time-segment PLS model. The index,

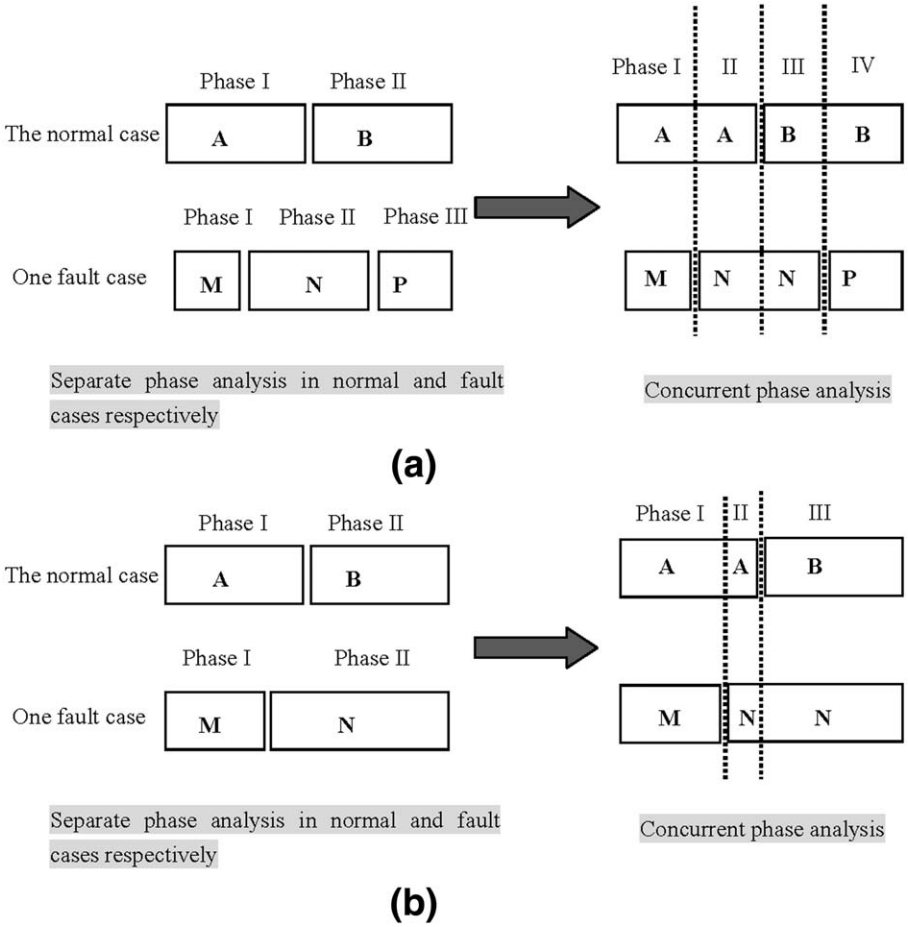

Figure 2. Simple illustration of quality-relevant concurrent phase division for batch processes of the normal case and one fault case.

$\mathrm{MSE}^{v}_{m,k}$, represents the prediction power of time-segment PLS models to each time-slice up to t in each case.

Step 4: Compare Model Accuracy. Compare $\mathrm{MSE}^{v}_{m,k}$ with $\mathrm{MSE}_{m,k}$ for each time slice within the concerned time region for each case. If more than three consecutive samples show $\mathrm{MSE}^{v}_{m,k} > \alpha_m * \mathrm{MSE}_{m,k}$ for some case, it means the addition of the current time-slice have imposed great influences on the time-segment PLS model and the resulting prediction performance. The time slices before t are denoted as one phase for all cases. α_m is a constant attached to $\mathrm{MSE}_{m,k}$, termed relaxing factor here, which determines how much the time-segment PLS model in the mth case is allowed to be less representative (i.e., insufficient predictive power) than the same-dimensional time-slice PLS models within the concerned time region.

Step 5: Data Updating and Recursive Implementation. For all cases, remove the time-slice data matrices of the first phase and the left batch process data are now used as the new input in Step 3. Recursively repeat Steps 3–4 based on the updated process beginning for each case to find the following phases.

From the above procedure, to simultaneously identify phases for the normal case and all fault cases, variable-unfolding modeling unit is arranged iteratively by adding new time-slices sequentially and the changes of the resulting model's predictive power are checked by prediction errors. The output is a simultaneous partition of process trajectory along time direction for all cases. A simple illustration is shown in Figure 2 for two-case batch processes with equal batch length to explain the concurrent phase division.

In Figure 2a, for separate phase analysis, the normal case shows two different kinds of process characteristics (A and B) corresponding to two phases, whereas the fault case covers three kinds of process characteristics (M, N, and P) corresponding to three phases. In particular, the process characteristics change differently under the influences of abnormalities in comparison with those in normal case. For one specific time slice measured at the kth time, it may lie in a different phase in the normal case and fault case, respectively. For example, the process characteristics within Phase I in the normal case are changed under the influences of disturbances so that the corresponding time region in fault case in fact covers two different kinds of fault characteristics (M and N). Also, the process characteristics within the later Phase I and the early Phase II in the normal case show similarity after the influences of disturbances so that the corresponding time region in fact only reveals one type of fault characteristics (N) in the fault case. By the concurrent phase partition instead of separate phase analysis, four phases are obtained by simultaneously checking the changes of process characteristics in both normal and fault cases.

In Figure 2b, another illustration is shown to explain the idea of concurrent phase analysis. Both the normal and fault cases cover two phases (i.e., two different types of characteristics) but with different phase spans. By the concurrent phase analysis, three phases are available for both cases.

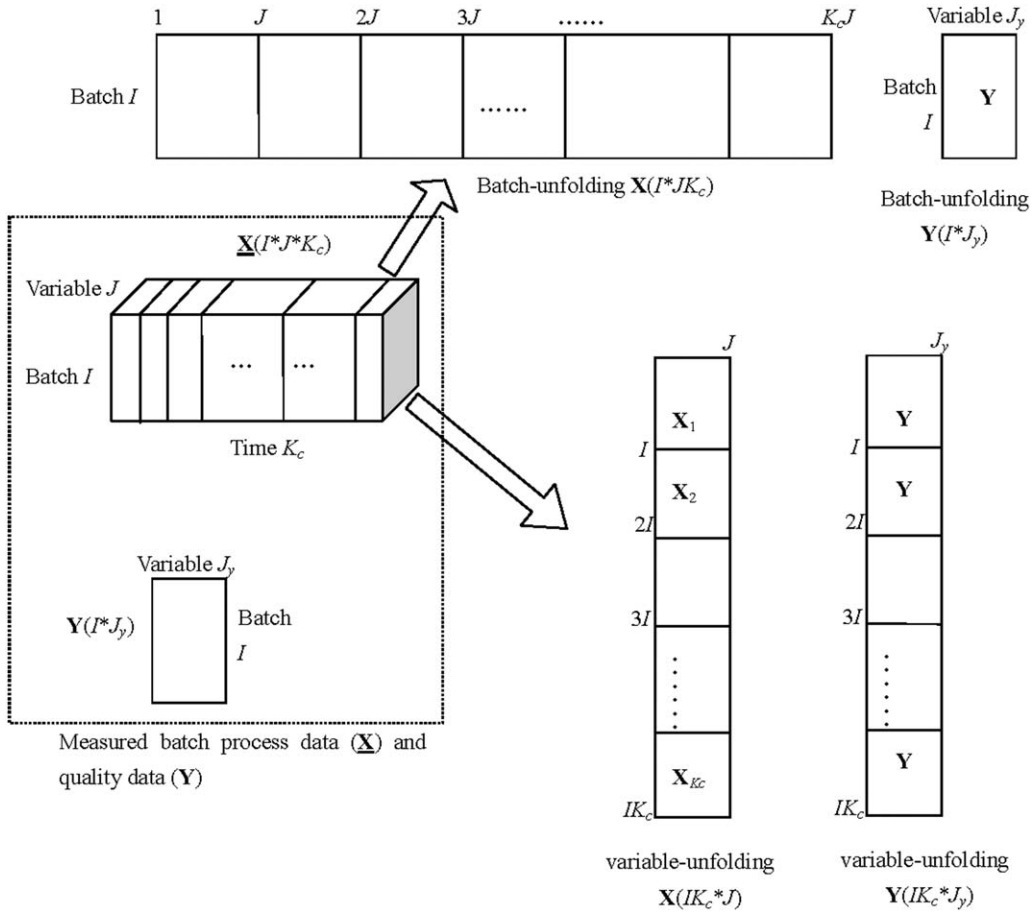

Figure 3. Illustration of batch-unfolding and variable-unfolding batch data arrangement for regression modeling.

Fault characteristics N are separated into two phases which will be described separately because of the changes from process characteristics A to B in the normal case.

In this way, similar process characteristics are guaranteed within the same phase for the normal case and each fault case, respectively. And, the changes of process characteristics in any case can be reflected by the switch from one phase to another. Also, it is noted that the process characteristics are not distorted as no change or transformation is made on the process trajectories. The specific process time can be used as an indicator to judge which phase the process has arrived at for both normal and fault cases. Then, the relative changes from normal to fault can be analyzed in the following subsection to reveal the fault effects within the specific phase for reconstruction modeling.

It is noted that the divided phases in the present work are in fact "modeling phases." They may not, and do not have to, have the exact correspondence to the physical operations of batch processes. If the process characteristics within the physical operation phase have changed, a physical operation phase can be further divided into multiple modeling phases. *A priori* process knowledge can also help to check and understand the phase division results. However, if no process knowledge is known, the phase division results can be checked by trial and error regarding the specific study purpose (fault diagnosis here). Therefore, for different batch processes or different study purposes, the phase division results may be different. α_m is an important parameter to adjust phase division results. A larger α_m value means a larger reduction of model accuracy is allowed, resulting in fewer phases; a smaller α_m value means a more accurate model is needed to describe the time slices, resulting in more phases. Its influences have been discussed in previous work[38] regarding single-case phase division. Here for simplicity, it is not addressed here.

Phase-based analysis of relative changes

After phase partition, the relationship between normal and fault cases can be analyzed so that the fault effects can be better understood. From the perspective of fault detection, the fault deviations responsible for the out-of-control monitoring statistics are in fact the concerned fault effects for data correction. The relative changes from normal to fault represent how the fault influences the normal status and disturbs the monitoring results which thus should be analyzed for the development of reconstruction models in each phase. The normal process status is the reference case. In comparison with the reference case, each alternative fault case is decomposed based on whether the fault variations increase in the PLS monitoring systematic subspace and RS, respectively. It is based on the fact that the increased relative variations in each fault case are responsible for alarm signals.

As shown in Figure 3, two phase-representative data sets are prepared using variable-unfolding in each phase for the normal case and one fault case, $\mathbf{X}_{c,r}^{v}(I_rK_c \times J)$ and $\mathbf{X}_{c,a}^{v}(I_aK_c \times J)$(where subscript r means the reference case

and a denotes one alternative fault case; c indicates phase, and superscript v means variable-unfolding is used for data arrangement; K_c denotes the phase duration; I_r and I_a are the number of batches in the reference normal case and the alternative fault case, respectively), each being composed of the same number of variables and maybe different number of samples. By phase partition in the previous subsection, for the normal case and each fault case, similar quality-related process characteristics have been collected in the same phase and different characteristics are separated into different phases. That is, the similar characteristics within the same phase can be described by the same phase model, revealing similar effects on the final product quality. Therefore, variable-wise unfolding is used in each phase to arrange the phase data for both normal case and fault case. The time-slices in $\mathbf{X}^v_{c,r}$ and $\mathbf{X}^v_{c,a}$ have been centered and scaled to have zero mean and unit standard deviation, respectively. The specific modeling procedure is described as follows.

In systematic subspace of PLS monitoring system:

Step (1). Perform PLS algorithm on $\{\mathbf{X}^v_{c,r}, \mathbf{Y}^v_{c,r}\}$ to get the monitoring system

$$\begin{cases} \mathbf{T}^v_{c,r}=\mathbf{X}^v_{c,r}\mathbf{R}_{c,r} \\ \hat{\mathbf{X}}^v_{c,r}=\mathbf{X}^v_{c,r}\mathbf{R}_{c,r}\mathbf{P}_{c,r}{}^{\mathrm{T}} \\ \mathbf{E}^v_{c,r}=\mathbf{X}^v_{c,r}\mathbf{R}^e_{c,r}\mathbf{P}^e_{c,r}{}^{\mathrm{T}} \end{cases} \tag{7}$$

where $\mathbf{T}^v_{c,r}(I_rK_{c,r}\times R_{c,r})$ and $\mathbf{R}_{c,r}(J\times R_{c,r})$ are LVs and the corresponding weights in the systematic subspace; also, the loadings $\mathbf{P}_{c,r}(J\times R_{c,r})$ can be calculated by $\mathbf{P}_{c,r}{}^T=(\mathbf{T}^v_{c,r}{}^T\mathbf{T}^v_{c,r})^{-1}\mathbf{T}^v_{c,r}{}^T\mathbf{X}^v_{c,r}$. They are the first distribution directions with the largest variation variance in the systematic subspace which are also the monitoring directions of T^2-statistic. $R_{c,r}$ is the number of retained PLS LVs determined by PRESS index.[5] Although it is possible to calculate as many systematic LVs as the rank of $\mathbf{X}$ matrix, not all of them are normally used. $\mathbf{E}^v_{c,r}(I_rK_{c,r}\times J)$, $\mathbf{R}^e_{c,r}(J\times R^e_{c,r})$, and $\mathbf{P}^e_{c,r}(J\times R^e_{c,r})$ are PLS residuals, the corresponding residual weights, and residual loadings, respectively. $R^e_{c,r}$ is the number of monitoring directions left in the PLS RS after the extraction of systematic LVs, $R^e_{c,r}=\mathrm{rank}(\mathbf{X}^v_{c,r})-R_{c,r}$.

Step (2). Project $\mathbf{X}^v_{c,a}$ onto $\mathbf{R}_{c,r}$ to get LVs $\mathbf{T}^v_{c,a}=\mathbf{X}^v_{c,a}\mathbf{R}_{c,r}$, that is, the reference model $\mathbf{R}_{c,r}$ is used to monitor the alternative fault case. Define the ratio of variation between the alternative fault case and the reference normal case in the systematic subspace as

$$\mathrm{Ratio}_{c,a,i}=\frac{\mathrm{var}(\mathbf{T}^v_{c,a}(:,i))}{\mathrm{var}(\mathbf{T}^v_{c,r}(:,i))}\,(i=1,2,...,R_{c,r}) \tag{8}$$

where var($\square$) denotes the score variance around the center of reference data (zero here) and $(:,i)$ denotes the ith column vector in a matrix. So, $\mathrm{Ratio}_{c,a}$ is a $R_{c,r}$-dimensional vector composed of $\mathrm{Ratio}_{c,a,i}$.

Step (3). Sort the values of $\mathrm{Ratio}_{c,a}$ index. If the $\mathrm{Ratio}_{c,a,i}$ value is larger than one, it means that the corresponding variation in the alternative fault case is larger than that in the reference normal case, which thus, is responsible for out-of-control T^2. Keep the directions corresponding to $\mathrm{Ratio}_{c,a,i}$ values of larger than one, we can get $\mathbf{R}_{c,a}{}^*(J\times R_{c,a}{}^*)$, and the number of retained directions is $R_{c,a}{}^*$. $\mathbf{R}_{c,a}{}^n$ are composed of the left directions in $\mathbf{R}_{c,r}(J\times R_{c,r})$, along which variations in the alternative fault case stay similar in comparison with those in the reference normal case.

Step (4). The process variations explained along $\mathbf{R}_{c,a}{}^*$ can be modeled as

$$\hat{\mathbf{X}}_{c,a,f}=\mathbf{X}^v_{c,a}\mathbf{R}_{c,a}{}^*\mathbf{P}_{c,a}{}^{*\mathrm{T}}=\mathbf{T}^v_{c,a}{}^*\mathbf{P}_{c,a}{}^{*\mathrm{T}} \tag{9}$$

where $\mathbf{T}^v_{c,a}{}^*$ are LVs separated from $\mathbf{T}^v_{c,a}$; $\mathbf{P}_{c,a}{}^{*\mathrm{T}}$ are the corresponding loadings, which are calculated as $(\mathbf{T}^v_{c,a}{}^{*\mathrm{T}}\mathbf{T}^v_{c,a}{}^*)^{-1}\mathbf{T}^v_{c,a}{}^{*\mathrm{T}}\mathbf{X}^v_{c,a}$. Clearly, they make contribution to out-of-control T^2 monitoring statistic in the alternative fault case. PCA is then performed on $\hat{\mathbf{X}}_{c,a,f}$ to extract the major systematic variations in order with $R_{c,a,f}$ components

$$\hat{\hat{\mathbf{X}}}_{c,a,f}=\hat{\mathbf{X}}_{c,a,f}\mathbf{P}_{c,a,f}\mathbf{P}_{c,a,f}{}^{\mathrm{T}} \tag{10}$$

where $R_{c,a,f}$ is up to rank $(\mathbf{P}_{c,a}{}^*)$. In this way, the increased quality-relevant relative variations in the alternative fault case are separated from those nonincreased ones and modeled orderly.

In RS of PLS Monitoring System:

Step (1). The same as Step (1) in systematic subspace of PLS monitoring system.

Step (2). Project $\mathbf{X}^v_{c,a}$ onto the column space of $\mathbf{R}^e_{c,r}$ to get the variations along each residual direction in $\mathbf{P}^e_{c,r}$: $\mathbf{E}^v_{c,a}=\mathbf{X}_{c,a}{}^v\mathbf{R}^e_{c,r}\mathbf{P}^e_{c,r}{}^T=\sum_{j=1}^{R^e_{c,r}}\mathbf{X}^v_{c,a}\mathbf{r}^e_{c,r,j}\mathbf{p}^e_{c,r,j}{}^{\mathrm{T}}$. The variation difference between the alternative fault case and the reference normal case in monitoring RS is defined as

$$\Delta_{c,a,i}=\|\mathbf{X}^v_{c,a}\mathbf{r}^e_{c,r}\mathbf{p}^e_{c,r}{}^{\mathrm{T}}\|^2-\|\mathbf{X}^v_{c,r}\mathbf{r}^e_{c,r}\mathbf{p}^e_{c,r}{}^{\mathrm{T}}\|^2(i=1,2,...,R^e_{c,r}) \tag{11}$$

where $\|\|$ denotes the Euclidean length. So, $\Delta_{c,a}$ is a $R^e_{c,r}$-dimensional vector composed of $\Delta_{c,a,i}$.

Step (3). Sort the values of $\Delta_{c,a}$ index. If the $\Delta_{c,a,i}$ index is larger than zero, it means that the variation in the alternative fault case is larger than that in the reference normal case, which thus may be responsible for out-of-control SPE monitoring statistic. Keep the directions corresponding to $\Delta_{c,a,i}$ values of larger than zeros, we can get $\mathbf{R}^e_{c,a}{}^*(J\times R^e_{c,a}{}^*)$, and the number of retained directions is $R^e_{c,a}{}^*$. $\mathbf{R}^e_{c,a}{}^n$ and $\mathbf{P}^e_{c,a}{}^n$ are composed of the left directions in $\mathbf{R}_{c,r}(J\times R_{c,r})$ and $\mathbf{P}_{c,r}(J\times R_{c,r})$, respectively, along which variations in the alternative fault case stay similar in comparison with those in the reference normal case.

Step (4). The process variations explained along $\mathbf{R}^e_{c,a}{}^*$ can be modeled as

$$\hat{\mathbf{X}}^e_{c,a,f}=\mathbf{X}^v_{c,a}\mathbf{R}^e_{c,a}{}^*\mathbf{P}^e_{c,a}{}^{*\mathrm{T}}=\mathbf{T}^{v,e*}_{c,a}\mathbf{P}^e_{c,a}{}^{*\mathrm{T}} \tag{12}$$

where $\mathbf{P}^e_{c,a}{}^{*\mathrm{T}}$ are calculated as $(\mathbf{T}^{v,e*\mathrm{T}}_{c,a}\mathbf{T}^{v,e*}_{c,a})^{-1}\mathbf{T}^{v,e*\mathrm{T}}_{c,a}\mathbf{X}^v_{c,a}$. Clearly, $\hat{\mathbf{X}}^e_{c,a,f}$ make contribution to out-of-control SPE monitoring statistic in the alternative fault case. PCA is then performed on $\hat{\mathbf{X}}^e_{c,a,f}$ to extract the major systematic variations in order with $R^e_{c,a,f}$ components

$$\hat{\hat{\mathbf{X}}}^e_{c,a,f}=\hat{\mathbf{X}}^e_{c,a,f}\mathbf{P}^e_{c,a,f}\mathbf{P}^e_{c,a,f}{}^{\mathrm{T}} \tag{13}$$

where $R^e_{c,a,f}$ is up to rank $(\mathbf{P}^e_{c,a}{}^*)$. In this way, the major increased relative variations in the alternative fault case are separated from those nonincreased ones and modeled orderly by PCA.

Based on the above analysis of relative changes, two different subspaces are separated from the original PLS systematic subspace and RS, respectively. The major fault effects

are extracted and used for fault reconstruction modeling. Two reconstruction models, $\mathbf{P}_{c,a,f}$ and $\mathbf{P}^{e}_{c,a,f}$, capture those major increased variations in each alternative fault case. They will be used for the correction of T^2 and SPE monitoring statistics, respectively.

Phase-based fault diagnosis via reconstruction

Analysis of fault deviations relative to normal status in each phase makes the fault effects responsible for alarm signals clear. The corresponding reconstruction models can thus better recover the fault-free part for diagnosis of fault causes. The fault data after the right correction action (i.e., adopting the correct reconstruction models) are expected to yield no alarm signals under the supervision of the original PLS monitoring system.

The reconstruction model-based corrections can be calculated in two monitoring subspaces within each phase

$$\begin{aligned}\hat{\mathbf{X}}^{v}_{c,a}{}^{\bullet} &= \hat{\mathbf{X}}^{v}_{c,a}(\mathbf{I}-\mathbf{P}_{c,a,f}(\mathbf{P}_{c,a,f}{}^{\mathrm{T}}\mathbf{P}_{c,a,f})^{-1}\mathbf{P}_{c,a,f}{}^{\mathrm{T}})\\ \mathbf{E}^{v}_{c,a}{}^{\bullet} &= \mathbf{E}^{v}_{c,a}(\mathbf{I}-\mathbf{P}^{e}_{c,a,f}(\mathbf{P}^{e}_{c,a,f}{}^{\mathrm{T}}\mathbf{P}^{e}_{c,a,f})^{-1}\mathbf{P}^{e}_{c,a,f}{}^{\mathrm{T}})\end{aligned} \tag{14}$$

where $\hat{\mathbf{X}}^{v}_{c,a}{}^{\bullet}$ and $\mathbf{E}^{v}_{c,a}{}^{\bullet}$ are the left parts of $\hat{\mathbf{X}}^{v}_{c,a}$ and $\mathbf{E}^{v}_{c,a}$ after the data correction, which are deemed to be normal if fault data are well reconstructed.

After the correction, the statistics are then updated by projecting the reconstructed data onto the original PLS monitoring system

$$\begin{aligned}\mathbf{T}^{v}_{c,a}{}^{*} &= \hat{\mathbf{X}}^{v}_{c,a}{}^{\bullet}\mathbf{R}_{c,r}\\ \mathbf{E}^{v}_{c,a}{}^{*} &= \mathbf{E}^{v}_{c,a}{}^{\bullet}\mathbf{R}^{e}_{c,r}\mathbf{P}^{e}_{c,r}{}^{\mathrm{T}}\end{aligned} \tag{15}$$

where $\mathbf{T}^{v}_{c,a}{}^{*}$ are the corrected PLS LVs, $\mathbf{E}^{v}_{c,a}{}^{*}$ are the final PLS residuals. The corresponding time-slice statistics are then separated from the variable-unfolding phase statistics based on the indication of process time. For example, $\mathbf{T}_{a,k}{}^{*}$ are the kth time-slice LVs separated from the phase-representative scores $\mathbf{T}^{v}_{c,a}{}^{*}$ where the current phase c is also known as indicated by the process time.

Based on different time-slice statistics, the monitoring statistics are updated at each time for each alternative fault case

$$\begin{aligned}T_{a,k,i}{}^{*2} &= (\mathbf{t}_{a,k,i}{}^{*}-\bar{\mathbf{t}}_{r,k})^{\mathrm{T}}\boldsymbol{\Sigma}_{r,k}{}^{-1}(\mathbf{t}_{a,k,i}{}^{*}-\bar{\mathbf{t}}_{r,k})\\ \mathrm{SPE}_{a,k,i}{}^{*} &= \mathbf{e}_{a,k,i}{}^{*\mathrm{T}}\mathbf{e}_{a,k,i}{}^{*}\end{aligned} \tag{16}$$

where subscript i denotes the ith batch in each time slice. $\mathbf{t}_{a,k,i}{}^{*}$ is the row vector in time-slice LVs $\mathbf{T}_{a,k}{}^{*}$; $\bar{\mathbf{t}}_{r,k}$ denotes the mean of time-slice LVs obtained from normal training data, which is in general zero vector resulting from the mean-centering preprocessing; $\boldsymbol{\Sigma}_{r,k}$ are the variance-covariances of time-slice LVs from normal training data. $\mathbf{e}_{a,k,i}{}^{*}$ is the row vector in the final PLS residuals, $\mathbf{E}^{v}_{c,a}{}^{*}$.

In comparison with the original out-of-control monitoring statistics, the updated monitoring statistics $T_{a,k,i}{}^{*2}$ and $\mathrm{SPE}_{a,k,i}{}^{*}$ should stay well within the predefined confidence limits after good fault correction. Online fault diagnosis is performed by finding the specific fault reconstruction models that can well correct the fault effects and bring the out-of-control signals back to normal. Whenever a new observation, $\mathbf{x}_{\mathrm{new}}\,(J\times 1)$, is available, it is first preprocessed using the data normalization information from the reference normal case. Based on the indication of process time, the current phase c is known. The normalized new observation is then

Table 1. Eleven Process Variables and One Quality Index Used in IM Process

	Variable's Descriptions	Unit
Process Variable No.		
1	Valve 1	%
2	Valve 2	%
3	Screw stroke	mm
4	Screw velocity	mm/sec
5	Ejector stroke	mm
6	Mold stroke	mm
7	Mold velocity	mm/sec
8	Injection press	Bar
9	Barrel temperature zone 3	°C
10	Barrel temperature zone 2	°C
11	Barrel temperature zone 1	°C
Quality Index No.		
1	Part weight	g

Table 2. The Setting of Different Operation Conditions for the Normal Case and Three Fault Cases in IM Process

	Operation Conditions	
Case	Barrel Temperature B.T. (°C)	Packing Pressure P.P. (Bar)
Normal data	200	30
Fault 1	180	30
Fault 2	200	25
Fault 3	220	35

projected onto the PLS monitoring models and the monitoring statistics are calculated

$$\begin{aligned}\mathbf{t}_{\mathrm{new}}{}^{T} &= \mathbf{x}_{\mathrm{new}}{}^{T}\mathbf{R}_{c,r}\\ \hat{\mathbf{x}}_{\mathrm{new}} &= \mathbf{x}_{\mathrm{new}}{}^{T}\mathbf{R}_{c,r}\mathbf{P}_{c,r}{}^{\mathrm{T}}\\ \mathbf{e}_{\mathrm{new}}{}^{T} &= \mathbf{x}_{\mathrm{new}}{}^{T}\mathbf{R}^{e}_{c,r}\mathbf{P}^{e}_{c,r}{}^{T}\end{aligned} \tag{17}$$

$$\begin{aligned}T_{r,\mathrm{new}}{}^{2} &= (\mathbf{t}_{\mathrm{new}}-\bar{\mathbf{t}}_{r,k})^{\mathrm{T}}\boldsymbol{\Sigma}_{r,k}{}^{-1}(\mathbf{t}_{\mathrm{new}}-\bar{\mathbf{t}}_{r,k})\\ \mathrm{SPE}_{r,\mathrm{new}} &= \mathbf{e}_{\mathrm{new}}{}^{T}\mathbf{e}_{\mathrm{new}}\end{aligned} \tag{18}$$

Compare the values of two monitoring statistics ($T_{r,\mathrm{new}}{}^{2}$ and $\mathrm{SPE}_{r,\mathrm{new}}$) with the predefined control limits in systematic subspace and RS, respectively. If both monitoring statistics stay well within the predefined normal regions, the current sample can be deemed to be normal. On the contrary, if alarm signals are issued, the current process is operating with some faults. Then, fault diagnosis should be taken online. The online fault diagnosis procedure is implemented as shown in Figure 1b. The correction includes two steps: data renormalization and fault reconstruction. For some faults, the only change relative to the normal status is the shift of center and/or increase of variable variances. In this way, the fault effects can be readily removed by data renormalization using the new center and variance in the fault data. For the other faults, simple data renormalization can not remove the alarm signals well enough. Reconstruction models have to be used to correct them. Therefore, during online application, first, data renormalization is performed and monitoring statistics are updated based on the renormalized data to check whether alarm signals can be removed. If not, reconstruction models defined for different alternative fault cases will be tested in turn to see which can best correct the current fault data

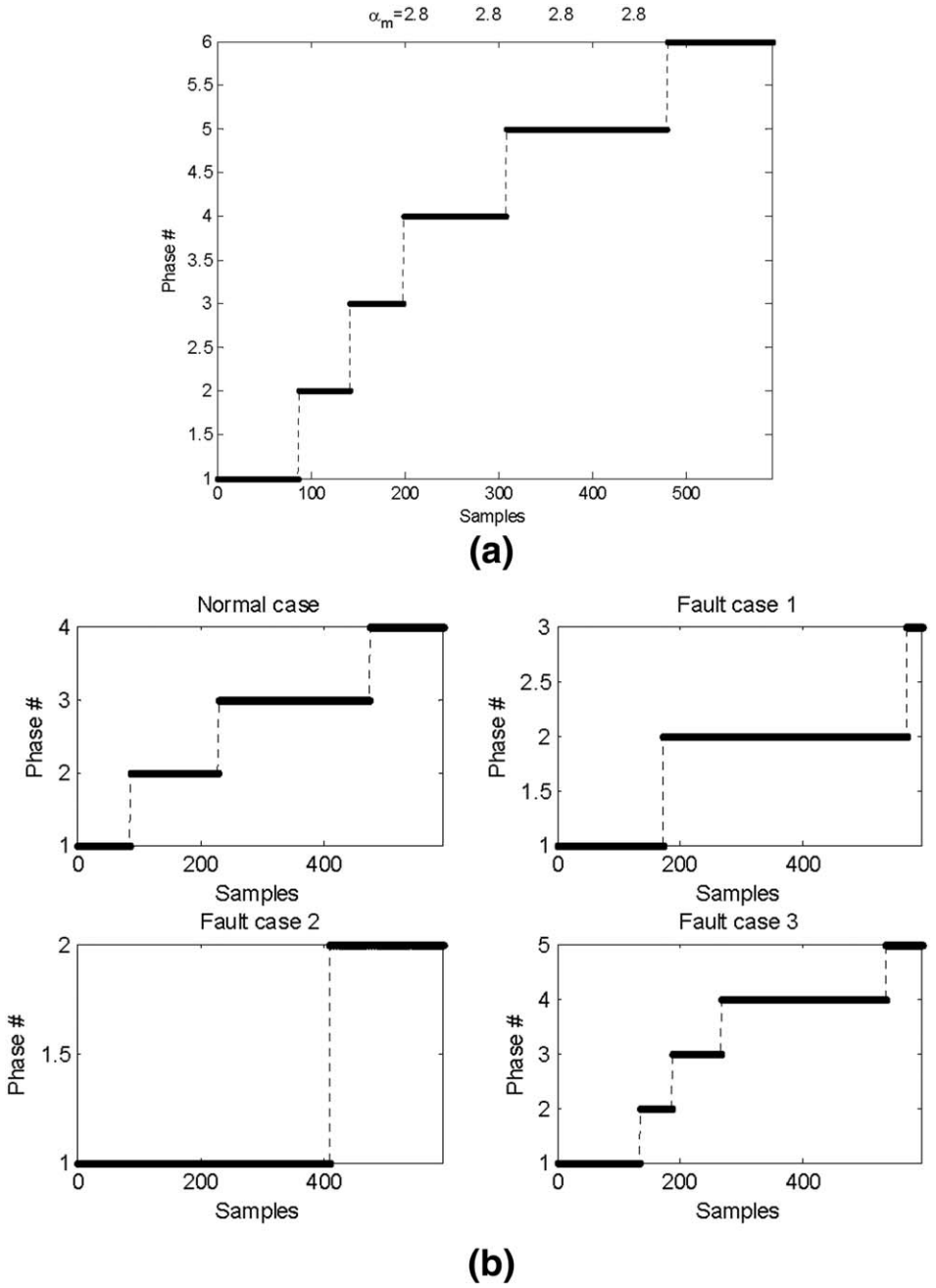

Figure 4. Phase partition results for multimode IM process with one normal case and three fault cases using (a) the proposed QCSSPP algorithm and (b) separate partition algorithm.

$$\begin{aligned}
\hat{\mathbf{x}}_{\text{new}}{}^{\bullet T} &= \hat{\mathbf{x}}_{\text{new}}{}^{T}\left(\mathbf{I}-\mathbf{P}_{c,a,f}\left(\mathbf{P}_{c,a,f}{}^{\mathrm{T}}\mathbf{P}_{c,a,f}\right)^{-1}\mathbf{P}_{c,a,f}{}^{\mathrm{T}}\right)\\
\mathbf{e}_{\text{new}}{}^{\bullet T} &= \mathbf{e}_{\text{new}}{}^{T}\left(\mathbf{I}-\mathbf{P}^{e}_{c,a,f}\left(\mathbf{P}^{e}_{c,a,f}{}^{\mathrm{T}}\mathbf{P}^{e}_{c,a,f}\right)^{-1}\mathbf{P}^{e}_{c,a,f}{}^{\mathrm{T}}\right)\\
\mathbf{t}_{\text{new}}{}^{*T} &= \hat{\mathbf{x}}_{\text{new}}{}^{\bullet T}\mathbf{R}_{c,r}\\
\mathbf{e}_{\text{new}}{}^{*T} &= \mathbf{e}_{\text{new}}{}^{\bullet T}\mathbf{R}^{e}_{c,r}\mathbf{P}^{e}_{c,r}{}^{\mathrm{T}}
\end{aligned} \tag{19}$$

$$\begin{aligned}
T_{\text{new}}{}^{*2} &= \left(\mathbf{t}_{\text{new}}{}^{*}-\bar{\mathbf{t}}_{r,k}\right)^{\mathrm{T}}\boldsymbol{\Sigma}_{r,k}{}^{-1}\left(\mathbf{t}_{\text{new}}{}^{*}-\bar{\mathbf{t}}_{r,k}\right)\\
\text{SPE}_{\text{new}}{}^{*} &= \mathbf{e}_{\text{new}}{}^{*\mathrm{T}}\mathbf{e}_{\text{new}}{}^{*}
\end{aligned} \tag{20}$$

If the fault data are well corrected, both updated monitoring statistics $T_{\text{new}}{}^{*2}$ and $\text{SPE}_{\text{new}}{}^{*}$ will stay well within the predefined confidence limits. Then, the fault cause is identified by finding the specific fault data normalization information and/or fault reconstruction models that can well correct the fault effects and bring the out-of-control signals back to normal.

Discussions and analyses

In the present work, the above analyses are made based on an implicit assumption that the normal and fault processes operate with equal run duration where the specific process time can be used as an indicator to judge which phase-representative reconstruction model should be adopted. However, in practical situations, the length of fault processes may be different from that of normal batches. For example, batches with severe faults may need to be aborted before the end of the batch and extra running time (reaction time) may

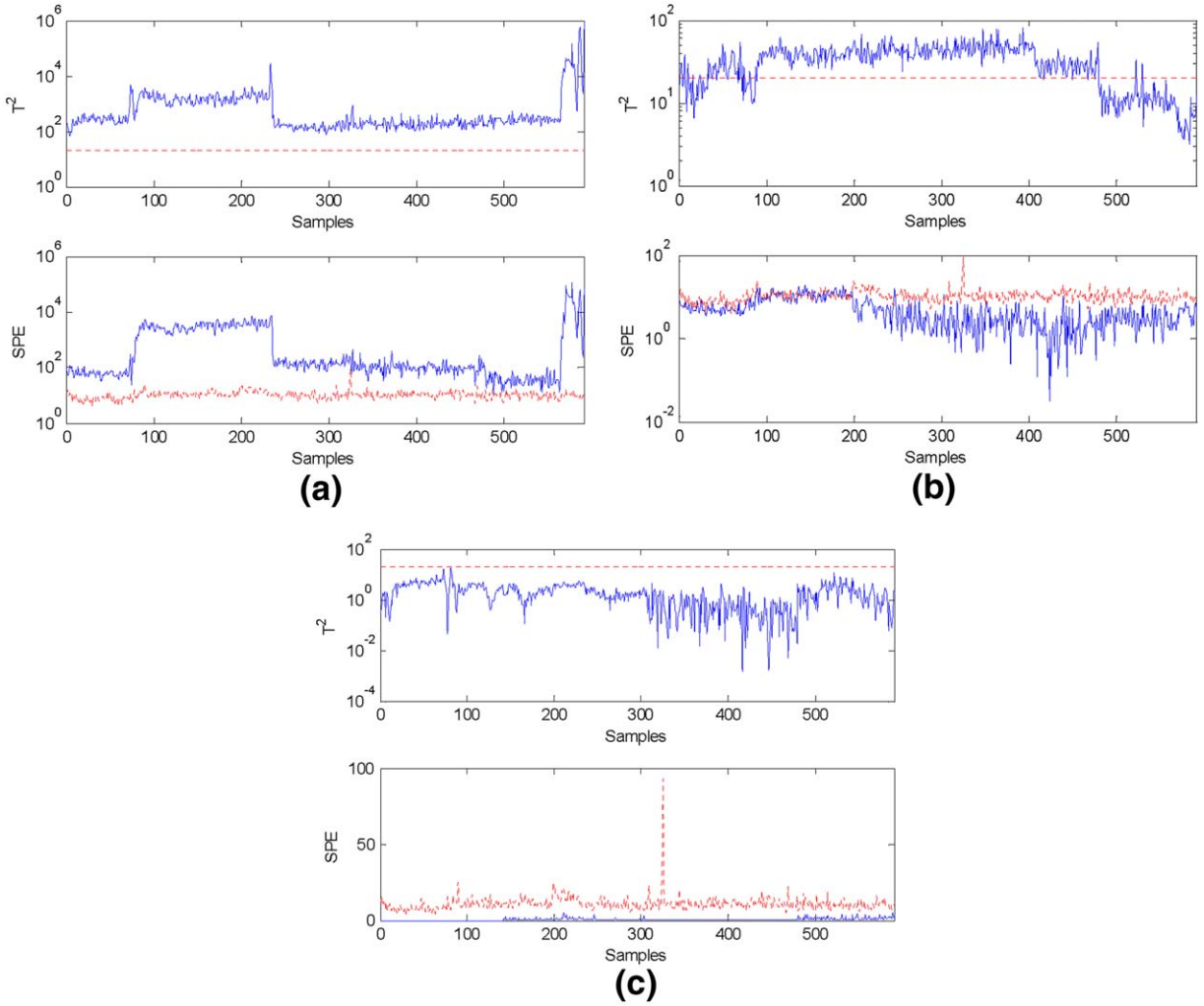

Figure 5. Online monitoring results for one batch from Fault 2 (a) with no correction, (b) corrected by data normalization information from Fault 2, and (c) corrected by reconstruction models from Fault 2 (dashed line, 99% control limits; solid line, the monitoring statistics).

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

be needed for some other faulty batches to meet the quality constraints. For this problem, two cases are analyzed and discussed as below:

1. Batches have the same duration in the same fault case and may have different lengths across different fault cases. If the major process characteristics and fault effects have been covered in the common region between normal and fault processes, only the common part is used for phase analysis, reconstruction modeling, and fault diagnosis. Here, the common region means the time region in which the batch trajectories of normal and fault cases overlap in a common time part. Using the concurrent phase analysis strategy which is conducted sequentially from the process beginning, multiple phases are obtained for the common part of normal and fault cases. The reconstruction model developed based on phase-based relative analysis is able to cover the major fault effects regarding their influences on monitoring performance. The process time still can be used as an indicator to adopt the phase-based reconstruction model and perform data correction for fault diagnosis. On such premise, the proposed method in the present work provides a simple solution to handle fault batches with unequal duration, which does not require equalizing the batch length beforehand.
2. If batches of one fault case have longer duration than the other cases and the major fault effects are not covered in the common time region, the fault data in the longer part have to be analyzed separately to reveal the fault characteristics and effects. The proposed strategy may not work well to handle this problem. Considering the complexity of unequal-length problem, there may be various circumstantialities should be integrated and paid special attention to. The further researches aiming at the uneven-length problem for phase analysis and fault diagnosis are promising and certainly will arouse extensive academic interest in future.

Illustrations

Injection molding process

IM, a key process in polymer processing for manufacturing industry, transforms polymer materials into various

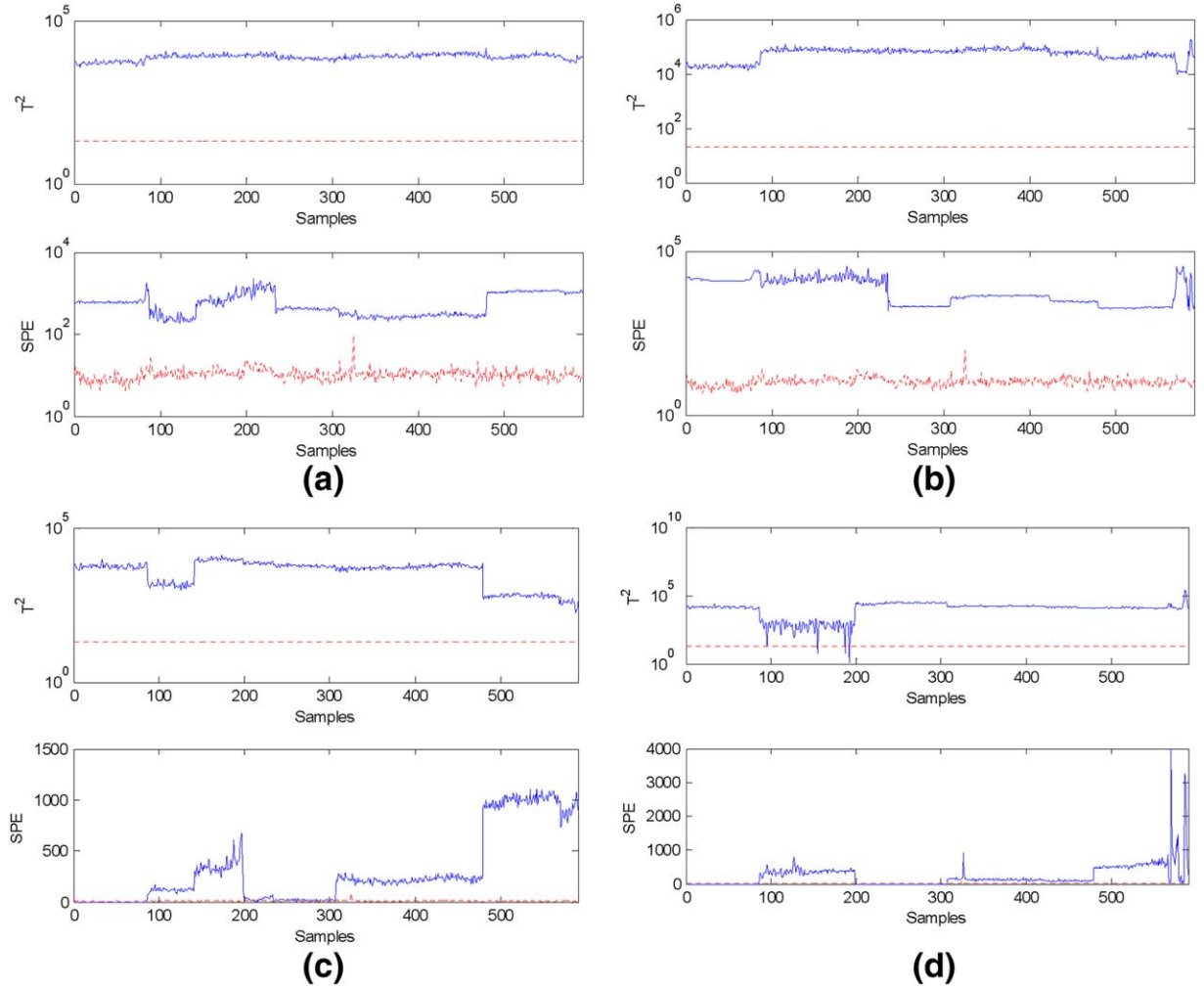

Figure 6. Online monitoring results for one batch from Fault 2 corrected using information from different faults (a) corrected by data normalization information from Fault 1, (b) corrected by data normalization information from Fault 3, (c) corrected by reconstruction models from Fault 1, and (d) corrected by reconstruction models from Fault 3 (dashed line, 99% control limits; solid line, the monitoring statistics).

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

shapes and types of products. A typical IM process consists of three major operation phases, injection of molten plastic into the mold, packing-holding of the material under pressure, and cooling of the plastic in the mold until the part becomes sufficiently rigid for ejection. Besides, plastication takes place in the barrel in the early cooling phase, where polymer is melted and conveyed to the barrel front by screw rotation, preparing for next cycle. It is a typical multiphase batch process and has been widely used for process monitoring and quality analysis.[38,42] It can be readily implemented for experiments, in which, all key process conditions, such as the temperatures, pressures, displacement, and velocity, can be online measured by various transducers, providing abundant process information.

The material used in this work is high-density polyethylene. Eleven process variables are selected for modeling as shown in Table 1, which can be collected online with a set of sensors. One dimension index, weight (g), is chosen to evaluate the product quality, whose real values can be directly measured by instruments at the end of each batch. Besides the normal operation process, three different operation statues are considered as shown in Table 2 by changing barrel temperature and packing pressures while keeping the other operation conditions invariable. Relative to the normal operation process, they are deemed to be fault cases resulting from changes of operation conditions. Twenty-eight batches are generated for the normal case, and 30 batches are collected for each fault case, all with even duration (591 samples in this experiment), which, thus, result in the descriptor arrays $\underline{\mathbf{X}}_1(28 \times 11 \times 591)$ and $\underline{\mathbf{X}}_m(30 \times 11 \times 591)(m = 2,3,4)$. The part weights are only measured at the end of each batch, generating the dependent vector $\mathbf{Y}_m(I_m \times 1)$ $(m = 1,2,3,4)$. For all fault cases, 20 batches are used for model development and the other 10 cycles are used for model testing.

Concurrent phase division and analysis of relative changes

Prepare the normalized time-slice data matrix $\mathbf{X}_{m,k}(I_m \times J)$ from the process beginning of each case as well as the normalized quality variable $\mathbf{Y}_m(I_m \times 1)$. Then, 591 time-slice data pairs are prepared for regression modeling in each case, $\{\mathbf{X}_{m,k}, \mathbf{Y}_m\}$. First, PLS is performed on each time-slice data pair for each case. The number of LVs at different time is determined by plotting PRESS[5] again the number of LVs to

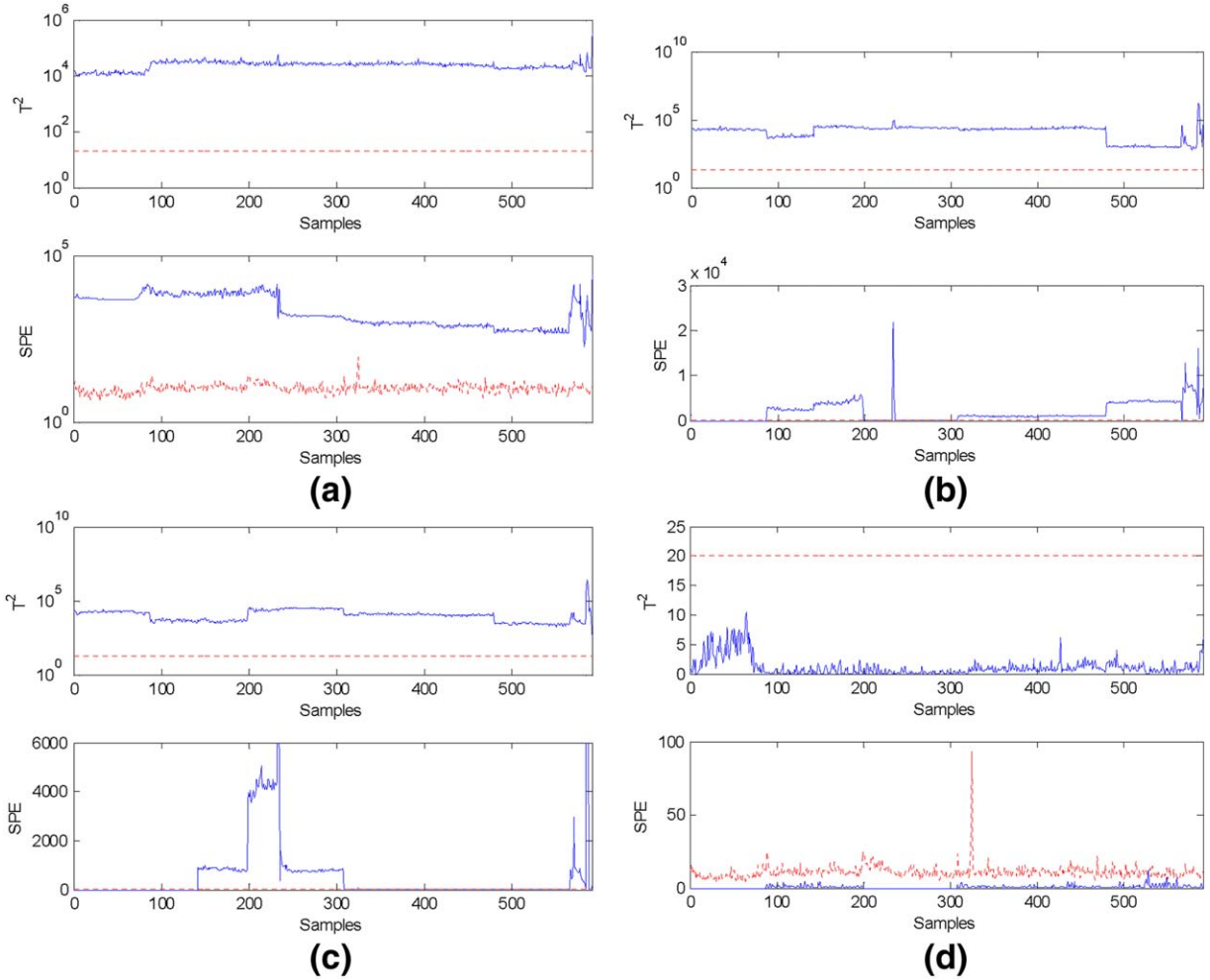

Figure 7. Online monitoring results for one batch from Fault 3 (a) with no correction, (b) corrected by information from Fault 1, (c) corrected by information from Fault 2, and(d) corrected by information from Fault 3 (dashed line, 99% control limits; solid line, the monitoring statistics).

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

keep good quality prediction. Since at most time the number of LVs is 6, the LV number used for the proposed QCSSPP algorithm is unified to be 6. Using QCSSPP algorithm, the normal case and three fault cases are considered simultaneously to check the changes of underlying process characteristics. Figure 4a shows the phase partition results using QCSSPP algorithm where the values of relaxing factor α_m are set to be 2.8, the same for all the concerned four cases (the normal case and three fault cases) for simplicity. This α_m value will not result in too many phases and the divided phases also in general reflect the changes of operation phases of IM. From the results, it is clear that using the proposed QCSSPP algorithm, the process can be automatically partitioned into different phases in time order with consistent phase landmarks across the normal case and three fault cases. In comparison, the separate phase partition results are also shown in Figure 4b for each single case. Separate phase division shows different phase landmarks for different cases

Table 3. Online Fault Diagnosis Results (Mean ± MAD[a]) by the Two-Step Fault Correction for Training Batches and Testing Batches

Fault #	Fault Case 1		Fault Case 2		Fault Case 3	
Data Correction	T^2	SPE	T^2	SPE	T^2	SPE
Training Batches						
By renormalization	9.85 ± 12.42	3.70 ± 3.83	15.60 ± 19.70	3.57 ± 2.76	8.61 ± 9.52	3.29 ± 2.21
By reconstruction models	0.03 ± 0.06	0.13 ± 0.19	0.19 ± 0.33	0.04 ± 0.07	0	0.06 ± 0.08
Test Batches						
By renormalization	11.90 ± 11.01	4.59 ± 1.75	53.01 ± 36.24	13.72 ± 7.87	89.71 ± 7.95	29.20 ± 12.37
By reconstruction models	4.59 ± 3.68	0.13 ± 0.08	25.11 ± 28.51	0.74 ± 0.66	20.08 ± 20.21	0.61 ± 0.58

[a]MAD: mean absolute deviation, which is calculated as $\frac{1}{I}\left(\Sigma_{q=1}^{N}|Z_q-\frac{\Sigma_{i=1}^{I}Z_i}{I}|\right)$ where Z denotes the values of MAR index for different batches. I denotes the number of training batches or testing batches. Mean is calculated to evaluate the batch-wise "average" fault diagnosis performance for each monitoring statistic; MAD can evaluate the batch-wise variability of fault diagnosis performance for each monitoring statistic.

which makes it inconvenient for the following phase-based analysis of relative changes. Also, the process characteristics of Fault 3 show faster changes than those of the normal case and the other two fault cases as it can be divided into more phases with the same relaxing factor α_m. Therefore, if phases are judged only based on the normal case, the faster changes of process characteristics of Fault 3 may not be well reflected. For simplicity, the phase division results using the clustering algorithm shown in Appendix A are not shown here, where the clustering results require a heavy postprocessing. The proposed QCSSPP algorithm jointly considers all cases to capture the changes of process characteristics so that a unified division result is available.

Based on the phase division result, phase-representative data pairs are arranged for each case and phase-based analysis of relative changes between the normal case and each alternative fault case is performed to decompose the fault effects. Fault reconstruction models are then developed in two PLS monitoring subspaces where the alarm-relevant fault effects are extracted for online fault diagnosis.

Online fault diagnosis

Based on the normal training data, the PLS monitoring models are developed and the normal confidence limits are defined. Online process monitoring is performed for three fault cases. In general, all faults can be clearly detected by both monitoring statistics. In Figure 5a, the online fault detection results of Fault 2 are plotted, revealing alarm signals throughout the whole process. Fault correction is required to remove the alarm signals for fault diagnosis. First, the new data normalization information (i.e., the new centers and standard deviations) which have been calculated from training data of each fault case is used for data correction. After that, fault detection is performed again. In comparison with the original online monitoring results, the alarm signals are reduced to a certain extent for Fault 2 as shown in Figure 5b. However, the fault effects are still noticed especially for T^2 monitoring statistics. Based on the developed reconstruction models, Fault 2 is further corrected and the updated monitoring statistics are shown in Figure 5c. Clearly, the alarm signals are completely restored to the normal region by the joint action of data renormalization and model reconstruction.

Besides, the batch from Fault 2 is also corrected using normalization information and reconstruction models obtained from the other two faults. The results are shown in Figure 6. For both monitoring statistics, if data normalization information from the other two faults is used, alarm signals are not restored. Instead, the out-of-control signals are more obvious as shown in Figures 6a, b. Further, the reconstruction models from Faults 1 and 3 are used to correct Fault 2 as shown in Figures 6c, d. For SPE monitoring chart, alarm signals are falsely brought back to normal in some phases (Phase I and/or Phase IV) if reconstruction models from the other two faults are used. But for T^2 monitoring chart, the alarm signals are not reconstructed throughout the batch. Combining the results for the two monitoring statistics, no other reconstruction models can falsely correct Fault 2. Therefore, the fault cause is identified as Fault 2 correctly. The online monitoring results and fault reconstruction results for one batch from Fault 3 are also plotted as shown in Figure 7. Using information from Faults 1 and 2, the out-of-control T^2 statistic values can not be brought back to normal while for SPE monitoring chart, some statistic values may be falsely restored in some phases as shown in Figures 7b, c. Using information from Fault 3, both out-of-control monitoring statistics are well corrected to stay well within the normal regions as shown in Figure 7d. Therefore, the fault cause is correctly judged.

For the concerned three fault cases, the reconstruction results are summarized in Table 3 based on the two-step reconstruction procedure. Only data renormalization is performed in the first step; and both data renormalization and model reconstruction are taken in the second step. Missing reconstruction ratio index is calculated for each batch by counting the number of alarm signals that are failed to be corrected and dividing it with the batch length (591 samples here). Then, the average results are evaluated for all batches using Mean $\pm$ MAD (mean absolute deviation). In general, the two-step fault reconstruction based on data renormalization and model reconstruction is much better than the first-step reconstruction which only performs data renormalization. Also, the results for testing data are worse than those for training data to a certain extent.

Conclusions

In this article, a quality-relevant concurrent phase partition and analysis of relative changes algorithm is proposed for reconstruction modeling and online fault diagnosis in multiphase batch processes. By the automatic concurrent phase partition algorithm, the phase landmarks are identified for the normal case and all fault cases simultaneously in a consistent way. The phase characteristics in fault cases can be better understood regarding the changes of process characteristics under the influences of disturbances. Then, phase-based relative analysis is performed between the normal and fault cases within each phase. By a comprehensive decomposition of fault data space, the fault effects responsible for the out-of-control monitoring statistics can be clearly extracted and captured for reconstruction modeling. Online fault diagnosis via reconstruction is performed in different phases to identify the fault cause where process time is used as an indicator to adopt the corresponding phase models. The case study has demonstrated the performance of the composed algorithm in fault diagnosis. It is hoped that this report will provide the basis for future work which might profitably take the following directions. First, in practice, it may be difficult to get enough batches for some faults. Therefore, how to extract fault subspaces and develop reconstruction models based on limited fault batches require special attentions and deserve further efforts. Second, how to perform fault prognostic via reconstruction results so as to avoid frequent downtime is also an interesting issue.

Acknowledgment

This work is supported by Program for New Century Excellent Talents in University (NCET-12-0492), Zhejiang Provincial Natural Science Foundation of China (LR13F030001), Project of Education Department of Zhejiang Province (Y201223159), the Open Research Project of the State Key Laboratory of Industrial Control Technology, Zhejiang University, China (No. ICT1320) and Technology Foundation for Selected Overseas Chinese Scholar of Zhejiang Province.

Literature Cited

1. Kourti T, MacGregor JF. Process analysis, monitoring and diagnosis, using multivariate projection methods. *Chemometr Intell Lab Syst*. 1995;28(1):3–21.

2. Jackson JE. *A User's Guide to Principal Components*. New York: Wiley, 2005.
3. Burnham AJ, Viveros R, MacGregor JF. Frameworks for latent variable multivariate regression. *J Chemometr*. 1996;10(1):31–45.
4. Dayal B, Macgregor JF. Improved PLS algorithms. *J Chemometr*. 1997;11(1):73–85.
5. Geladi P, Kowalski BR. Partial least-squares regression: a tutorial. *Anal Chim Acta*. 1986;185:1–17.
6. Qin SJ. Statistical process monitoring: basics and beyond. *J Chemometr*. 2003;17(8–9):480–502.
7. Raich A, Cinar A. Statistical process monitoring and disturbance diagnosis in multivariable continuous processes. *AIChE J*. 1996; 42(4):995–1009.
8. Gertler J, Li W, Huang Y, McAvoy T. Isolation enhanced principal component analysis. *AIChE J*. 1999;45(2):323–334.
9. Qin SJ, Li W. Detection, identification, and reconstruction of faulty sensors with maximized sensitivity. *AIChE J*. 1999;45(9):1963–1976.
10. Chiang LH, Russell EL, Braatz RD. Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares, and principal component analysis. *Chemometr Intell Lab Syst*. 2000;50(2):243–252.
11. Chiang LH, Kotanchek ME, Kordon AK. Fault diagnosis based on Fisher discriminant analysis and support vector machines. *Comput Chem Eng*. 2004;28(8):1389–1401.
12. Qin SJ, Li W. Detection and identification of faulty sensors in dynamic processes. *AIChE J*. 2001;47(7):1581–1593.
13. Yoon S, MacGregor JF. Fault diagnosis with multivariate statistical models part I: using steady state fault signatures. *J Process Control*. 2001;11(4):387–400.
14. Yue HH, Qin SJ. Reconstruction-based fault identification using a combined index. *Ind Eng Chem Res*. 2001;40(20):4403–4414.
15. Zhao CH, Sun YX, Gao FR. A multiple-time-region (MTR)-based fault subspace decomposition and reconstruction modeling strategy for online fault diagnosis. *Ind Eng Chem Res*. 2012;51(34):11207–11217.
16. Westerhuis JA, Gurden SP, Smilde AK. Generalized contribution plots in multivariate statistical process monitoring. *Chemometr Intell Lab Syst*. 2000;51(1):95–114.
17. Choi SW, Lee IB. Multiblock PLS-based localized process diagnosis. *J Process Control*. 2005;15(3):295–306.
18. Zhao CH, Wang FL, Gao FR, Zhang Y. Enhanced process comprehension and statistical analysis for slow-varying batch processes. *Ind Eng Chem Res*. 2008;47(24):9996–10008.
19. Alcala CF, Qin SJ. Reconstruction-based contribution for process monitoring. *Automatica*. 2009;45(7):1593–1600.
20. Dunia R, Qin SJ. Subspace approach to multidimensional fault identification and reconstruction. *AIChE J*. 1998;44(8):1813–1831.
21. Nomikos P, MacGregor JF. Monitoring batch processes using multiway principal component analysis. *AIChE J*. 1994;40(8):1361–1375.
22. Nomikos P, MacGregor JF. Multivariate SPC charts for monitoring batch processes. *Technometrics*. 1995;37(1):41–59.
23. Nomikos P, MacGregor JF. Multi-way partial least squares in monitoring batch processes. *Chemometr Intell Lab Syst*. 1995;30(1):97–108.
24. Kosanovich KA, Dahl KS, Piovoso MJ. Improved process understanding using multiway principal component analysis. *Ind Eng Chem Res*. 1996;35(1):138–146.
25. Wise BM, Gallagher NB, Butler SW, White DD, Barna GG. A comparison of principal component analysis, multiway principal component analysis, trilinear decomposition and parallel factor analysis for fault detection in a semiconductor etch process. *J Chemometr*. 1999; 13(3–4):379–396.
26. Wold S, Kettaneh N, Fridén H, Holmberg A. Modelling and diagnostics of batch processes and analogous kinetic experiments. *Chemometr Intell Lab Syst*. 1998;44(1):331–340.
27. Louwerse DJ, Smilde AK. Multivariate statistical process control of batch processes based on three-way models. *Chem Eng Sci*. 2000; 55(7):1225–1235.
28. Sprang EN, Ramaker HJ, Westerhuis JA, Gurden SP, Smilde AK. Critical evaluation of approaches for on-line batch process monitoring. *Chem Eng Sci*. 2002;57(18):3979–3991.
29. Smilde AK. Comments on three-way analyses used for batch process data. *J Chemometr*. 2001;15(1):19–27.
30. Undey C, Cinar A. Statistical monitoring of multistage, multiphase batch processes. *IEEE Control Syst*. 2002;22(5):40–52.
31. Kourti T. Multivariate dynamic data modeling for analysis and statistical process control of batch processes, start-ups and grade transitions. *J Chemometr*. 2003;17(1):93–109.
32. Lu NY, Gao FR, Wang FL. Sub-PCA modeling and on-line monitoring strategy for batch processes. *AIChE J*. 2004;50(1):255–259.
33. Kosanovich K, Piovoso M, Dahl K, MacGregor J, Nomikos P. Multi-way PCA applied to an industrial batch process. *American Control Conference*, Baltimore, MD, 1994.
34. Dong D, McAvoy TJ. Multi-stage batch process monitoring. *Proceedings of American Control Conference*, Seattle, WA, 1995.
35. Zhao CH, Wang FL, Gao FR, Lu NY, Jia MX. Adaptive monitoring method for batch processes based on phase dissimilarity updating with limited modeling data. *Ind Eng Chem Res*. 2007;46(14):4943–4953.
36. Yoo CK, Villez K, Lee IB, Rosen C, Vanrolleghem PA. Multi-model statistical process monitoring and diagnosis of a sequencing batch reactor. *Biotechnol Bioeng*. 2007;96(4):687–701.
37. Liu J, Wong DSH. Fault detection and classification for a two-stage batch process. *J Chemometr*. 2008;22(6):385–398.
38. Zhao CH, Sun YX. Step-wise sequential phase partition (SSPP) algorithm based statistical modeling and online process monitoring. *Chemometr Intell Lab Syst*. 2013;125:109–120.
39. Martens H. Multivariate calibration. New York: Wiley, 1991.
40. Zhao CH, Gao FR, Wang FL. An improved independent component regression modeling and quantitative calibration procedure. *AIChE J*. 2010;56(6):1519–1535.
41. Lowry CA, Montgomery DC. A review of multivariate control charts. *IIE Trans*. 1995;27(6):800–810.
42. Zhao CH, Wang FL, Mao ZZ, Lu NY, Jia MX. Quality prediction based on phase-specific average trajectory for batch processes. *AIChE J*. 2008;54(3):693–705.

Appendix A: Clustering-Based Phase Division Algorithm

Inputs: the patterns to be partitioned, $\{\breve{\mathbf{P}}_1,\breve{\mathbf{P}}_2,.,\breve{\mathbf{P}}_k\}$, and the threshold θ for cluster elimination, the minimal phase length, $L_{\min}$.

Outputs: the number of clusters C, the cluster centers $\{\mathbf{W}_1, \mathbf{W}_2, ..., \mathbf{W}_C\}$, and the strict membership of $\breve{\mathbf{P}}_k$ to C centers, $m(k)$.

The index variables are the iteration index i, and the pattern index, k.

1. Choose $C^0(i=0)$ cluster centers $\mathbf{W}_c^0(c=1,2,\ldots,C^0)$ from the K patterns along the time direction. Practically, the initial cluster centers can be assumed to be uniformly distributed in the pattern set.
2. Merge pairs of clusters whose intercenter distance, $\text{dist}\,(\mathbf{W}_{c1}^{i-1}, \mathbf{W}_{c2}^{i-1})$, is below the predetermined threshold θ.
3. Calculate the distances from each pattern $\breve{\mathbf{P}}_k$ to all of the centers, $\text{dist}\,(\breve{\mathbf{P}}_k, \mathbf{W}_c^{i-1})$, assign $\breve{\mathbf{P}}_k$ to the nearest center $\mathbf{W}_{c^*}^{i-1}$, and denote its membership as $m(k)=c^*$.
4. Eliminate the clusters that catch few patterns (less than the minimal phase length, $L_{\min}$) after a set number of iterations to avoid singular clusters.
5. Update the number of cluster centers to be C^i, recompute the new cluster centers $\mathbf{W}_c^i(c=1,2,\ldots,C^i)$, using the current cluster membership, $m(k)$.
6. Go back to Step 2 if a convergence criterion is not met. Typical convergence criteria are minimal changes in the cluster centers and/or minimal rate of decrease in squared errors.

Appendix B: SSPP Algorithm

Step 1: Data preparation

Prepare the time-slice data matrix $\mathbf{X}_k(I\times J)$ from the process beginning. The variables at each time are then preprocessed to have zero mean and standard deviation. Input the normalized time-slice data matrix $\bar{\mathbf{X}}_k(I\times J)$.

Step 2: Time-slice-based PCA modeling

Perform PCA algorithm on the time-slice data matrices and get the initial time-slice models. The number of principal components (PCs) is determined by cumulative explained variance rate[2] to keep most of the process variability (90% here). Then, find the number of PCs that occurs most throughout the batch process and set it as the unified dimension of time-slice PCA models, R. Calculate the monitoring statistic values of SPE after the explanation of PCA model at each time and determine the confidence limit, Ctr_k, by a weighted Chi-squared distribution.[22] It represents the reconstruction power of time-slice PCA model.

Step 3: Time-segment-based PCA modeling

From the beginning of batch processes, add the next time-slice to the existing ones and variable-unfold them within the current time region, $\mathbf{X}_{v,k^*}(Ik^* \times J)$ (where subscript k^* denotes the current time). Perform PCA on the rearranged data matrix and get the time-segment PCA model, $\mathbf{P}_{v,k^*}(J \times R)$. Calculate the SPE values for each time-slice data matrix up to k^* by the explanation of the current time-segment PCA model ($\mathbf{P}_{v,k^*}(J \times R)$) and determine the confidence limit at each time, $Ctr_{v,k}$, by a weighted Chi-squared distribution.[22] It represents the reconstruction power of this time-segment PCA model to each time-slice up to k^*.

Step 4: Compare model accuracy

Compare $Ctr_{v,k}$ with Ctr_k for each time slice up to k^*. If more than three consecutive samples show $Ctr_{v,k} > \alpha * Ctr_k$, where α is a constant attached to Ctr_k, termed relaxing factor here. It means that the addition of the current time-slice have imposed great influences on the time-segment PCA monitoring model and the resulting monitoring performance. The accuracy of time-segment model is, thus, significantly worse than that of time-slice models. So, α determines how much the time-segment PCA model is allowed to be less representative than time-slice PCA models, that is, insufficient reconstruction power than the same-dimensional time-slice PCA models. The time slices before k^* are thus denoted as one phase.

Step 5: Data updating and recursive implementation

Remove the first phase and the left batch process data are now used as the new input data in Step 3. Recursively repeat Steps 3–4 from the updated beginning of the batch process to find the following segments.

Manuscript received Aug. 6, 2013, and revision received Dec. 19, 2013.