Improved Batch Process Monitoring and Quality Prediction Based on Multiphase Statistical Analysis

ARTICLE *in* INDUSTRIAL & ENGINEERING CHEMISTRY RESEARCH · JANUARY 2008

Impact Factor: 2.59 · DOI: 10.1021/ie0707624

5 AUTHORS, INCLUDING:

Chunhui Zhao
Zhejiang University
91 PUBLICATIONS **580** CITATIONS
SEE PROFILE

Zhizhong Mao
Northeastern University (Shenyang, China)
66 PUBLICATIONS **335** CITATIONS
SEE PROFILE

Mingxing Jia
Northeastern University (Shenyang, China)
36 PUBLICATIONS **360** CITATIONS
SEE PROFILE

Improved Batch Process Monitoring and Quality Prediction Based on Multiphase Statistical Analysis

Chunhui Zhao, Fuli Wang,* Zhizhong Mao, Ningyun Lu,† and Mingxing Jia

College of Information Science and Engineering, Northeastern University, Shenyang, Liaoning Province, People's Republic of China

An integrated online process monitoring, fault diagnosis, and quality prediction framework based on multiple models is developed for multiphase batch processes. Two types of different-level PLS models are designed for each phase. In the first level, multiple PLS models based on variable unfolding, termed MVPLS, are employed for online monitoring and revealing the dynamic progressing direction that the quality will take at each time. In each phase, MVPLS model is established by performing PLS on variablewise data rearrangement, which can focus on the process information more relevant to the final quality from an overall phase-specific point of view. Moreover, control limits of quality prediction are explored and, thus, employed in process monitoring. Combined with conventional Hotelling-T^2 and SPE statistics, they can enhance the process understanding and monitoring performance. Once an abnormality is detected, the contributions of process variables to quality prediction are calculated in combination with control limits to help check the fault variables quantitatively. In the second level, regression models are designed based on the phase-specific average score trajectory for more credible prediction of end-of-batch quality, where the phase-specific average trajectory reveals the accumulation effects of process variations on quality in a simpler structure. For a normal batch, whenever a phase is successfully completed, a more accurate quality prediction can be naturally obtained by implementing the second-level regression analysis. The case studies from a simulated fed-batch penicillin cultivation process demonstrate the power and advantage of the proposed method.

1. Introduction

Batch and semibatch processes play an important role in the processing of specialty chemical, semiconductor, food, and biology industries for producing high-value-added products to meet today's rapidly changing market. Characterized by finite duration, batch process operation is carried out to produce products of desired quality at the lowest possible cost. The common natures of nonsteady, time-varying, finite duration and nonlinear behaviors make batch processes more difficult to control than continuous processes. Hence, proper process monitoring and diagnosis is important to not only quality improvement but also process safety.[1−7] Multivariate statistical methods, such as principal component analysis (PCA) and partial least-squares (PLS), have been successfully used in modeling multivariate continuous processes. Several extensions of the conventional PCA/PLS to batch processes have also been reported, among which most batch process monitoring methods are based on unfold-PCA (uPCA) and unfold-PLS (uPLS). Numerous review books and papers have been given by many researchers.[8−15] Lee et al.[14] proposed a new uPCA-based statistical batch monitoring approach via variablewise unfolding and time-varying score covariance structures, which need not estimate the unavailable future measurements and can reflect the dynamics of the scores in the Hotelling-T^2 monitoring chart. Chen and Liu[15] integrated the time-lagged windows of process dynamic behavior with the principal component analysis and partial least-squares. respectively, for online batch monitoring, which can easily track the running progress and monitor the occurrence of upsets. However, the conventional uPCA/uPLS method takes the entire batch data as a single object, thus unable to differentiate the phase-specific local behaviors of the batch processes. Considering that the multiplicity of operation phases is an inherent nature of many batch processes and each phase exhibits significantly different underlying behaviors, it is desirable to develop phase-based models. Then each model represents a specific phase and focuses on the local behaviors of the batch processes.[16] Various strategies have been proposed,[17−26] which effectively enhanced process understanding and improved monitoring reliability. Kosanovich et al.[17] and Dong and McAvoy[18] developed two uPCA/nonlinear uPCA 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. Zhao et al.[20,21] investigated multiple PCA/PLS models for different operating modes based on metrics in the form of principal angles to measure the similarities of any two models, which have been successfully applied to continuous processes. Moreover, Camacho and Picó[23,24] developed a new algorithm for the detection of phases in a batch processes, which first automatically divide the batch into different phases so as to improve the prediction power of the multiphase models. Lu et al.[25,26] developed a series of phase-based sub-PCA modeling and monitoring methods based on the fact that process correlations remain similar within the same phase and their changes closely relate to the alteration of different phases in multiphase batch processes.

In the present work, a process monitoring and quality prediction scheme based on two-level multiple PLS models is proposed for multiphase batch processes. On the basis of PLS analysis in each phase, one can extract the critical inherent information from the process measurement trajectories that are more relevant to the final quality. On the one hand, using the first-level regression models, here termed MVPLS models, the real-time quality prediction can be carried out at each time to reveal the dynamic progress direction that the quality prediction

* Corresponding author. Tel.: +86-024-83687434. Fax: +86-024-23890912. E-mail: flwang@mail.neu.edu.cn.

† College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu Province, P. R. China.

10.1021/ie0707624 CCC: $40.75
Published on Web 01/04/2008

will take. On the other hand, at the ending of each phase, the quality is predicted more precisely using phase-average trajectory, which takes the accumulation effects of process variations on quality into account.

Compared with other process monitoring and quality prediction methods, the scheme proposed here has the following advantages:

(1) Combined with the prior phase clustering algorithm, it allows one to better understand the process phase-specific nature and classify different phases based on their different relations to quality, which can help to locate local effects of process variable trajectories on final quality and quickly identify the fault causes.

(2) Modeling based on variablewise unfolding way, the quality prediction can be conducted online at each time without estimating the unknown future process observations. Moreover, the control limits for real-time quality prediction are established using kernel density estimation method. Therefore, it enhances the reliability of process monitoring performance from a more direct supervision view related with quality guarantee. In other words, combined with traditional statistical process control (SPC) charts, the validity of quality prediction can be checked online without blindly accepting the predicted results.

(3) The second-level quality prediction is developed based on phase-specific score average trajectories. On the one hand, the data has been compressed prior, which removes the impact of nonsignificant variables and measurement noise since the extracted score features are used as the input. On the other hand, by means of the phase-specific average process trajectory, the phase-specific accumulative effects of process variations on quality are explored and the regression models are developed in a simpler structure.

(4) Control limits for contributions to quality prediction are explored to directly and reliably check those fault variables that are mainly responsible for the deteriorated quality. Thereupon it provides a determinate evaluation standard to quantitatively identify the process variables that are impacting the product quality.

This paper is organized as follows. In Section 2, based on different unfolding ways, conventional uPLS modeling method is depicted in outline. Consequently, details of the MVPLS strategy are explained and the monitoring statistics are established in Section 3. Moreover, second-level quality prediction using phase-specific score average trajectory is developed. The effectiveness of process monitoring and quality prediction using the proposed method is illustrated by applying it to the simulation benchmark of fed-batch penicillin fermentation process in Section 4. Finally, conclusions are drawn in Section 5.

2. uPLS Modeling for Batch Processes

As two common data compression and dimension reduction techniques, uPCA and uPLS have widely been used to reveal the latent structures of the batch data. Both project the information in the high-dimensional data space down onto the low-dimensional space defined by a small number of latent features. These new latent variables summarize all the important information contained in the original data sets. uPCA focuses only on the latent operating nature underlying the process measurements (*X*-data). As a consequence, it will flag any variation in the process measurements even though it may be irrelevant to the product quality.[10] In general, it may be beneficial to focus on those process deteriorations more relevant to product quality and correct them before they lead to permanent malfunctions. The only difference between PLS and PCA would be that the process variations that are closely concerned with the quality variables would be used in PLS, rather than all the variations of the process measurements as used in PCA. In fact, the PLS model attempts to explain and analyze the "common-cause" variations in the process data and to exclude the random variations and measurement errors that are linearly uncorrelated with quality property.

2.1. PLS Algorithm Based on Different Unfolding Ways. In each batch run, assume that J process variables are measured at K time instances throughout the batch and one quality variable at the end of each batch. I batch runs, a similar data set for each run, result in a three-way data array $\underline{\mathbf{X}}(I \times J \times K)$. The final quality variables for each batch are summarized in a $I \times 1$ vector $\mathbf{Y}$. Here, it should be noted that, in the present paper, we shall treat only the case with a univariate dependent variable; for the multivariate case, one just needs to treat all of the dependent variables simultaneously.

Before performing statistical analysis, the three-dimension data array $\underline{\mathbf{X}}(I \times J \times K)$ has to be unfolded, resulting in different two-dimensional matrices. Each of these two-dimensional matrices corresponds to checking different types of variability. In the method by Nomikos and MacGregor,[9,10] each of the vertical time slices represents the process variability information along all the batches at a common time interval k. They are put side by side to the right to create the matrix: $\mathbf{X}(I \times JK)$. In order to check deviations from the mean trajectory, it is mean centered by subtracting the mean. Moreover, each variable is scaled to unit variance to handle different measurement units, thus giving each equal weight. Once the unfolded matrix is normalized, PLS is performed as follows,

$$\mathbf{X}(I \times JK) = \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E}, \quad \mathbf{Y}(I \times 1) = \mathbf{T}\mathbf{Q}^{\mathrm{T}} + \mathbf{F} \tag{1}$$

where $\mathbf{T}$ is given by

$$\mathbf{T}(I \times A) = \mathbf{X}\mathbf{W}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1} = \mathbf{X}\mathbf{R} \tag{2}$$

where A is the number of retained latent variables and $\mathbf{R}(JK \times A) = \mathbf{W}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1}$. For batch data, PLS decomposes $\mathbf{X}(I \times JK)$ and $\mathbf{Y}(I \times 1)$ matrices into the loadings matrix for $\mathbf{X}$, $\mathbf{P}(JK \times A)$, the loadings matrix for $\mathbf{Y}$, $\mathbf{Q}(1 \times A)$, the scores matrix of $\mathbf{X}$, $\mathbf{T}(I \times A)$, and the weights matrix for $\mathbf{X}$, $\mathbf{W}(JK \times A)$, plus residuals matrices $\mathbf{E}(I \times JK)$ and $\mathbf{F}(I \times 1)$ for $\mathbf{X}$ and $\mathbf{Y}$, respectively. Each row of the $\mathbf{T}$ matrix corresponds to a single batch over the whole process and depicts the variability along batch direction. The $\mathbf{P}$ and $\mathbf{W}$ matrices summarize the variations of the measurement variables related to $\mathbf{Y}$, and their elements give each variable the weight to calculate the scores for each batch. The $\mathbf{Q}$ matrix relates the variability of the process measurements to the final product quality.

The regression relationships can also be expressed more directly and concisely in terms of the $\mathbf{X}$ data

$$\mathbf{Y}(I \times 1) = \mathbf{X}\boldsymbol{\beta} + \mathbf{F} \tag{3}$$

where the matrix of regression coefficients is given by

$$\boldsymbol{\beta}(JK \times 1) = \mathbf{W}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1}\mathbf{Q}^{\mathrm{T}} \tag{4}$$

In this way, uPLS takes process trajectories over the entire batch course as the input to pick up the critical information from process variations that are closely related with the quality variable. Therefore, it delivers such a concept that the final quality should depend on the whole process operation trajectories, i.e., the time-cumulative effects of process behaviors on

quality can be further investigated. However, it bears the drawback that the missing future process observations should be forecasted online.

Recently, applications have been presented in which the three-way process data matrix was unfolded variablewise[3,11,27−29] for process monitoring and diagnosis. K number of vertical time-slice matrices, $\mathbf{X}_k(I \times J)$, are placed beneath one another to form the two-way data matrix $\mathbf{X}(IK \times J)$, which can obtain the variance information along all batches and time. In practical industrial application, the explained variable is often a local time-stamp vector or a continuously measured maturity variable,[29] which is also arranged to keep the same row dimensionality as $\mathbf{X}(IK \times J)$. The evolution of the batch can then be monitored by developing a PLS regression model between the process measurements $\mathbf{X}(IK \times J)$ and the response variable, $\mathbf{Y}(IK \times 1)$. The scores matrix for $\mathbf{X}$, $\mathbf{T}(IK \times A)$, and the loadings matrices, respectively, for $\mathbf{X}$ and $\mathbf{Y}$, $\mathbf{P}(J \times A)$ and $\mathbf{Q}(1 \times A)$, are extracted using the PLS method as follows,

$$\mathbf{X}(IK \times J) = \mathbf{TP}^{\mathrm{T}} + \mathbf{E}, \quad \mathbf{Y}(IK \times 1) = \mathbf{TQ}^{\mathrm{T}} + \mathbf{F} \qquad (5)$$

where $\mathbf{E}(IK \times J)$ and $\mathbf{F}(IK \times 1)$ are the residuals. The scores contain the major dynamic relations along both time and batch directions, while the loadings show the overall correlations among process variables.

This unfolding way does not require all batches to be of equal length, nor does it require estimation of future data for online application. The conventional mean centering for the unfolding matrix, however, leaves the nonlinear time-varying trajectories in the data matrix because it simply subtracts a constant, the grand mean of each variable over all batches and time, from the trajectory of each variable. Therefore, this technique may produce weak results for small disturbances when the goal is to check deviations from the mean trajectory.[29]

2.2. Phase-Division Algorithm. It is well-known that there have been various techniques employed to get phase landmarks, such as prior process expertise, clustering algorithm, and so on. Generally, all the alternative ways provide diverse solutions from different views and aspects with different advantages and specific applicability under some conditions. A variant of k-means clustering algorithm has been developed in our previous work.[25,26] Different from the division algorithm by Camacho and Picó,[23,24] our clustering algorithm is developed assuming that the alternation between phases can be reflected by checking the changing of process correlations. In our opinion, the process correlations remain similar within the same phases, indicating that the operation is driven by the similar underlying characteristics. On the basis of this idea, in our clustering algorithm, those consecutive samples with similar correlations are collected together as a phase, and thus, one phase-representative monitoring model can be naturally established. The clustering algorithm used for the determination of phases is simple and straightforward without increasing comprehension and treatment complexity.

In our work, the batches are of equal length without special declaration so that the specific process time can be used as an indicator to data preprocessing, modeling, and online monitoring. First, the data matrices are aligned along time and autoscaled, resulting in K number of time-slice data matrices $\mathbf{X}_k(I \times J)$. Then PCA is performed on these time-slice matrices $\mathbf{X}_k$ ($k = 1, 2, ..., K$), generating K number of loadings matrices, $\mathbf{P}^k$, which represent the process correlations at K time intervals. Then an improved K-means clustering algorithm is employed to group the K patterns into different numbers of clusters, and finally C subphases are obtained by combining the clustering result with process time. Then phase-specific models are developed, which reflect the different phase-specific underlying process behaviors. However, the representative phase models[25,26] are simply obtained by averaging time-slice loadings within the same phase. Thus, they actually only focus on the variance variations along the batch axis localized at each separated time interval.

3. PLS-Based Online Monitoring and Quality Prediction

3.1. Phase-Division and Phase-Based Multiple PLS Modeling. As shown in Figure 1a, data normalization is performed on each time-slice matrix $\mathbf{X}_k(I \times J)$. Subtracting the mean of each column can eliminate the main nonlinearity due to the dynamic behaviors of the process and look at the deviation from the average trajectory; each variable is scaled to unit variance, thus giving each equal weight, as is quality variable, $\mathbf{Y}$. Then PLS analysis is conducted on these normalized time slices and the final quality variable, $\{\tilde{\mathbf{X}}_k, \tilde{\mathbf{Y}}\}$. Corresponding to each time interval, the loadings matrix, $\mathbf{P}_k(J \times J)$, and the scores matrix, $\mathbf{T}_k(I \times J)$, for process data measurements thus can steadily be extracted. Here, it should be noted that the time-slice loadings matrices used in the clustering algorithm, $\mathbf{P}_k$, are obtained from the PLS algorithm instead of PCA in our previous work.[25,26] They represent the time-specific covariance information and incline to focus more on revealing the underlying process behaviors closely linearly related to the quality. Moreover, to more properly implement the clustering algorithm, it is necessary to consider the importance of each principal projection direction, $\mathbf{P}_{k,j}$, and give them different weights. $\mathbf{S}_k(J \times J)$, the covariance matrix of $\mathbf{T}_k$, is calculated at each time. Since different latent variables are interorthogonal, i.e., $\mathbf{t}_i^{\mathrm{T}}\mathbf{t}_j = 0$ ($i \neq j$), it will satisfy $\mathbf{S}_k(J \times J) = \mathrm{diag}(\lambda_{k,1}, \lambda_{k,2}, ..., \lambda_{k,J})$ and $\lambda_{k,j}$ actually denotes the variance information of the jth latent variable at the kth time, $t_{k,j}$. Consequently, the loadings matrices are transformed into a weighted form,

$$\check{\mathbf{P}}_k = [\mathbf{P}_{k,1} \cdot w_{k,1}, \mathbf{P}_{k,2} \cdot w_{k,2}, ..., \mathbf{P}_{k,J} \cdot w_{k,J}] \qquad (6)$$

where $w_{k,j} = \lambda_{k,j}/\sum_{j=1}^{J} \lambda_{k,j}$.

By clustering the weighted loadings matrices, the process will be partitioned into different groups. In most cases, since the process correlations show the developing trend along the time axis, the clustering result can be associated with the operation time, which makes the partition of the patterns well-interpretable. However, sometimes there may be some exceptions. For example, several disjoint operation periods with the similar underlying characteristics are grouped into the same cluster or some short periods are separated from one cluster, which break the time continuity of the cluster. Therefore, the clustering result must be postprocessed to obtain the phases, where process operation time or an indicator variable can be used to indicate the phase-division information besides the clustering result. In the postprocessing procedure, those disjoint operation periods are checked and redesignated into the corresponding phases based on both the distance and time indices. So the duration can be partitioned into several phases along the process time, where each separated phase contains a series of successive samples more than a predefined minimum phase length, and process correlations are different over phases.

After phase partition, it is proper to extract a phase-representative PLS model in each phase to conduct statistical analysis, since effects of process variables on quality remain similar within the same phase, i.e., they indeed have the similar correlation relationships. The phase-based modeling procedure is presented as follows:

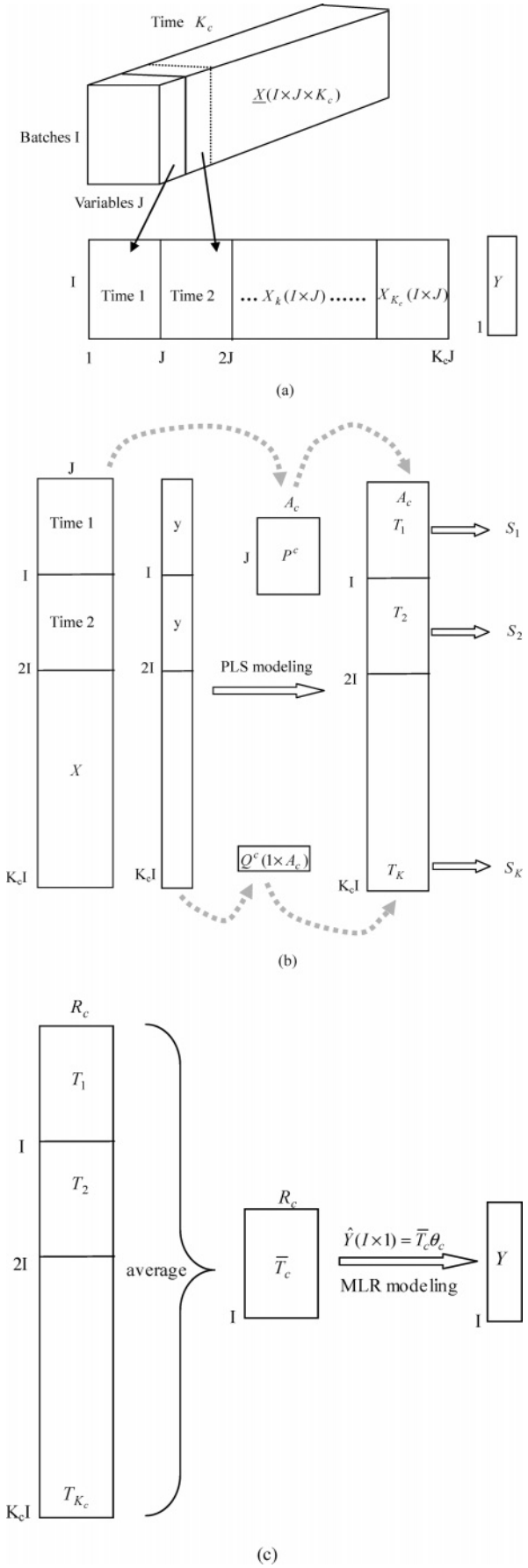

Figure 1. New approach of PLS based on phase partition: (a) data preprocessing on each time-slice matrix; (b) process modelling based on variablewise unfolding; and (c) end-of-phase quality prediction using phase-specific score average trajectory.

First, in each phase, the normalized time-slices matrix is rearranged into a variable-unfolding matrix $\tilde{\mathbf{X}}_c(IK_c \times J)$ (K_c is the number of samplings belonging to phase c). In this way, extended samples composed of batches and time series from the same phase offer more sufficient process information. Moreover, a simpler regression structure can be obtained since only J process variables are employed as input in the regression model. In order to carry out the following PLS regression analysis, the number of rows of the quality variable should be properly rearranged to make a consistent size with that of $\tilde{\mathbf{X}}_c$-$(IK_c \times J)$. As shown in Figure 1b, the normalized quality variable of I batches are duplicated by K_c to form $\tilde{\mathbf{Y}}_c(IK_c \times 1)$. The data is structured in this way because quality at the end of the batch run is accepted if the process operation up to the current time point k is normal.

Next, as shown in Figure 1b, the scores matrix for $\tilde{\mathbf{X}}_c(IK_c \times J)$, $\mathbf{T}^c(IK_c \times A_c)$, and the phase-representative loadings matrices, respectively, for process measurements and quality variables, $\mathbf{P}^c(J \times A_c)$ and $\mathbf{Q}^c(1 \times A_c)$, are extracted using PLS analysis. Then one strategy must be adopted to decide the number of retained latent variables. From eq 5, one possible solution is to choose a threshold level and, when $||F||$ goes below the threshold, the recursive procedure ends and the current number of latent variables is set as the final one. Another possibility is to look at the difference between the current and previous $||F||$ values and determine the number of components when this becomes small. A combination of threshold and difference methods would be preferable.[30] If higher prediction accuracy is desired, another class of method, called cross-validation, must be used to determine the number of latent variables needed. However, in our present work, since the ability of online process monitoring and fault detection is preferred rather than quality prediction, the check of $||F||$ is enough for the selection of latent variables to make sure that all the normal process variations closely linearly related to quality are covered in the monitoring models. Moreover, it is simpler and more straightforward without increasing computation and treatment complexity. Of course, in actual application, the combination of some alternative methods may be preferable under some specific conditions.

Instead of only focusing on the feature information at any separate time, the phase-representative models take the entire history trajectories within the same phase into account and reveal the phase-specific overall correlations among process variables more relevant to product quality. The scores matrix $\mathbf{T}^c(IK_c \times A_c)$ is a good summary of the important information contained in the original process measurements. It has strong explanations to quality variation, and its rows cover the major dynamic trajectory along both time and batch directions. Then it is separated into $\mathbf{T}_k(I \times A_c)$ corresponding to each time. The covariance matrix at each time, $\mathbf{S}_k(A_c \times A_c)$, is calculated from the scores matrix, $\mathbf{T}_k(I \times A_c)$. Thus, the dynamics of the scores can be revealed in the Hotelling-T^2 monitoring chart.[14,28]

In the above way, the quality can be predicted online at any time. However, it is worth remarking that the final quality is not dependant upon the separate process measurement only at a single time but should be settled together by the overall process operation state within the same phase. Therefore, the prediction based on individual sampling time may not be accurate enough. However, the directions that the quality variable will take can be trusted in general and can help considerably in detecting and diagnosing the source of the abnormality.

3.2. Quality Prediction Based on Phase-Specific Average Trajectory. In real industrial processes, it is common that the process measurement at each time should have certain percent

explanation and contribution to response variables, showing phase-specific accumulation effects along time in each phase. With the execution of the above-mentioned clustering algorithm, the process is partitioned into several phases associated with process time. Thus, the phase-specific score average trajectory, $\bar{\mathbf{T}}^c(I \times A_c)$, revealing the pivotal information in the cth phase, can be utilized as the modeling input, which can be steadily obtained by simply averaging all $\mathbf{T}_k(I \times A_c)$ within the same phase c obtained from the first-level statistical analysis without increasing the heavy calculation burden and complexity. Meanwhile, it is possible to reveal the cumulative effects of process variations on quality.

$$\bar{\mathbf{T}}^c(I \times A_c) = \frac{1}{K_c}\sum_{k\in c}\mathbf{T}_k = \frac{1}{K_c}\sum_{k\in c}(\tilde{\mathbf{X}}_k \cdot \mathbf{R}^c) = \left(\frac{1}{K_c}\sum_{k\in c}\tilde{\mathbf{X}}_k\right)\cdot\mathbf{R}^c = \bar{\mathbf{X}}_c(I \times J)\cdot\mathbf{R}^c \quad (7)$$

where $\bar{\mathbf{X}}_c(I \times J)$ is the phase-specific average process trajectory.

Consequently, as shown in Figure 1c, some alternative methods, such as multiple linear regression (MLR), principal component regression (PCR), and PLS, can be employed to simply get the phase-specific quality prediction. Here, multiple linear regression (MLR) algorithm can be simply and directly performed on the phase-specific average scores, $\bar{\mathbf{T}}^c(I \times A_c)$, and the quality variable, $\tilde{\mathbf{Y}}(I \times 1)$, to extract the phase-specific regression relationships:

$$\begin{aligned}\tilde{\mathbf{Y}}(I \times 1) &= \bar{\mathbf{T}}^c\boldsymbol{\theta}^c + \mathbf{E}^c \\ \boldsymbol{\theta}^c &= (\bar{\mathbf{T}}^{c\mathrm{T}}\bar{\mathbf{T}}^c)^{-1}\bar{\mathbf{T}}^{c\mathrm{T}}\tilde{\mathbf{Y}}\end{aligned} \quad (8)$$

where $\boldsymbol{\theta}^c(A_c \times 1)$ is the phase-specific regression parameter vector and $\mathbf{E}^c$ is the residual in phase c.

Obviously, eq 8 also hints at the most frequent problem in MLR: the inverse of $\bar{\mathbf{T}}^{c\mathrm{T}}\bar{\mathbf{T}}^c$ may not exist.[31] As mentioned before, $\bar{\mathbf{T}}^c(I \times A_c)$ are obtained by simply averaging all the $\mathbf{T}_k$-$(I \times A_c)$ within the same phase c, where $\mathbf{T}_k(I \times A_c)$ actually denotes the orthogonal critical features at each time and thus excludes the collinearity problem. Then, as their average, $\bar{\mathbf{T}}^c(I \times A_c)$, phase-specific latent features generally may not seriously linearly depend on each other. Moreover, it satisfies $A_c \ll I$, which meets the condition that there always have to be at least as many samples as variables so as to make the least-squares solution possible. Hence, we can approximately assume that $\bar{\mathbf{T}}^{c\mathrm{T}}\bar{\mathbf{T}}^c$ is full rank for simplicity. However, when the reversibility cannot be exactly satisfied, we could introduce simple modifications, such as working in the invertible subspace by retaining the proper number of latent features or adopting partial least-squares (PLS) to fit the regression parameters.

From the above, it is obvious that, in each phase, two kinds of different-level PLS quality prediction models are developed. The first-level PLS regression models are constructed to reveal and monitor the progressing trend that the quality variable will move forward by predicting quality at each time. Although it may be impossible to get an extremely precise prediction result, however, the process can be judged whether it is tracking the desired operation trajectory by checking whether the quality is following the expected varying range. In the second level, end-of-phase representative regression models are designed using the phase average score trajectory for more credible prediction of quality than the first-level models. They have simpler model structures and emphasize particularly on the phase-specific accumulative effects of process measurements on quality. Whenever the current phase is completed, the end-of-phase quality prediction can be readily obtained. It is obvious that the real-time and end-of-phase quality predictions in the two levels hold different purposes and, thus, stress different aspects of statistical analysis. It is natural that the effective combination of the two-level models may enhance one's process analysis and understanding.

According to the execution of kernel PLS algorithm,[32] two types of covariance matrices are actually employed in PLS analysis to reveal the regression model structure. They are the covariance matrix between process measurements, $\mathbf{S_{XX}}$, which indicates the underlying process correlation information, and the covariance matrix between process variables and quality, $\mathbf{S_{XY}}$, which exposes the correlation relationship between process variations and quality. In the first-level PLS regression analysis, we can deduce that using variablewise $\tilde{\mathbf{X}}_c(IK_c \times J)$ and the rearranged quality vector $\tilde{\mathbf{Y}}_c(IK_c \times 1)$, the covariance matrices are constructed as follows:

$$\begin{aligned}\tilde{\mathbf{X}}_c^{\mathrm{T}}\tilde{\mathbf{X}}_c &= \sum_{k\in c}(\tilde{\mathbf{X}}_k^{\mathrm{T}}\tilde{\mathbf{X}}_k) \\ \tilde{\mathbf{X}}_c^{\mathrm{T}}\tilde{\mathbf{Y}}_c &= \sum_{k\in c}(\tilde{\mathbf{X}}_k^{\mathrm{T}}\tilde{\mathbf{Y}}_k)\end{aligned} \quad (9)$$

From eq 9, we are aware that the covariance structure, $\tilde{\mathbf{X}}_c^{\mathrm{T}}\tilde{\mathbf{X}}_c$, is actually the sum of process correlations belonging to the same phase. In this way, the phase-specific regression relationship is extracted covering the overall phase.

In the second-level regression analysis, phase-specific score average trajectory, $\bar{\mathbf{T}}^c(I \times A_c)$, is employed in the PLS modeling, which exhibits the critical variation information closely linearly related with quality. Thus, it is actually equal to feature extraction and data compression prior to modeling, which eliminates the impact of harmful noises. Moreover, the second-level models can capture phase-specific accumulative effects on quality prediction, which is a consequence of averaging phase-specific score trajectory. Therefore, the relative importance of each process variable to the quality variable is identified from an "overall" phase-specific perspective.

In conclusion, by means of the full use of phase-specific average process trajectory, the extracted regression relationship can stack the accumulative effects of process variations on quality along time progressing within the same phase in a simpler structure. Meanwhile, the relative importance of each explanatory variable to the explained variable is identified from an "overall" phase-specific perspective.

3.3. Online Monitoring Charts for First-Level Quality Prediction. For online monitoring, some conventional multivariate statistics in PCA such as SPE and T^2 can be introduced straightforwardly into PLS and the corresponding control limits can be determined directly as well. Here, it should be noted that, because of the use of time-varying score covariance, T^2 can be defined corresponding to each time as follows and its control limits can be obtained from the indicated F-distribution,

$$T_k^2 = (\mathbf{t}_{\mathrm{new},k} - \bar{\mathbf{t}}_k)\mathbf{S}_k^{-1}(\mathbf{t}_{\mathrm{new},k} - \bar{\mathbf{t}}_k)^{\mathrm{T}} \approx \frac{A_c(I^2-1)}{I(I-A_c)}\mathbf{F}_{A_c,I-A_c} \quad (10)$$

where $\mathbf{t}_{\mathrm{new},k}(1 \times A_c)$ is the score of the new batch at time k, $\bar{\mathbf{t}}_k(1 \times A_c)$ is the mean of the scores matrix $\mathbf{T}_k(I \times A_c)$, and $\mathbf{S}_k(A_c \times A_c)$ is the covariance matrix of $\mathbf{T}_k(I \times A_c)$. In the

proposed method, $\bar{\mathbf{t}}_k(1 \times A_c)$ is a zero vector since the time-slice data $\mathbf{X}_k(I \times J)$ has been mean-centered in the preprocessing procedure.

Similar with PCA, in PLS, fault operation will result in either larger scores in the principal subspace or larger residuals in the original process measurement space, or both. However, the systematic variations and the residual variations supervised by the Hotelling-T^2 and SPE statistics, respectively, only deliver directly the process variations of measurement trajectory. Although the variations are closely related with the quality, they are not direct and convenient for one to observe the influence of process variations on quality. As aforementioned, the purpose of online monitoring is to enhance the reproducibility of product quality. Thus, it can give a more visible and straightforward observation and explanation if the predicted quality based on first-level PLS models can be visually monitored online so that the operator can catch the performance of process operation in convenience. Here, a new control limit is developed based on the first-level quality prediction values derived from modeling batches to online directly detect process deviations relevant to off-spec quality, which is different from the confidence interval developed by Nomikos and MacGregor.[10] In their work, they have been concerned with the quantitative estimate of the uncertainty of quality prediction under normal operation conditions in terms of the confidence interval since it was fully recognized that the estimated quality might differ from the real one. So their confidence interval, which the real quality should lie in, was actually calculated using the new quality prediction at each time for the current new batch and, thus, was unknown after the current time. Therefore, it was unable to be employed for process monitoring. Comparatively, in our present work, the control limits for quality prediction are derived from the quality predictions of reference batches for the purpose of process monitoring. The calculation reveals the fact that, for normal batch processes, the quality prediction values should vary commonly within the normal control regions, where the fluctuation may be caused by the common stochastic process variations. The control limits for quality prediction can, thus, be successfully employed in process monitoring as a standard for fault detection by inspecting whether the online quality prediction is going beyond the control limit. This provides another statistical criterion differing from the conventional statistics, such as SPE and Hotelling-T^2, in process monitoring.

In conventional PCA/PLS methods, all the modeling batches are assumed to be collected under normal operation conditions. A normal batch implies that the process follows a set of predetermined sequences with acceptable process variations within a finite duration to convert raw materials to products. A high degree of reproducibility is necessary to obtain successful batches. Batch-to-batch variations from the mean trajectories are caused by stochastic factors, i.e., the successful batches in the modeling database are deemed to be subject to common-cause variations. Therefore, people often inherit this conclusion and easily deduce that the batch-to-batch variations of process measurements at the same time approximately follow normal distribution without strict theoretical deduction and proof. In fact, as the linear combination of process variables, the quality predictions don't exactly conform to norm-distribution, which can be illustrated in the latter simulation. Hence, its control limit cannot be determined directly using the normal distribution rule. Otherwise, the credibility of the control limits will be compromised and, thus, the reliability of the multivariate monitoring system may be deteriorated. On such a situation, alternative approaches have to be investigated to more accurately and reasonably define the normal regions, such as the Box−Cox approach,[33] hypothesis-based statistical test,[34] percentile approach,[35] likelihood-based approach using the bootstrap and nonparametric empirical density estimates,[36,37] and so on.

In our present work, a kernel-based density estimation method is employed to calculate the distribution rule of quality prediction. A univariate kernel estimator is defined as

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left\{\frac{x - x_i}{h}\right\} \tag{11}$$

where x is the data point under consideration; x_i is the observation value from the data set; h is the window width, also known as the smoothing parameter; n is the number of observations; and $K\{\cdot\}$ is the kernel function. Here, Gauss kernel function $K(x) = 1/\sqrt{2\pi}\ \exp(-x^2/2)$ is chosen to act in the kernel-based density estimation. $\hat{f}(x)$ is the density value evaluated at the data point x.

The kernel estimator is, therefore, a sum of "bumps" located at the observations. The kernel K defines the shape of the bumps while the smoothing parameter h determines their width. Different kernel functions express different ways in which each sample point is assigned to the density with different contribution weights based on distance dissimilarity to x. However, the type of kernel function is not the key factor to influence the density estimation. Generally, the estimate is derived based on a Gauss kernel function. Like the histogram, the window width, h, has the dominant effect on the density smoothing. The problem of choosing how much to smooth is of crucial importance in density estimation. If h is chosen too large, there will be more samples related with the density of x, which share the similar weights to x without obvious distinctions. Therefore, $\hat{f}(x)$ will be oversmoothed, erasing detail. If h is too small, only a few points will assume the weights to x in each considered window unit h. In this way, their weights will be greatly impacted by distance, showing larger differences. Therefore, $\hat{f}(x)$ will be undersmoothed, failing to filter out spurious detail. A number of alternative measures exist to estimate h, the window width or smoothing parameter.[37] The more advanced methods are based on cross-validation, such as least-squares cross-validation.

The control limits used in quality prediction monitoring charts can be commonly obtained using kernel density estimation like this: First, the quality predictions at each time interval k are obtained corresponding to different modeling batches and will vary around the average level commonly. Then, the univariate kernel density estimator is used to estimate the density function of the normal quality predictions. The points, occupying the 99.5% area and 0.5% area of the obtained density function, can be calculated as the upper and lower limits of normal quality prediction, respectively, where too large and too small quality prediction beyond the upper and lower limits are both regarded as abnormal outliers. In this way, the control limits represent the 99% coverage of normal variations of the predicted quality at each time.

One major advantage of kernel density estimation is that it needs no specific statistical distribution in prior and the determined control limits follow the data more closely than those obtained on the basis of specific distribution.[37] With enough training samples, no matter which kernel function is adopted, a reliable estimation result can be finally obtained in theory. However, the number of samples needed in the kernel density estimation is relatively large. When the number of reference batches is not enough, it may not be possible to calculate a

reliable enough density estimation result. Therefore, two different treatment ways are developed, respectively, in case of two different types of circumstances.

For the first case, the number of batches used during modeling is sufficient enough, which generally should be >50.[38,39] The density estimation can be simply obtained based on all the reference batches, and then the control limits can be naturally calculated based on the estimated density distribution.

For the second case, the number of reference batches is not sufficient. The control limits can, thus, be calculated using a jackknife procedure in which each of the reference batches is left out once, and the density distribution is estimated based on the remaining batches. Therefore, there will be the same number of density estimation results as the number of reference batches. The mean of all the estimated density distributions can then be utilized as the representative density to calculate the final control limits.

As pointed out in Section 3.2, the online predictions of first-level PLS models might not be accurate enough since they only make use of the single sampling, whereas the entire operation state of each phase should be in charge of the final quality. However, tracking the prediction trajectory can well-indicate the developing trend that the quality will take along process evolvement. The fault information can be detected by checking whether the predicted quality is surpassing the control limits, revealing that the process violations are spoiling the quality. Therefore, compared with the conventional Hotelling-T^2 and Q statistics, the monitoring chart of quality prediction can more straightforwardly inspect the impacts of those unusual process behaviors on the quality. With online application, the monitoring charts of the traditional Hotelling-T^2 and Q and the quality prediction should be combined together and compared with each other to judge whether the process is tracking the expected trajectory so as to make the monitored result more reliable and comprehensive. It is usual that the three charts should all yield alarms when the process is departing from the normal state. However, it is possible that not all of these plots can indicate process upset under some certain instance. Thus, it is necessary for the operator to check and analyze further the process operation.

It has to be emphasized that, once a fault has been detected by inspecting the traditional SPC charts, indicating the process is not following the expected operation trajectory, the quality predictions have to be treated with caution. The predicted values may be inaccurate; however, the trends of the curves can be very helpful in showing how the quality will deteriorate.

3.4. Contribution Plots of Quality Prediction. Through statistical process monitoring charts, it is possible to detect the variations of process variables in a new batch that are different from common-cause process variations. However, the monitoring charts can only acquaint one with the occurrence of more unusual events than the common-cause variations captured in the NOC process regions. The monitoring charts cannot provide the information on what is wrong with the process or which measurement variables make the dominant contributions to the process upsets. When the monitored statistics go beyond the control limits, revealing the occurrence of some abnormality, the contribution plot,[40,41] a commonly used diagnosis tool, is often used to check the underlying fault variables. However, the previous work only examined the contributions of process variables to the process variations by defining the contribution plots of T^2, SPE, or scores. Here, to more directly reveal the process variables mainly responsible for nonconforming quality, the contribution of each process variable to quality prediction is first studied in the present work. On the basis of eq 3, the contribution calculation is defined as follows,

$$C_{ijk} = [0, \cdots, x_{i,j,k}, \cdots, 0]\cdot\boldsymbol{\beta}^c \tag{12}$$

where the subscripts i, j, and k denote the ith batch, the jth process variable, and the kth sampling time, respectively. The regression parameter of size $J \times 1$, $\boldsymbol{\beta}^c$, in phase c, to which the current kth sampling belongs, is defined as $\mathbf{W}^c(\mathbf{P}^{c\mathrm{T}}\mathbf{W}^c)^{-1}\mathbf{Q}^{c\mathrm{T}}$, and C_{ijk} is the contribution to the quality of the jth variable at time k for the ith batch, $x_{i,j,k}$.

The contribution value of the process variable $x_{i,j,k}$ to the quality, as presented in eq 12, can be positive or negative. This implies that the former holds a positive relationship with quality, i.e., it varies along the consistent ascending or descending trend with quality. Comparatively, the latter possesses a negative correlation with quality, i.e., it moves toward the opposite direction different from quality changing. However, it should be noted that it is not necessary that those variables holding larger contributions are surely the dominating causes of process fault because they may have greater shares even under normal conditions. There is inevitably individual subjectivity if only checking the absolute contribution value of each process variable. Thus, a critical level should be defined to provide a quantitative reference standard.

Recently, control limits have been suggested for contribution plots of T^2 and SPE.[29,41] These control limits are used to compare the absolute size of contributions in the new batch with the magnitude of reference contributions in the normal modeling batches. If in the reference data a certain process variable has high contribution values, it can also be expected to have large ones in the new batch. However, if a certain process variable possesses a large share of contribution in the current new batch, but a small amount in the reference data, this probably is due to abnormality in the current process operation. Therefore, not only the absolute contribution values but also the relative size compared with the normal values should be examined. Using the control limits, it is easier to quantitatively find those process variables that are really different from the common ones in modeling batches.

Here, the common batch-to-batch variations will be used to establish the control limits of quality prediction contribution statistic at each time. The control limits are calculated in the same way as those of quality prediction, i.e., using the kernel density estimation. However, instead of summing all the predictor variables at each time for quality prediction, it only focuses on separate process variable. With online monitoring, it is usually easy to identify a more specific location of off-spec quality in a process by the phase-based statistical analysis. Then, combining contribution plots of quality prediction as well as their control limits, one can directly zoom in during this period to inspect the contribution of each process variable to the predicted quality so as to find those responsible ones. This can enhance the process understanding and help considerably in diagnosing the source of the abnormality.

3.5. Outline of the Proposed Algorithm for Batch Processes. We have developed phase-based multiple PLS modeling, online monitoring, and prediction strategy to supervise the progress of a batch process. Now the procedure of modeling and online application can be summarized as follows:

A. Developing statistical models

1. Normalize each time-slice data matrix $\mathbf{X}_k(I \times J)$, as well as the final quality variable $\mathbf{Y}(I \times 1)$.

2. Relate normalized data matrix $\tilde{\mathbf{X}}_k(I \times J)$ with $\tilde{\mathbf{Y}}(I \times 1)$, so that PLS performed on them can get K weighted loadings

matrices $\check{\mathbf{P}}_k$, which represent the process correlations closely related with the final quality at each time interval.

3. Cluster the weighted loadings matrices $\check{\mathbf{P}}_k$ to get C different phases.

4. In each phase, rearrange the normalized time-slice matrices variablewise, forming $\tilde{\mathbf{X}}_c(IK_c \times J)$ and $\tilde{\mathbf{Y}}_c(IK_c \times 1)$.

5. Extract the phase-specific weights matrix, $\mathbf{R}^c(J \times A_c)$, which can be used to compute the phase-representative scores matrix, $\mathbf{T}^c(IK_c \times A_c)$, directly from $\tilde{\mathbf{X}}_c(IK_c \times J)$; and the loadings matrices, respectively, for $\tilde{\mathbf{X}}_c$ and $\tilde{\mathbf{Y}}_c$, $\mathbf{P}^c(J \times A_c)$, and $\mathbf{Q}^c(1 \times A_c)$, by performing PLS regression analysis.

6. Separate the scores matrix $\mathbf{T}^c(IK_c \times A_c)$ into $\mathbf{T}_k(I \times A_c)$ corresponding to each time interval k and calculate $\mathbf{S}_k(A_c \times A_c)$, the time-varying covariance matrix of $\mathbf{T}_k$.

7. Obtain steadily the phase-specific average score trajectory, $\bar{\mathbf{T}}^c(I \times A_c)$, by averaging all the $\mathbf{T}_k(I \times A_c)$ belonging to the same phase c and then conduct multiple linear regression (MLR) directly on the phase-based average score, $\bar{\mathbf{T}}^c(I \times A_c)$, and the quality variable, $\tilde{\mathbf{Y}}^c(I \times 1)$, to extract the phase-specific regression relationship.

8. Calculate the control limits of quality prediction at each time k as well as those of conventional T^2 and SPE statistics.

9. Obtain contribution of each predictor variable to the response variable and their control limits.

B. Online application and analysis

1. For new sampling data at time k, $\mathbf{x}_{\text{new},k}(1 \times J)$, normalize it using the same mean and standard deviation obtained in the modeling procedure.

2. According to process time, project the normalized new data $\tilde{\mathbf{x}}_{\text{new},k}$ onto the corresponding model $\mathbf{R}^c$ to calculate the score vector $\mathbf{t}_{\text{new},k} = \tilde{\mathbf{x}}_{\text{new},k}\mathbf{R}^c$.

3. Calculate the T^2 and SPE statistics,

$$\mathbf{T}_{\text{new},k}{}^2 = (\mathbf{t}_{\text{new},k} - \bar{\mathbf{t}}_k)S_k^{-1}(\mathbf{t}_{\text{new},k} - \bar{\mathbf{t}}_k)^{\text{T}},$$

$$\text{SPE}_{\text{new},k} = (\hat{\mathbf{x}}_{\text{new}.k} - \tilde{\mathbf{x}}_{\text{new}.k})(\hat{\mathbf{x}}_{\text{new}.k} - \tilde{\mathbf{x}}_{\text{new}.k})^{\text{T}}$$

where $\hat{\mathbf{x}}_{\text{new},k} = \mathbf{t}_{\text{new},k}\mathbf{P}^c$.

4. Calculate the quality prediction $\hat{y}_{new,k}$ at each time k: $\hat{y}_{new,k} = \tilde{\mathbf{x}}_{\text{new},k}\mathbf{R}^c\mathbf{Q}^{c\text{T}}$.

5. Conduct process monitoring by judging whether the quality prediction at the current time, $\hat{y}_{new,k}$, is exceeding its control limit combined with the online monitoring of T^2 and SPE statistics.

6. For a normal process, whenever a phase is completed, calculate the phase-specific average score trajectory $\bar{\mathbf{t}}^c_{\text{new}}(1 \times A_c)$ to forecast the end-of-phase quality, $\hat{y}^c_{\text{new}} = \bar{\mathbf{t}}^c_{\text{new}}\theta^c$.

7. If an abnormality is detected, identify the variables closely relevant to the quality aberrancy by combining the contributions as defined in eq 12 with their control limits determined using kernel density estimation.

The modeling procedure in each phase is illustrated in Figure 1.

4. Illustration and Discussion

4.1. Fed-Batch Penicillin Fermentation Process. In this section, the proposed method is applied to a well-known benchmark process, the fed-batch penicillin fermentation process.[42,43] A flow diagram of the penicillin fermentation process is given in Figure 2. The production of secondary metabolites such as antibiotics has been the subject of many studies because of its academic and industrial importance. Here, we focus on the process to produce penicillin, which has nonlinear dynamics and multiphase characteristics. In a typical operating procedure for the modeled fed-batch fermentation, most of the necessary cell mass is obtained during the initial preculture phase. When most of the initially added substrate has been consumed by the microorganisms, the substrate feed begins. The penicillin starts to be generated at the exponential growth phase and continues to be produced until the stationary phase. A low substrate concentration in the fermentor is necessary for achieving a high product formation rate due to the catabolite repressor effect. Consequently, glucose is fed continuously during fermentation instead of being added one-off at the beginning.

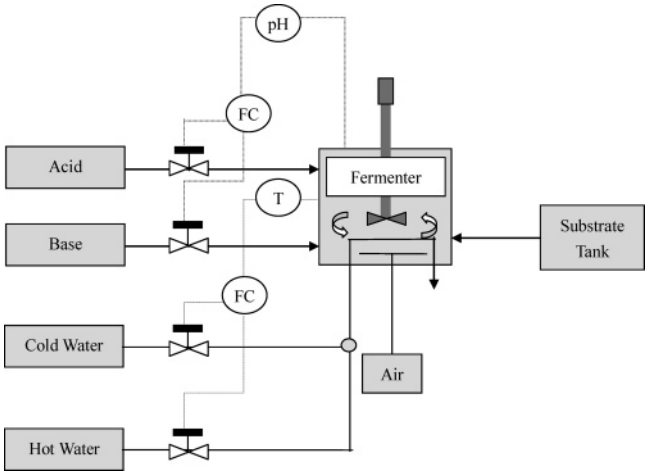

Figure 2. Flow diagram of the penicillin fermentation process.

Table 1. Variables Used in the Monitoring of the Benchmark Model

no.	variable's descriptions
	Process Variables
1	aeration rate (L/h)
2	agitator power (W)
3	substrate feed rate (L/h)
4	substrate feed temperature (K)
5	dissolved oxygen concentration (g/L)
6	culture volume (L)
7	carbon dioxide concentration (g/L)
8	pH
9	fermentor temperature (K)
10	generated heat (kcal)
11	cooling water flow rate (L/h)
	Quality Variable
	penicillin concentration (g/L)

In the present simulation experiment, a total of 40 reference batches are generated using a simulator (PenSim v1.0 simulator) developed by the monitoring and control group of the Illinois Institute of Technology. These simulations are run under closed-loop control of pH and temperature, while glucose addition is performed open-loop. Small variations are automatically added by the simulator to mimic the real normal operating conditions. The process variables and quality variable selected for modeling in this work are listed in Table 1. The only knowledge required for PLS regression modeling, process monitoring, and diagnostic analysis is a good historical database of normal or "in-control" process operation where good productivity and quality have been achieved. In the present simulation experiment, the duration of each batch is 400 h, consisting of two physical operation phases, a preculture phase of ~45 h and a fed-batch phase of ~355 h. With 1 h as the sampling interval, 400 points of 11 process variables collected in each batch and 40 normal experiments carried out under the same default initial conditions yield the process array $\underline{\mathbf{X}}(40 \times 11 \times 400)$. The quality is only measured at the end of process, thus generating the quality vector $\mathbf{Y}(40 \times 1)$. These data are selected to represent normal process

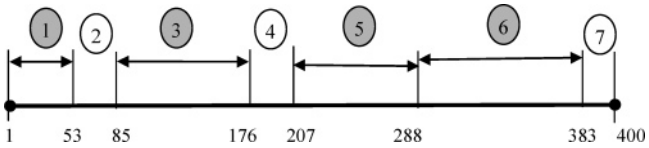

Figure 3. Phase-clustering result for penicillin cultivation trajectory.

operation when only common variations are present and acceptable product quality is achieved. The 40 reference batches are used for modeling, process analysis, and knowledge extraction. Moreover, 25 normal batches are generated to test the performance of the developed models.

4.2. Online Application and Process Analysis. According to the modeling procedure, first 400 normalized time slices $\tilde{\mathbf{X}}_k$-($I \times J$) are obtained from $\underline{\mathbf{X}}(40 \times 11 \times 400)$. Then PLS is applied to the normalized data sets, $\{\tilde{\mathbf{X}}_k, \tilde{\mathbf{Y}}\}$, generating 400 weighted time-slice loadings matrices, $\tilde{\mathbf{P}}_k(J \times J)$, which are fed to the clustering algorithm introduced in Section 3.1. The phase division result is shown in Figure 3, which clearly reveals that, without using any prior process knowledge, the fermentation process is automatically divided into seven phases in detail. The result reflects the changing trend of process underlying behaviors closely linearly related with the quality along the time direction, among which four long phases (marked with shadow circle, phases 1, 3, 5, and 6) correspond to the four main subphases. A few short temporary phases (marked with shadowless circle) emerge between the four main phases, corresponding to the transition regions from one major phase to the next. Moreover, a short time region is separated at the end of process as an individual phase. The more elaborate phase-partition results emphasize the changes of process correlations relevant to the quality rather than only the real physical operation, which can not only benefit more detailed analyses of underlying process behaviors but also enhance process monitoring and prediction performance by establishing multiple specific statistical models. Without losing generality, each phase hints at different effects on the final product as well as at different correlations between process measurement and product.

After the phase partition, the MVPLS models are established focusing on each phase and employed for process monitoring, where the retained number of latent variables in each phase are 4, 3, 3, 6, 4, 5, and 5, respectively. Therefore, one can calculate at each time interval the system variations and the residual variations represented by the T^2 and SPE statistics, respectively, and the predicted quality. In Section 3.2, we have analyzed that the quality predictions are not always exactly Gaussian distributed. Figure 4 shows the graphical normality testing result, taking an example for the quality prediction at the 33rd time interval acquired from the reference modeling data. A normal probability plot is a useful graph for assessing whether data comes from a norm-distribution. If the data does come from a normal distribution, the plot will appear linear. Other probability density functions will introduce curvature in the plot. From the example shown in the plot, it is clear that the values do not strictly follow a norm-distribution. Consequently, the calculation of control limits based on norm-distribution is not strictly reliable and well-founded, which may compromise the credibility of monitoring performance. Therefore, it is necessary to determine the control limits of quality prediction using an alternative method, such as nonparametric kernel density estimation method in the present work.

Then those 25 normal testing batches are employed to verify the performance of the MVPLS models. On the basis of quality prediction index as well as the conventional statistics, i.e., *D*- and *Q*-statistics, the effective combination of the three statistics can provide more reliable and detailed process information and analysis guidelines for the current running batch. Figure 5 shows the online monitoring results of T^2, SPE, and quality prediction for one successful batch, where the time trajectories of monitored values stay well within the control bounds of normal operation for each type of statistic, indicating that the batch behaves free of any process abnormality throughout the batch run. Especially, by comparison, we have simultaneously plotted two types of control limits with 5% significance level in Figure 5b. One is calculated using the kernel-based estimation method and the other is derived from normal-distribution, where the former seems to reveal a more detailed changing trend than the latter. Along the process evolvement, the quality predictions of the normal batch commonly vary within the expected control limits obtained using kernel-based method but go beyond those calculated based on the norm-distribution rule now and then, especially during the period spanning 300−375 h, thus delivering false alarms. From the plot, we can visually validate the rationality of determining control limits of quality prediction based on kernel-density estimation. Moreover, it is clear that the online quality prediction shows a correct developing trend, which is approximating and reaching the actual final quality value.

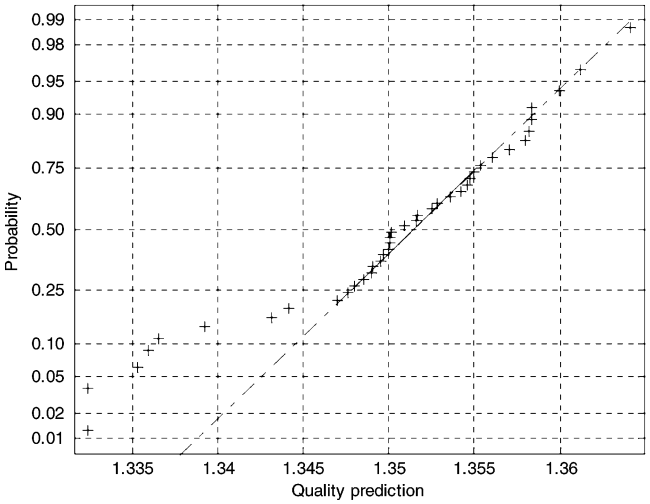

Figure 4. Normal probability check at the 33rd time interval for quality prediction.

In the movement of process operation, whenever a subphase is completed, the phase-specific average score trajectory $\bar{\mathbf{T}}^c(I \times A_c)$ is steadily obtained using eq 7, which is employed to more accurately forecast the final quality. To illustrate the quality prediction performance based on phase-specific average score trajectory, the prediction result for another normal batch is exhibited in Figure 6, revealing the different phase-specific prediction abilities. For the four main phases (phases 1, 3, 5, and 6), the maximum predicted error is 0.33% in phase 6, which is acceptable in real industrial application. Table 2 summarizes the quality prediction results of the 25 normal testing batches. The popular performance indices, mean squared errors (MSEs), are employed to evaluate the prediction power of second-level models in each phase, which is modified and calculated as $\mathrm{MSE}_c = ((\mathbf{Y} - \hat{\mathbf{Y}}^c)^{\mathrm{T}}(\mathbf{Y} - \hat{\mathbf{Y}}^c))/I$ in our study ($\mathbf{Y}$ is the real final quality and $\hat{\mathbf{Y}}^c$ is the predicted quality in the cth phase). Meanwhile, the prediction error rate (%) values are also listed in Table 2, taking an example for 10 of 25 normal testing batches for brevity. In addition, the MSE values obtained from the 25 normal test batches, respectively, using the first-level models,

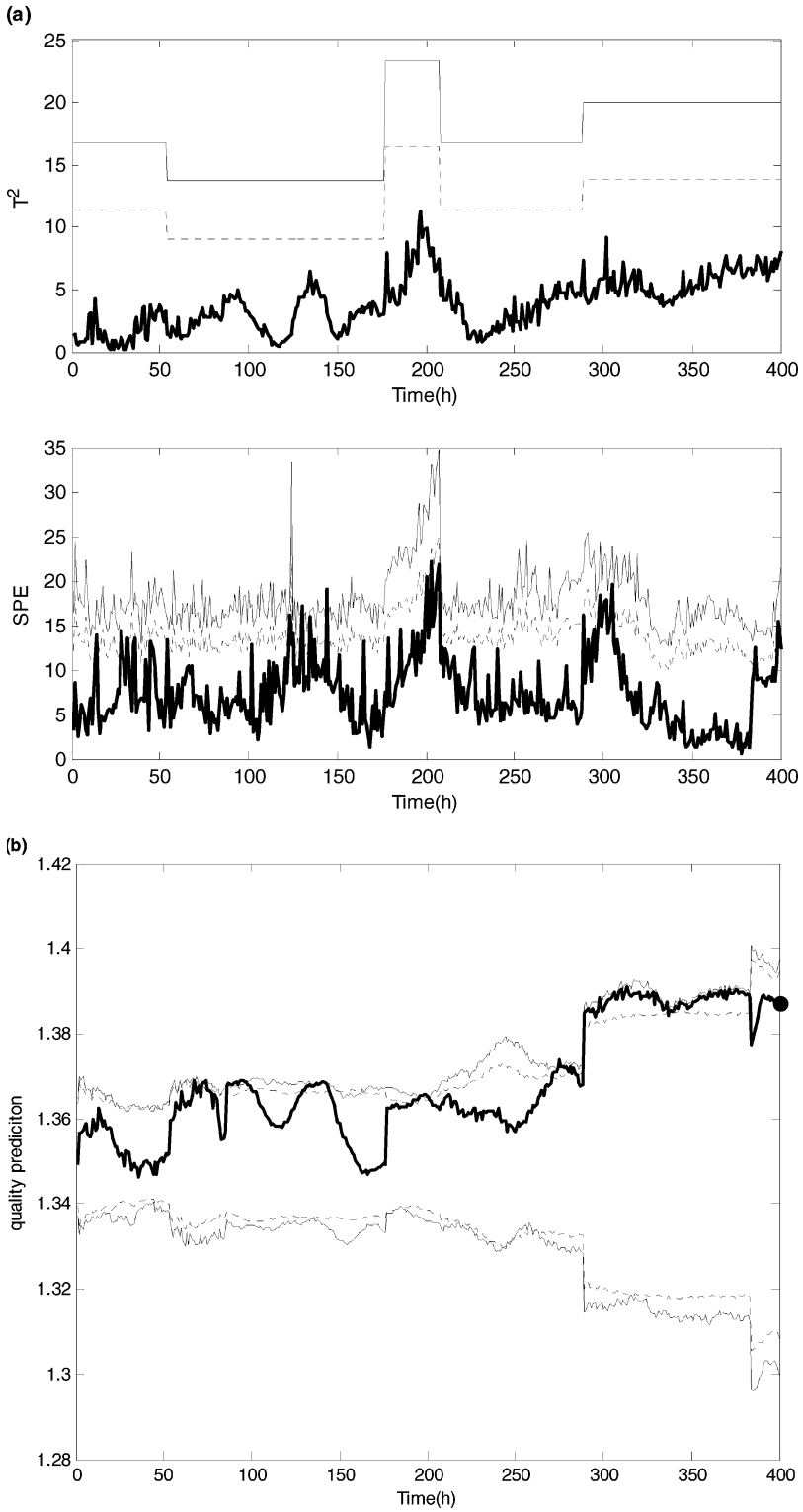

Figure 5. Monitoring charts for a normal batch using (a) T^2 and SPE statistics (solid line, 99% control limit; dashed line, 95% control limit; bold solid line, on-line T^2 and SPE statistics) and (b) quality prediction metric (solid line, control limit using kernel estimation; dashed line, control limits using normal distribution; bold solid line, on-line quality prediction value; (●), the real final quality measurement).

the second-level models, and the unfold-PLS method by Nomikos and MacGregor[10] are comparatively plotted along the time direction in Figure 7. In our study, the unknown future measurements are filled with zeros. Moreover, for the convenience of comparison, the phase-specific MSE values obtained from second-level quality prediction are stretched along the sampling time. It can be clearly seen that MSE values of second-level PLS models are lower than those of the other two types of models, revealing better precision of quality prediction. Both of the figures and the table illustratively demonstrate that the second-level PLS models show superiority and applicability for prediction of quality variables that have significant accumulative correlations with process measurements within each phase.

To gain more insight into the performance of the proposed method for online monitoring and fault detection, five batches are generated, respectively, under each of the three abnormal

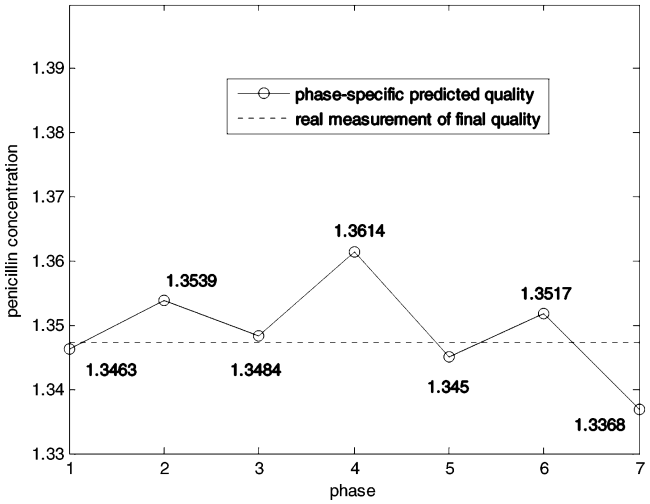

Figure 6. Quality prediction based on phase-specific average score trajectory (−○−, the predicted values; dashed line, the real final quality measurement).

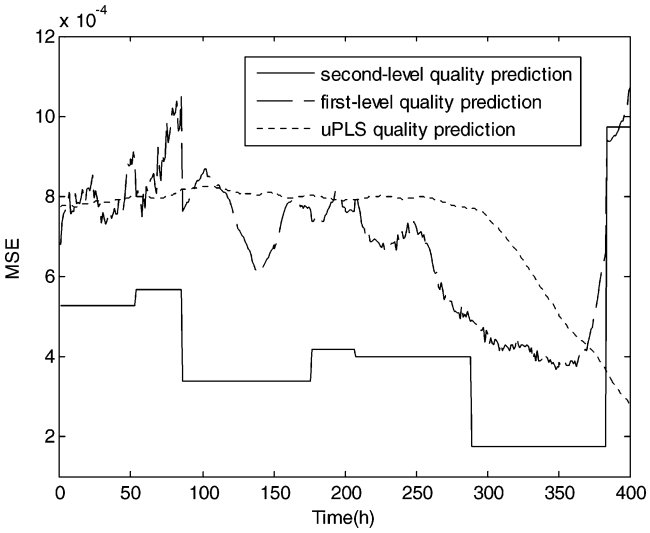

Figure 7. Quality prediction comparison using MSE index for test batches (solid line, MSE index from second-level models; (— —), MSE index from first-level models; dashed line, MSE index from uPLS model).

conditions (faults 1−3) presented in Table 3. The first fault is implemented by introducing a step decrease into the agitator power from 30 h until the end of the process. The monitoring results are shown in Figure 8. This upset can be quickly picked up by all three monitoring charts, where the statistic values obviously go beyond the control limits almost immediately once the fault occurs. Thus, the monitoring models effectively capture the unusual batch variations. In fact, for the bad batch, since the process characteristics relevant to quality are no longer valid after the occurrence of fault with poor explanation to quality, the quality prediction derived from them will move out of the normal region, which thus will indicate whether the process variations are deteriorating the final quality.

For the second fault, a ramp decrease is imposed on the batch process, i.e., the substrate feed rate is linearly decreased from 0.043 to 0.035 L/h from 70 h to the end of the process operation. A decrease in the substrate feed rate results in a reduction of penicillin concentration since substrate is the main carbon source to be fed during the fermentation,[43] although the process variation itself is slight. As shown in Figure 9, process variations are soon detected since those pieces of process information closely related with quality are especially emphasized so that

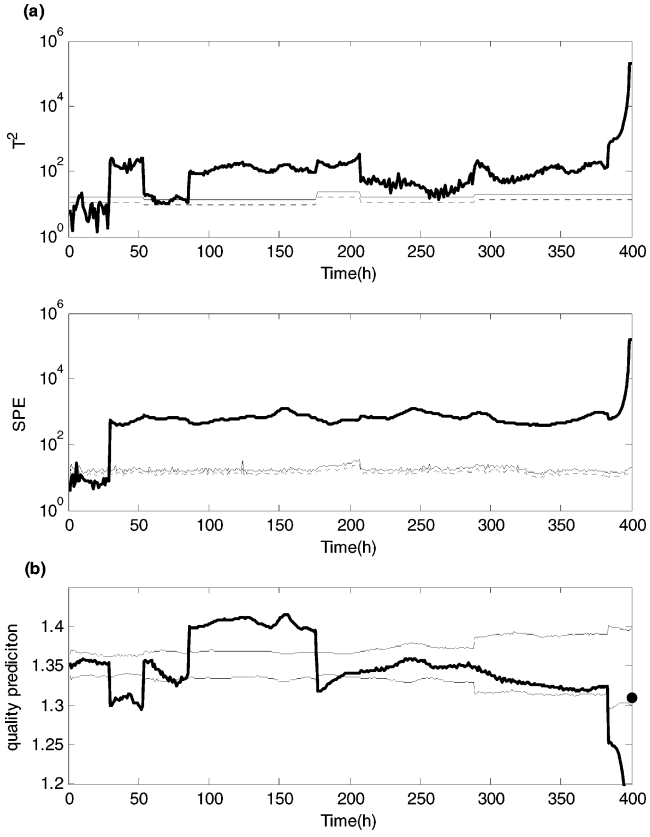

Figure 8. Monitoring charts for fault 1 using (a) T^2 and SPE statistics and (b) quality prediction metric (solid line, 99% control limit; dashed line, 95% control limit; bold solid line, on-line T^2 and SPE statistics, or quality prediction metric; (●), the real quality measurement).

the slight process disturbance is unveiled as soon as predicted quality is corrupted, showing obvious slopping over the normal range. Comparatively, the monitoring results using the sub-PCA modeling method,[25,26] based on the three-phase partition result, are also given in Figure 10. The T^2 statistic fails to detect the process outlier with small magnitude, indicating that the process systematic variations captured by sub-PCA models may not be focused on those closely linearly related with the quality, and thus, the small process disturbance cannot be tracked successfully. Moreover, without using the time-varying score covariance structures, the lack of dynamics of scores in the Hoteling-T^2 chart might also hold up the timely reflection of process violation. The fault detection results show that, under some situations, the sub-PCA method might not perform well to detect certain small process upsets since it does not directly model those process variations linearly correlated with quality.

In the case of fault 3, temperature control failure is introduced by turning off the temperature controller at the very beginning of the process operation. The monitoring results for fault 3 are shown in Figure 11, which deliver obvious and clear fault indications agreeing well with the real case. All the above simulation illustrations have effectively demonstrated that, based on the MVPLS method, one can easily track the progress of each batch run and detect the occurrence of the faults with simple monitoring charts.

Moreover, to gain more insight into the performance of the proposed method for batch-process monitoring, it is compared with conventional uPCA and uPLS algorithms for both normal and fault cases. Two performance indices are used to evaluate how well each approach performs: the overall type I error (OTI)[9,23,28] and the relative action signal time (AST).[28,44]

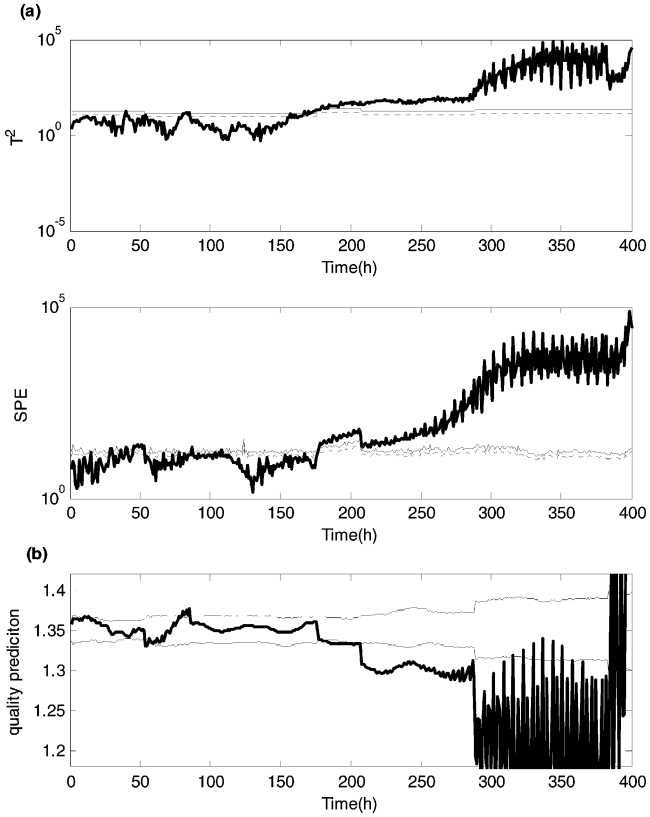

Figure 9. Monitoring charts for fault 2 using (a) T^2 and SPE statistics and (b) quality prediction metric (solid line, 99% control limit; dashed line, 95% control limit; bold solid line, on-line T^2 and SPE statistics, or quality prediction metric).

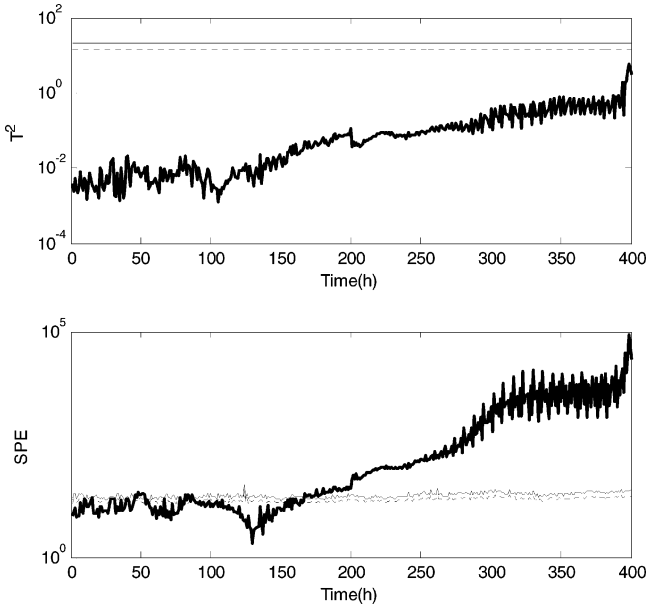

Figure 10. Monitoring charts for fault 2 using sub-PCA modeling (solid line, 99% control limit; dashed line, 95% control limit; bold solid line, on-line T^2 and SPE statistics).

Following their definition, the OTI is the proportion of faults in the batches representing normal operation conditions: OTI = nf/I·K, where nf is the total number of values of monitoring statistic outside the control limits in the Normal Operation Conditions (NOC) set. The AST is defined as the time elapsed between the introduction of an error and the out-of-control signal in the control chart. Thus, both the false alarms and the time

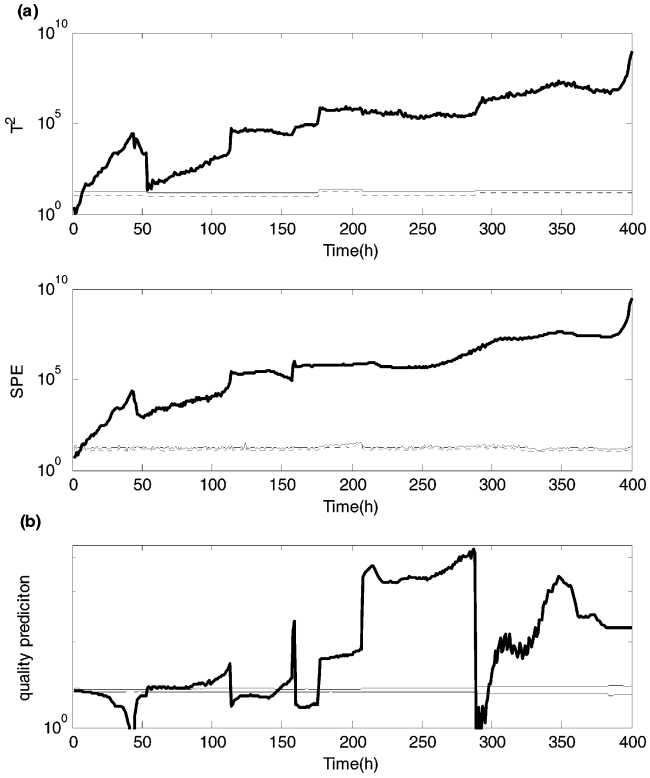

Figure 11. Monitoring charts for fault 3 using (a) T^2 and SPE statistics and (b) quality prediction metric (solid line, 99% control limit; dashed line, 95% control limit; bold solid line, on-line T^2 and SPE statistics, or quality prediction metric).

delay in detecting the fault can be availably taken into account. The simulation comparison results summarized in Table 4 support the effectiveness of the proposed method. Compared with conventional uPCA and uPLS approaches, it shows the similar monitoring performance or even better result under some situations.

Most of the time, it is important to know which physical variables cause the quality spoilage. This can be done by interrogating the underlying model and checking the contribution of each process variable to the detected deviation of quality. Once the abnormal condition is detected by the monitoring charts, the contribution plot is used to analyze the fault cause, which can enhance the process understanding and improve the performance of fault diagnosis. The contributions of process variables to quality in the case of the three types of faults are illustrated in Figure 12 as well as the control limits. For fault 1, taking the contribution at 60 h as an example, the diagnosis result is illustrated in Figure 12a. From the plot, if there is no control limit imposed on the contribution plot, pH (variable 8) has the largest contribution rate to the current predicted quality. In view of this, one will possibly make an arbitrary judgment that variable 8 rather than variable 2 acts as the dominant fault cause. However, combined with their control limits, it can be revealed that the contribution of variable 8 stays well within the normal coverage and variable 2 escapes from its expected operation boundary. Thus, the analysis result displays that the primary fault variable should be agitator power (variable 2), which correctly responds to the real process failure. For fault 2, Figure 12b indicates that, at time 200, three variables, i.e., substrate feed rate (variable 3), culture volume (variable 6), and generated heat (variable 10), migrate outside the desired control intervals, which should be the fault variables. Moreover, it is noteworthy that, from the contribution plot of fault 2, it can be

Table 2. Prediction Performance for 25 Normal Testing Batches

Err(%)	I	II	III	IV	V	VI	VII
1	0.945 09	0.683 95	−0.178 12	0.291 19	0.385 72	0.042 03	−0.660 54
2	0.361 38	−0.420 07	−0.416 54	0.221 34	−0.408 23	0.291 19	0.297 59
3	0.294 25	0.493 61	−0.501 65	−0.624 05	−0.712 86	−0.435 00	1.190 47
4	−0.669 12	−0.655 82	−0.932 14	−0.278 51	−0.608 31	−0.635 98	−0.937 98
5	0.043 97	−0.055 78	−0.093 88	0.589 81	−0.953 08	−0.467 05	0.262 85
6	0.632 91	−0.246 33	0.409 75	−0.672 44	−0.334 77	−0.381 85	0.162 79
7	0.786 98	−0.532 41	−0.061 77	−0.223 82	0.627 48	1.003 60	−1.301 14
8	0.314 98	1.250 80	0.306 64	−0.537 79	0.533 35	−1.575 01	0.036 30
9	−0.125 24	−0.680 27	−0.704 97	−0.419 62	−1.309 60	0.423 24	1.016 17
10	−0.688 62	−0.014 39	−0.286 64	−0.925 69	0.529 88	0.150 55	−0.882 72
MSE	0.000 525 93	0.000 564 8	0.000 337 13	0.000 417 06	0.000 401 09	0.000 175 58	0.000 973 78

Table 3. Summary of Fault Types Introduced at Different Time of Fermentation

fault no.	fault type	occurrence (h)
1	15% step decrease in agitator power	30
2	0.002 ramp decrease in substrate feed rate	70
3	temperature controller failure	0

Table 4. Comparison Result of Monitoring Performance (Using 99% Control Limits)

	performance indices					
	OTI			AST		
methods	D-st	SPE	quality	D-st	SPE	quality
uPCA	0.0211	0.0597		0.0200	0.0255	
				0.2940	0.3915	
				0.0205	0.0295	
uPLS	0.0210	0.0753		0.0125	0.0210	
				0.2335	0.3250	
				0.0275	0.0315	
proposed method	0.0205	0.0562	0.0571	0	0	0
				0.1460	0.1645	0.1490
				0.0155	0.0135	0.0105

inferred that the deviation of substrate feed rate affects other variables, especially culture volume (variable 6) and generated heat (variable 10). Since the processes variables tend to be complicatedly related to each other in the practical batch industry, the deviation of one variable will naturally cause the others to move out of control from different aspects and to various extents. With the development of time, if fault detection is delayed, the fault diagnosis will be more complicated when various variables indicate significant deviations from their expected values. Hence, fast fault detection is important for correct fault diagnosis and will help the operator to check the fault cause in time and reduce the economic loss. With respect to fault 3, the diagnosis figure at time 140 exposes that culture volume (variable 6), pH (variable 8), and generated heat (variable 10) have the largest contribution values and also go beyond normal control limits, which in fact reflects the effects of temperature-control failure. The above diagnosis and analysis results based on contribution plots may draw such a conclusion that the relative, not absolute, contribution value of each process variable should be supervised so as to check the fault cause correctly. This is derived from such a fact that, under normal conditions, different variables possess diverse contribution abilities to different extents. Those variables with higher contributions in the reference data should also have larger values in the current new batch if the process is free of any abnormality. Thus, one cannot simply infer that the variables with the higher contributions should be responsible for the violation of predicted quality. Likewise, those variables with lower contribution values are not necessarily excluded from the disorder variables, i.e., smaller absolute contribution values do not mean that the corresponding variables are not responsible for the process failure. One correct and reliable conclusion can be drawn only if the absolute contributions are compared with the reference standard of control limits. In conclusion, using the contribution plots, although it is still not capable of unambiguously diagnosing the cause of an event (this can only be achieved by statistical analysis incorporating exhaustive fundamental process knowledge), one can clearly focus on a particular phase of the operation process and on a small number of specific variables located in that region, thereby making it much easier for operators to employ their knowledge to diagnose possible assignable causes.

Moreover, taking the first six process variables for example, their control limits of contributions to quality prediction along with process progress are presented in Figure 13 combined with the division of process phases. From it, an interesting phenomenon is exposed that, within the same phase, their explanation

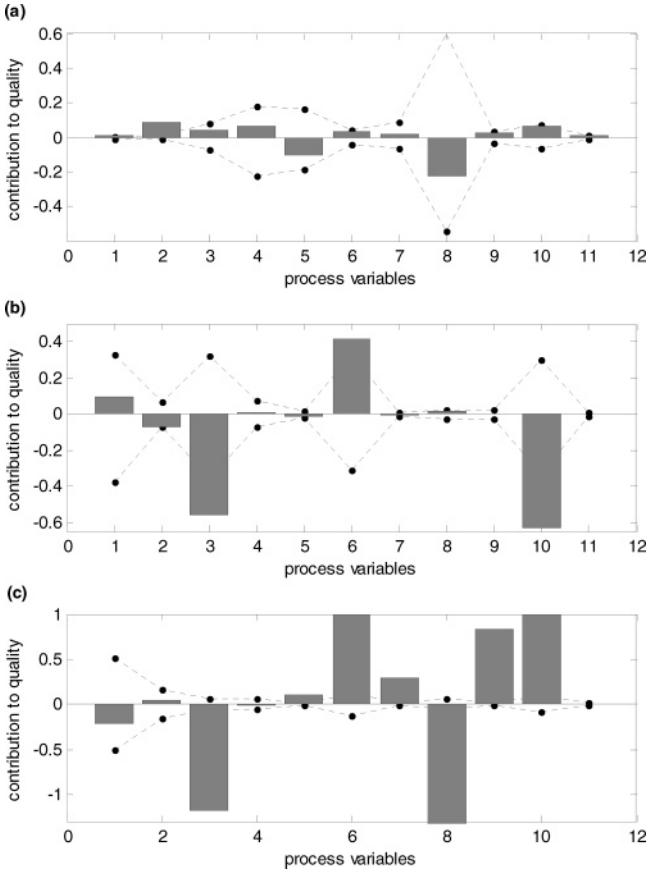

Figure 12. Variable contributions to predicted quality in case of (a) fault 1 introduced at 30 h; (b) fault 2 introduced at 70 h; and (c) fault 3 introduced at 0 h ((--·--), 99% control limit).

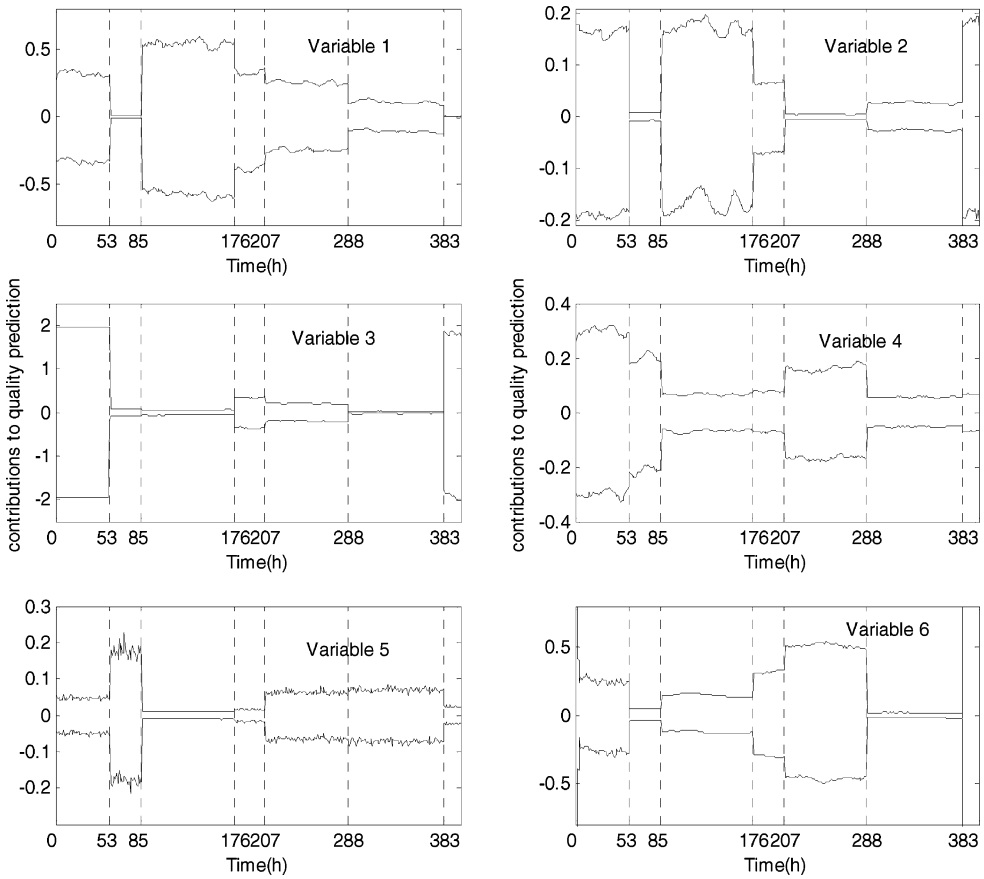

Figure 13. Control limits for contributions of process variables to quality prediction (solid line, 99% control limit).

abilities to the quality remain similar, revealing the local phase-specific effects of process variable trajectory on final quality from another aspect.

5. Conclusion

A new approach to multivariate batch process modeling, monitoring, and quality prediction is described based on phase-specific statistical analysis for batch processes with multiphase characteristics. On the basis of the phase-specific analysis, the key idea of the proposed method is to develop two-level PLS regression models for different statistical analysis. On the first level, it focuses on the monitoring of the individual time point using multiple PLS models. Control limits for quality prediction index intensify the detection of process failures that are closely linearly related with quality deterioration. With the completion of each phase, the second-level models are developed using phase-specific score average trajectory. By means of the two kinds of models with different levels, one can integrate the real-time monitoring with accurate quality prediction to jointly ensure the process evolvement with desired product property. The proposed method not only can give a valid monitoring scheme but also allows accommodating the phase-specific accumulative effects of process variables on quality. The applications to the typical fed-batch fermentation process show that the proposed method is effective.

In conclusion, the proposed algorithm has provided a significant attempt to conduct process monitoring, diagnosis, and quality prediction from the point of view of tracking quality performance, whose further application will be promising and should have more efforts devoted to it in future.

Acknowledgment

The work was supported in part by the National Natural Science Foundation of China (No. 60374003 and No. 60774068) and Project 973 (No. 2002CB312200), China.

Appendix

Modified *k*-Means Clustering Algorithm.

Inputs: the patterns to be partitioned, $\{\check{P}_1, \check{P}_2, ..., \check{P}_k\}$, and the threshold θ for cluster elimination.

Outputs: the number of clusters C, the cluster centers $\{W_1, W_2, ..., W_C\}$, and the cluster index of $\check{P}_k$, $m(k)$.

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

1. Choose C^0 ($i = 0$) cluster centers W_c^0 ($c = 1, 2, ..., C^0$) from the K patterns along the time series. Practically, the initial cluster centers can be assumed to be uniformly distributed in the pattern set.

2. Merge pairs of clusters whose intercenter distance, dist(W_{c1}^{i-1}, W_{c2}^{i-1}), is below the predetermined threshold θ.

3. Calculate the distances from each pattern $\check{P}_k$ to all of the centers, dist($\check{P}_k$, W_c^{i-1}), assign $\check{P}_k$ to the nearest center $W_{c^*}^{i-1}$, and denote its cluster index as $m(k) = c^*$.

4. Eliminate the clusters that catch few patterns after a set number of iterations $i > I_{\_num}$ to avoid singular clusters.

5. Update the number of cluster centers to be C^i and recompute the new cluster centers W_c^i ($c = 1, 2, ..., C^i$), using the current cluster index, $m(k)$.

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.

Remark: The detailed programs for implementing the proposed algorithm and the experimental data are not included in the present work because of lack of space. A more complete document will be provided upon request addressed to the first author.

Literature Cited

(1) Kourti, T.; MacGregor, J. F. Process analysis, monitoring and diagnosis, using multivariate projection methods. *Chemom. Intell. Lab. Syst.* **1995**, *28*, 3.

(2) Kosanovich, K. A.; Dahl, K. S.; Piovoso, M. J. Improved process understanding using multiway principal component analysis. *Ind. Eng. Chem. Res.* **1996**, *35*, 138.

(3) Wold, S.; Kettaneh, N.; Fridén, H.; Holmberg, A. Modelling and diagnostics of batch processes and analogous kinetic experiments. *Chemom. Intell. Lab. Syst.* **1998**, *44*, 331.

(4) Louwerse, D. J.; Smilde, A. K. Multivariate statistical process control of batch processes based on three-way models. *Chem. Eng. Sci*. **2000**, *55*, 1225.

(5) Chen, G.; McAvoy, T. J. Process control utilizing data based multivariate statistical models. *Can. J. Chem. Eng.* **1996**, *74*, 1010.

(6) Kourti, T. Multivariate dynamic data modeling for analysis and statistical process control of batch processes, start-ups and grade transitions. *J. Chemom.* **2003**, *17*, 93.

(7) Wold, S.; Esbensen, K.; Geladi, P. Principal Component Analysis. *Chemom. Intell. Lab. Syst*. **1987**, *2*, 37.

(8) Nomikos, P.; MacGregor, J. F. Monitoring of batch processes using multiway principal component analysis. *AIChE J.* **1994**, *40*, 1361.

(9) Nomikos, P.; MacGregor, J. F. Multivariate SPC charts for monitoring batch processes. *Technometrics* **1995**, *37*, 41.

(10) Nomikos, P.; MacGregor, J. F. Multiway partial least squares in monitoring batch processes. *Chemom. Intell. Lab. Syst*. **1995**, *30*, 97.

(11) Wise, B. M.; Gallagher, N. B.; Butler, S. W.; White, D. D., Jr.; Barna, G. G. A comparison of principal component analysis, multiway principal components analysis, trilinear decomposition and parallel factor analysis for fault detection in a semiconductor etch process. *J. Chemom*. **1999**, *13*, 379.

(12) Wang, X. Z. *Data mining and knowledge discovery for process monitoring and control*; Springer: London, 1999.

(13) Smilde, A. K. Comments on three-way analyses used for batch process data. *J. Chemom.* **2001**, *15*, 19.

(14) Lee, J.-M.; Yoo, C. K.; Lee, I.-B. Enhanced process monitoring of fed-batch penicillin cultivation using time-varying and multivariate statistical analysis. *J. Biotechnol.* **2004**, *110*, 119.

(15) Chen, J. H.; Liu, K.-C. On-line batch process monitoring using dynamic PCA and dynamic PLS models. *Chem. Eng. Sci.* **2002**, *57*, 63.

(16) Ündey, C.; Cinar, A. Statistical monitoring of multistage, multiphase batch processes. *IEEE Control Syst. Mag.* **2002**, *22*, 40.

(17) Kosanovich, K. A.; Piovoso, M. J.; Dahl, K. S. Multi-Way PCA Applied to an Industrial Batch Process. In Proceedings of the American Control Conference, Baltimore, MD, June 1994; p 1294.

(18) Dong, D.; McAvoy, T. J. Multi-stage Batch Process Monitoring. In Proceedings of the American Control Conference, Seattle, WA, June 1995; p 1857.

(19) Kourti, T.; Nomikos, P.; MacGregor, J. F. Analysis, monitoring and fault diagnosis of batch processes using multiblock and multiway PLS. *J. Process Control* **1995**, *5*, 277.

(20) Zhao, S. J.; Zhang, J.; Xu, Y. M. Monitoring of Processes with Multiple Operating Modes through Multiple Principle Component Analysis Models. *Ind. Eng. Chem. Res.* **2004**, *43*, 7025.

(21) Zhao, S. J.; Zhang, J.; Xu, Y. M. Performance monitoring of processes with multiple operating modes through multiple PLS models. *J. Process Control* **2006**, *16*, 763.

(22) Yuan, B.; Wang, X. Z. Multilevel PCA and inductive learning for knowledge extraction from operational data of batch processes. *Chem. Eng. Commun*. **2001**, *185*, 201.

(23) Camacho, J.; Picó, J. Online monitoring of batch processes using multi-phase principal component analysis. *J. Process Control* **2006**, *16*, 1021.

(24) Camacho, J.; Picó, J. Multi-phase principal component analysis for batch processes modeling. *Chemom. Intell. Lab. Syst*. **2006**, *81*, 127.

(25) Lu, N. Y.; Gao, F. R.; Wang, F. L. A sub-PCA modeling and on-line monitoring strategy for batch processes. *AIChE J*. **2004**, *50*, 255.

(26) Lu, N. Y.; Yang, Y.; Gao, F. R.; Wang, F. L. Stage-based multivariate statistical analysis for injection molding. Proceedings of International symposium on advanced control of chemical processes, Hong Kong, 2003; p 471.

(27) Westerhuis, J. A.; Kourti, T.; Macgregor, J. F. Comparing alternative approaches for multivariate statistical analysis of batch process data. *J. Chemom.* **1999**, *13*, 397.

(28) van Sprang, E. N. M.; Ramaker, H.-J.; Westerhuis, J. A.; Gurden, S. P.; Smilde, A. K. Critical evaluation of approaches for on-line batch process monitoring. *Chem. Eng. Sci*. **2002**, *57*, 3979.

(29) Ündey, C.; Ertunc, C. S.; Cinar, A. Online Batch/Fed-batch Process performance monitoring, Quality Prediction, and Variable-Contribution Analysis for Diagnosis. *Ind. Eng. Chem. Res*. **2003**, *42*, 4645.

(30) Geladi, P.; Kowalski, B. R. Partial least-squares regression: A tutorial. *Anal. Chim. Acta* **1986**, *185*, 1.

(31) Belsley, D.; Kuh, E.; Welsch, R. *Regression diagnostics: Identifying influential data and sources of collinearity*; Wiley: New York, 1980.

(32) Dayal, B. S.; MacGregor, J. F. Improved PLS Algorithms. *J. Chemom.* **1997**, *11*, 73.

(33) Box, G. E. P.; Cox, D. R. An analysis of transformation. *J. R. Stat. Soc., Ser. B* **1964**, 211.

(34) He, P. *Mathematical Statistics and Multivariate Statistics*; Southwest Jiaotong University Press: Chengdu, China, 2004.

(35) Levinson, W. Approximate confidence limits for Cpk and control limits form non-normal process capabilities. *Quality Eng.* **1997**, 635.

(36) Chen, Q.; Wynne, R. J.; Goulding, P.; Sandoz, D. The application of principal component analysis and kernel density estimation to enhance process monitoring. *Control Eng. Pract*. **2000**, *8*, 531.

(37) Martin, E. B.; Morris, A. J. Non-parametric confidence bounds for process performance monitoring charts. *J. Process Control* **1996**, *6* (6), 349.

(38) Li, Y. Q.; Liu, H. Y.; Zhao, L. W. *Non-parametric statistical methods*; Southwest Jiaotong University Press: Chengdu, China, 1998.

(39) Zhang, J.; Yang, H. X. *Multivariate statistical process control*; Chemical Industry Press: Beijing, China, 2000.

(40) Miller, P.; Swanson, R. E.; Heckler, C. E. Contribution plots: A missing link in multivariate quality control. *Appl. Math. Comput. Sci*. **1998**, *8*, 775.

(41) Westerhuis, J. A.; Gurden, S. P.; Smilde, A. K. Generalized contribution plots in multivariate statistical process monitoring. *Chemom. Intell. Lab. Syst*. **2000**, *51*, 95.

(42) Birol, G.; Ündey, C.; Parulekar, S. J.; Cinar, A. A morphologically structured model for penicillin production. *Biotechnol. Bioeng.* **2002**, *77*, 538.

(43) Birol, G.; Ündey, C.; Cinar, A. A modular simulation package for fed-batch fermentation: penicillin production. *Comput. Chem. Eng*. **2002**, *26*, 1553.

(44) Ramaker, H.-J.; van Sprang, E. N. M.; Westerhuis, J. A.; Smilde, A. K. Fault detection properties of global, local and time evolving models for batch process monitoring. *J. Process Control* **2005**, *15*, 799.

Received for review June 2, 2007
Revised manuscript received October 11, 2007
Accepted October 15, 2007

IE0707624