Quality Prediction Based on Phase-Specific Average Trajectory for Batch Processes

Chunhui Zhao, Fuli Wang, Zhizhong Mao, Ningyun Lu, and Mingxing Jia
School of Information Science and Engineering, Northeastern University, Shenyang, Liaoning Province, P. R. China

DOI 10.1002/aic.11405
Published online January 18, 2008 in Wiley InterScience (www.interscience.wiley.com).

A new process analysis and quality prediction scheme is presented, based on phase-specific average process trajectory, and is developed for the improvement of quality prediction in multiphase batch processes. After the process trajectory is separated into different phases, based on the different inherent process correlations relevant to quality, quality prediction is performed using average trajectory focusing on critical phases. In this way, it explores the phase-specific cumulation effects of process variables on product quality with a simpler regression model. Then the critical-to-quality phases are identified, and the online quality predicting algorithm is conducted correspondingly during those critical phases. Moreover, to improve the estimation precision of unknown data in each phase, the past batches are made use of, and the missing observations are complemented by searching for the most similar process trajectory to the current one. The applications of the proposed scheme to injection molding show its effectiveness and feasibility. © 2008 American Institute of Chemical Engineers *AIChE J,* 54: 693–705, 2008

Keywords: quality prediction, phase-specific average trajectory, critical-to-quality phases, phase-specific cumulation effects

Introduction

In order to meet the need of constantly changing market situations, batch and semibatch processes play an important role in most industries. This especially comes true in the processes mainly involved in the production and processing of low-volume and high-value-added products, including certain polymers, specialty chemicals, pharmaceuticals, biochemicals, etc. Characterized by the precise sequencing and automation of all phases in sequence, the objective of batch processes is to achieve consistent and reproducible quality at competitive prices within the finite duration. Rapidly changing market competition and demand for consistent and high-quality products have spurred the development of quality-related investigations for batch processes.

Due to complicated nonlinear process behaviors, the absence of online quality measurements, and the redundancy of high-dimensional correlated process variables, however, online quality prediction and control in batch processes suffer a lack of reproducibility from batch-to-batch variations. It is, thus, necessary to make significant efforts for the development of methods for quality prediction, allowing us to estimate quality variables in a simpler, faster, but more accurate way. Recently, multivariate statistical procedures for monitoring the progress of processes have been widely developed, among which PCA[1] and PLS[2] are the most popular. Multiway principal component (MPCA), and multiway partial-least squares (MPLS) modeling pioneered by Nomikos and MacGregor[3,4] are applied in batch processes, to extract directly useful underlying information from process measurements with little prior process knowledge. Conventional MPLS modeling uses process trajectories over the entire batch course as the input to pick up those process variations that are most predictive of the quality variables. Therefore, it exposes such a concept that the final quality should depend on the whole process trajectory, that is, the time cumulative

Correspondence concerning this article should be addressed to F. Wang at flwang@mail.neu.edu.cn
N. Lu is also affiliated with the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu Province, P.R. China

effects of process behaviors on quality can be more explored. However, when three-way batch data structure is unfolded batch-wise into two-way form, the number of input predictors in regression model dramatically increases with high-autocorrelation and cross-correlation complexity. Although MPLS is well-known as an effective data compression technique, one cannot expect superior feature extraction results from such redundant data information, thus, deteriorating the performance of quality prediction. In fact, the final product quality is mainly directly determined by some critical time regions during a batch cycle, and closely related with only a small portion of process measurements.[5–8] MPLS is inefficient in revealing the local time-specific effects of process variables on the final quality. Recently, many works[5–8] have been reported focusing on the time-specific effects in process interpretation and prediction. Duchesne and MacGregor[5] proposed a new pathway multiblock PLS algorithm. They incorporated information provided by intermediate quality measurement to help in identifying the time-specific effects of trajectory features on quality associated with stages of operation in which different physical phenomena dominate. However, for most industrial processes, online measurements of intermediate quality are rarely available, which prevents its further application. Bootstrapping-based generalized variable selection of Chu et al.[6] can correctly extract quality-related variables from typical "fat-type" unfolded batch data with limited samples, and isolate the local effects of process variables on the final quality. They extended PLS/MPLS modeling for prediction improvement by focusing on the critical-to-quality time periods[7] to enhance process interpretation and analysis. Considering that multiplicity of phase is an inherent nature of many batch processes, and process variables of different phases may have different effects on the final product quality, a stage-based process analysis strategy has been developed by Lu and Gao.[7,8] In their method, the quality prediction is allowed to be conducted online at each separate sampling time, which can get earlier quality prediction without having to wait until the end of process operation. The representative stage PLS model obtained from averaging time-slice PLS models[7,8] within the same stage, however, overlooks the time-cumulative effects within the same phase. In real multiphase batch processes, product quality tends to be determined together by the operation state of the entire critical phase rather than each separate sampling time. Therefore, it is necessary to analyze the time accumulation phenomenon in detail, and develop the corresponding quality prediction algorithm.

In this article, a phase-based quality prediction modeling method is developed for multiphase batch processes. Inherited from the clustering algorithm by Lu et al.[9] the batch process is automatically and properly divided into different phases, revealing different process correlation characteristics. Then in each phase, instead of modeling isolated at an individual time point,[7,8] the proposed method employs the phase-representative average trajectory to reveal the phase-specific cumulation effects of process variations on quality. To improve the precision of future data estimation in each phase, the past batch trajectories are made use of, and the unknown observations are complemented by the most similar batch trajectory through comparing the new current batch trajectory with those in the history library. In this way, it effectively considers the process dynamics along time sensitive enough to process variations. Moreover, the critical-to-quality phases are identified, and then the online quality prediction algorithm is developed correspondingly, focusing on the critical phases. The proposed process analysis and quality prediction method can overcome many shortcomings of conventional MPLS, and has the following advantage: By means of phase-specific average trajectory, it allows one to conduct detailed statistical process analysis focusing on each phase in simpler regression model structure, which differs from the redundant "fat" form of batch-wise unfolding in conventional MPLS modeling. Meanwhile, it covers the cumulation correlations between process variables and quality from an "overall" phase-specific perspective.

This article is organized as follows. First, the details of the proposed method are described. Then, the effectiveness and feasibility of the proposed predicting method are illustrated by applying it to the injection molding. Finally, conclusions are drawn in the last section.

Methodology

Multiway partial-least squares (MPLS)

MPLS is an extension of PLS to handle three-dimensional (3-D) data arrays. In each batch run, assume that J variables are measured at $k = 1,2,\ldots,K$ time instances throughout the batch. Then vast amount process data collected from similar I batches can be organized as a three-way array X ($I \times J \times K$), and a corresponding quality variable Y vector of dimension $I \times 1$. Here, it should be noted that in this article, we shall treat only the case with univariate dependent variable; for the multivariate case one just needs to treat each of dependent variables separately. The relation between MPLS and PLS is that MPLS is equivalent to performing ordinary PLS on a large 2-D unfolded matrix. Batch-wise unfolding has been widely used to analyze batch process data in many previous studies. The idea of quality predicting based on these data arrangement reveals that the final product quality is determined together by the entire process operation trajectory, and each operation time interval imposes different influences on the quality. In order to eliminate the influence of nonlinearity and different measuring scale for further modeling analysis, the two-way unfolded matrix are mean centered and scaled to unit variance. Then PLS can be performed as follows[10,11]

$$X(I \times JK) = TP^T + E, \quad Y(I \times 1) = TQ^T + F \qquad (1)$$

where T is given by

$$T(I \times A) = XW(P^TW)^{-1} \qquad (2)$$

This decomposition summarizes and compresses the process observations over the entire running duration into low-dimensional space that is most relevant to the final product quality. Each row of the score matrix $T(I \times A)$, corresponds to a single batch and depicts the overall variability of this batch cycle with respect to the other batches in the database. $P(JK \times A)$ and $W(JK \times A)$, loading and weight matrices for $X(I \times JK)$, respectively, summarize the process variations and give the weights applied to obtain the t-score for that batch. $Q(1 \times A)$, loading matrix for $Y(I \times 1)$, relates the process variations with the final product quality. A is the retained number of latent variables.

One major problem in the online predicting of quality based on MPLS is that it assumes that the prediction relationship remains constant throughout the entire batch course, without reflecting the dynamics of correlations between process variables and quality along time evolving. Especially for multiphase batch processes, it cannot reveal the common phase-specific effects on quality. Moreover, one cannot expect that the pivotal process information closely related to the final quality can be extracted correctly enough from such a redundant "fat" form of batch-wise unfolding, although MPLS is a well-known data compression technique.

Modeling based on phase-specific average process trajectory

Concept of phase division algorithm. In statistical analysis for batch processes, the three-way array $X(I \times K \times J)$ can be rearranged into 2-D data structure $X(I \times JK)$, using batch-wise unfolding, where each row accounts for the process variable trajectories of each batch throughout the entire run duration. With this kind of unfolding, one can investigate the variations over different batches. Moreover, such kind of data rearrangement requires that the phases obtained from different batches should be equal or should not change significantly, so that the influence of uneven phases can be neglected. However, if uneven-phase problem is serious, it would be necessary to apply some data synchronization methods to align process measurements prior. In this work, our algorithm is developed for the batch processes, where the phase duration is equal, or their difference is minor over batches without special declaration, so that the specific process time can be successfully used as an indicator to data normalization, modeling and online quality predicting. The means of each column are subtracted to approximately 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 to handle different measurement units, thus, giving each equal weight. Then the means and standard deviations are denoted with the specific process time, which will be used in the latter data normalization for online quality prediction.

As mentioned by Cenk Ündey and Ali Cinar,[12] when batch process covers different phases due to operational or phenomenological changes, the data structure should be carefully analyzed to improve the precision of statistical modeling. This can be done by developing local models based on observations from each phase instead of the whole data from the entire duration. Since each phase has its own underlying characteristics, and a batch process can exhibit significantly different effects on product quality over different phases it, is, therefore, natural to develop phase-based statistical modeling methods to reflect the inherent phase nature relevant to quality and improve the performance of quality prediction. Using local models with proper division of phase has advantages. It allows one to unveil the correlation structures that exist in the process data specific to each phase, which might not be possible by using the process data over the entire cycle as the input. This will help us to discriminate local phase-specific effects of process variables on the final quality, where the regression models in critical-to-quality phases will achieve better quality predicting result.

The key to the phase-based modeling strategy is to divide a batch process into several phases by proper clustering of process characteristics. However, generating more accurate models depends on how precisely the phases are identified, so that the data can be separated properly into different segments. Covariance structure changes, reflecting the changes of process characteristics, and their different influences on the quality performance, may be employed in the phase partition algorithm, as pointed out by our earlier work.[8] Loading matrices of each time instance $P_k(J \times J)$, obtained by performing PLS algorithm on normalized data set $\{\widetilde{X}_k(I \times J), \widetilde{Y}(I \times 1)\}$, instead of PCA, incline to reflect the local process covariance information closely related to quality rather than that only between process variables. Then the loading matrices derived from PLS analysis $P_k(J \times J)$, are transformed into weighted forms after considering the importance of each column $P_{k,j}$, which actually represent the process correlation patterns at every time interval

$$\begin{aligned}\breve{P}_k &= \left[P_{k,1} \cdot g_{k,1}, P_{k,2} \cdot g_{k,2}, \ldots, P_{k,J} \cdot g_{k,J}\right] \\ &= P_k \cdot diag(g_{k,1}, g_{k,2}, \ldots, g_{k,J}) \qquad (3)\end{aligned}$$

where $g_{k,j} = \lambda_k^j / \sum_{j=1}^{J} \lambda_k^j$ and λ_k^j is exactly the variance of the associated jth principal component at time k, which indeed measures the variability of process features. Here, it should be noted that λ_k^j differs from the eigenvalue of covariance matrix in PCA.

In the clustering algorithm, the modeling accuracy and complexity depend on the specification of the clustering threshold. A larger threshold results in fewer clusters and more sampling points in each cluster, which contain stable and sufficient process characteristics, but cannot more sensitively unveil the correlations related with quality varying over different patterns. In contrast, a smaller one generates more clusters focusing on reflecting the evolvement of process correlation dynamics from one phase to another. However, less samples in each class cannot provide enough process operation information interrelated with quality in each phase, and induce the lack of regression model robustness. That is, larger threshold conforms to the capability of extracting stable and sufficient phase characteristics from each cluster, and smaller value corresponds to the required ability to track the varying process dynamics between different phases along time. In conclusion, the threshold value should be determined by a tradeoff between the aforementioned two appealing abilities by trial and error. The effect of threshold value on clustering result will be illustrated in the latter simulation section.

Sometimes, in real industrial processes, one can get the phase division information, based on abundant process knowledge and experience when distinct phases are present in a batch process. However, for those complex processes, often the process knowledge is unavailable prior, and data-driven statistical analysis, is, thus, commonly used where phase information can be revealed from the process measurement data. Under this circumstance, clustering is necessary to automatically divide a process into several different time segments with different correlations with quality. Moreover, obtained from the clustering result, the phase division does not exactly have the same meanings as the physical operation phases, since the clustering algorithm is carried out based on the correlations between process measurements and quality variable. In fact, if some prior process knowledge can be obtained, they might assist the clustering result with the better comprehension of phase division. For example, an indicator variable,[12] which provides phase completion information, can help to jointly divide phases instead of only depending on the clustering result.

In this work, the phase-based modeling begins with analyzing and clustering these weighted loading matrices $\widetilde{P}_k$. In this way, those process variations that are more correlated with the quality variables would be used, rather than all the common behaviors in the process variables. Therefore, a process duration may be classified into different phases along time direction by conducting the phase clustering algorithm[8] as shown in the Appendix, revealing different process correlations with the final quality. Consequently, some strategy should be developed to identify the critical phases, which contribute significantly to the quality prediction.

Phase-specific average process trajectory

After phase division, representative two-way data sets $X_c(I \times K_cJ)$ (K_c is the phase duration), are naturally generated for different phases. Although the KJ unfolded variables of the entire process have been reduced quantitatively to K_cJ generalized variable measurements of the cth phase, however, they are still comparatively large and fat vs. the small number of batches I. The developing trajectories of the same process variables at different time within the same phase may cause the regression model complex and confused, thus, increasing the difficulty in data compression. As mentioned before, MPLS is also inefficient in extracting the critical features from such redundant candidate predictors.

According to the aforementioned, each observed process variable should have similar correlations with quality within the same phase. The regression parameters, containing the information of relationships between observed process variables and quality, should correspondingly bear phase-specific nature, ie., show approximately similar regression relationships within the same phase, and significant differences over different phases. Moreover, quality attribute depends on the whole operation performance within the same phase from an overall viewpoint, rather than individual time interval. In fact, process variables of every time slice should have certain percent explanation and contribution to the response variable, showing phase-specific accumulation phenomenon along time, which provides reasonable basis for our proposed approach. Here, the phase-specific average trajectory of process variables $\overline{X}_c(I \times J)$, is utilized as the modeling input instead of the direct unfolded data $X_c(I \times K_cJ)$, which can be steadily obtained by averaging all the scaled measurements slices $\widetilde{X}_k(I \times J)$ belonging to the same phase c

$$\overline{X}_c(I \times J) = \frac{1}{K_c}\sum_{k \in c} \widetilde{X}_k\ (I \times J) \tag{4}$$

where K_c is the number of samplings belonging to the cth phase.

According to the real industrial circumstances, the quality tends to be settled with respect to the average level of process operation. The accidental variation of individual time cannot determine the final quality performance. For example, in a specific batch process, some critical-to-quality variables run at a lower level at the very beginning period of a certain phase, and then ascend to a higher level in time, which make up for the shortage of the variable's function on the quality, and, thus, the entire operation status keeps basically invariable at the phase-specific average level. In this case, the overall effects of the current phase on the quality deliver no significant change, and, thus, the final quality might not depart from its normal reproducibility. The use of average phase-specific process trajectory superiorly reveals the inherent nature of the aforementioned circumstances, thus, avoiding the overfitting problem caused by only focusing on individual time, and enhancing the generalization ability from the overall phase-specific aspect.

Moreover, the use of average process trajectory has advantages over other methods in regression modeling. On the one hand, it allows the conventional PLS model to be applied directly to the simple two-way phase-specific input structure $\overline{X}_c(I \times J)$, rather than the fat unfolded array $X_c(I \times K_cJ)$, which overcomes the impact of data redundancy on the accurate extraction of latent features. On the other hand, the phase-representative regression relationship can be extracted synthetically covering the accumulative effects of process variations on quality along time progressing within the same phase, and revealing the relative importance of each explanatory variable to the explained variable from an "overall" phase-specific perspective. Moreover, based on the average trajectory, it simplifies the identification of critical-to-quality phases, which will be further clarified in the corresponding subsection.

In each phase, we prepare the reference predictor dataset $\overline{X}_c(I \times J)$, and the normalized response dataset $\widetilde{Y}(I \times 1)$ after data preprocessing. Then PLS algorithm performed on $\{\overline{X}_c(I \times J), \widetilde{Y}(I \times 1)\}$ at each phase is formulated

$$\overline{X}_c(I \times J) = T_cP_c^T + E, \qquad \widetilde{Y}(I \times 1) = T_cQ_c^T + F \tag{5}$$

where $W_c(J \times A_c)$, $P_c(J \times A_c)$ are, respectively, weighting matrix and loading matrix for $\overline{X}_c(I \times J)$, $Q_c(1 \times A_c)$, is loading vector for $\widetilde{Y}(I \times 1)$, A_c is the retained number of latent variables. The score vectors T_c can be directly computed from the input matrix $\overline{X}_c(I \times J)$, by the equation

$$T_c(I \times A_c) = \overline{X}_cR_c = \overline{X}_cW_c(P_c^TW_c)^{-1} \tag{6}$$

Then the regression coefficients for the PLS model can be summarized as

$$B_c(J \times 1) = R_cQ_c^T \tag{7}$$

The final phase-based PLS regression model for quality prediction can be deduced as

$$\hat{Y}_c(I \times 1) = \overline{X}_c \cdot B_c = \overline{X}_c \cdot R_c \cdot Q_c^T = T_c \cdot Q_c^T \tag{8}$$

where $\hat{\mathrm{Y}}_c$ is the predicted quality value for the cth phase.

In conclusion, the procedure for the proposed phase-based modeling method is outlined as follows:

First, the three-way array $X(I \times J \times K)$ is unfolded into number of time slices $X_k(I \times J)$. Then they are normalized to be zero mean and unit variance, as well as the final quality variable. Consequently, PLS algorithm can be performed on these normalized time-lice dataset $\{\widetilde{X}_k(I \times J), \widetilde{Y}(I \times 1)\}$ to extract loading matrices P_k, which inclines to reflect the phase-specific process characteristics more relevant to quality variation rather than the simplex process behaviors.

Second, weighted loading matrices $\widetilde{P}_k$, are fed to the phase clustering algorithm, so as to partition the process duration into different phases. Then the data within the same phase are collected together, resulting in a three-way data structure $X_c(I \times J \times K_c)$ (K_c is the time duration of the cth phase).

Third, phase-specific average trajectory $\overline{X}_c(I \times J)$ can be easily obtained by Eq. 4 as the reference predictor variables. Then PLS algorithm is performed on the predigested dataset $\left(\overline{X}_c(I \times J), \widetilde{Y}(I \times 1)\right)$, to extract the phase-representative regression parameter matrices $B_c(J_c \times 1)$, $R_c(J \times A_c)$, and $Q_c(1 \times A_c)$ for the cth phase. Consequently, the realtime score matrix T_k at each time can be correspondingly and steadily obtained by projecting the scaled time slice $\widetilde{X}_k(I \times J)$ $(k = 1, 2, \ldots K_c)$ onto $R_c(J \times A_c)$, which will be used to estimate the missing future measurements when online quality predicting.

The phase-based PLS modeling scheme is shown in Figure 1.

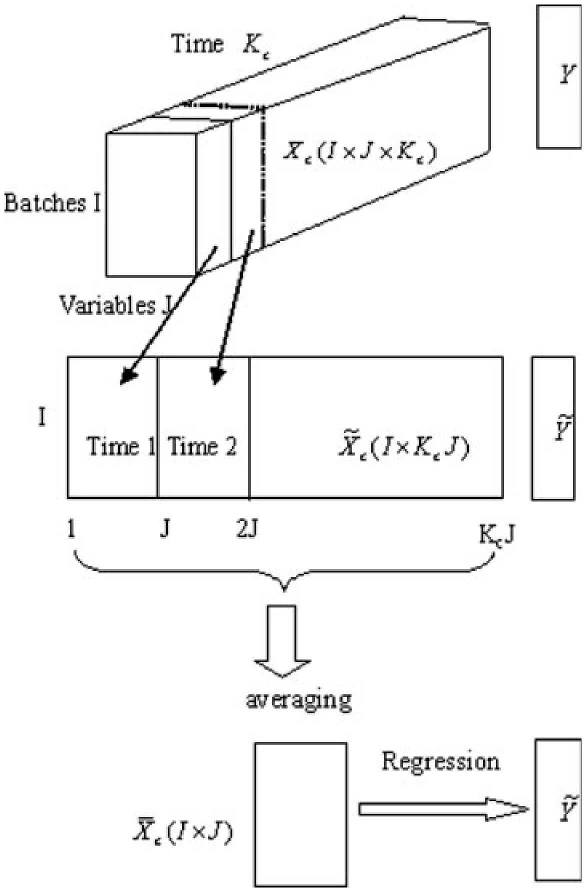

Figure 1. Phase-based PLS modeling scheme for the *c*th phase.

Critical phases checking and process analysis

In conventional MPLS algorithm, the entire batch trajectory is considered simultaneously in the regression model, covering both key and insignificant information to quality prediction. Thus, the performance of feature extraction will be inevitably compromised by the redundant measurements and further, quality prediction performance will be impacted worse. If we can judge the key critical-to-quality phases prior to modeling, the regression models can be greatly simplified, and, thus, the spoiling influence of unimportant factors can be removed. Therefore, it is reasonable to partition the process into different phases, and identify those critical ones based on some statistical index. Then quality prediction can be performed by placing emphasis on the process analysis in critical phases. Noncritical phases might not directly or significantly influence the quality prediction, whose effects on quality, however, can be reflected in the regression models of critical phases due to the correlations between different phases. Therefore, every phase-specific regression model, which represents the prediction relationship in the current phase, actually has impliedly included the effects of those previous phases because of the interactions over phases.

To identify the critical phases, the predictive abilities of the models should be compared. Generally, the quality predictive abilities of regression models in critical phases should be higher than those in other phases. Thus, the nature of checking critical-to-quality phases is to find regression models that provide superior quality prediction performance. Several measures of a model's ability to fit the relationships between process measurements and product quality have been introduced,[13,14] such as R^2, *MSE*, *PRESS,* and $AdjR^2$, which provide an estimation of the average deviation of the model from the real data. Here, the sample squared multiple correlation, namely, the coefficient of multiple determination, is a natural candidate for deciding which model is best fitted, so we will discuss the criterion first

$$R^2 = 1 - \frac{SSR}{SSY} = 1 - \frac{\sum_{i=1}^{I}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{I}(y_i - \bar{y}_i)^2} \tag{9}$$

where $\hat{y}$ is the predicted quality at the ending of each phase. *SSR* is the sum of squares of the prediction residuals, and *SSY* is the sum of squares of the explained variables corrected for the mean. Corresponding to each phase, there is a value to measure the proportionate reduction of total variation Y in association with the use of the predictor set. Generally, R^2 ranges between 0 and 1. The larger R^2, the better fitness of the phase-specific PLS model. Normally, the regression model in critical-to-quality phase is more accurate and reliable for the prediction of quality variable, so it should have larger R^2. A value of R^2 approaching 1 denotes that real quality measurement falls well on the fitted regression surface. Thus, a phase-specific model with a higher R^2 implies a critical-to-quality phase in which the regression model can fit the relationship between predictors and response variable better. F-test[15] can be used to qualify the critical value of R^2 with significance factor α ($\alpha = 0.01$ or 0.05)

$$F = \frac{I - J - 1}{J}\left(\frac{R^2}{1 - R^2}\right) \tag{10}$$

The critical values of R^2 can be conversely calculated by the previous equation, where the critical values of F-statistic with significance factor $\alpha = 0.01$ or 0.05, can be found in the statistical table of F-distribution. If R^2 of the cth phase is larger than the critical value, the phase c is defined as the critical phase to the concerned quality. By Eqs. 9 and 10, we can check the distinct explanation abilities to quality over different phases, and find those critical phases with pivotal effects on quality prediction. The information is extremely helpful for us to further understand the different phase-specific effects of process variables on the final quality, and establish the corresponding modeling and predicting guideline.

As we shall see, using only the criterion R^2, one may not find the best model in a general sense, since it is computed only focusing on the calibration batches to estimate the model's goodness of fit. The similar statistical criterion Q^2, for the predicted values from the test set, is defined the same as in Eq. 9 to more appraise the generalized predictive performance of each phase-specific model. We may say that our inferential model is reliable if it works well for a new sample. Usually R^2, of a calibration set is larger than of a validation set, because calibration models can easily lead to overfitting of the data. Here it should be noted that Q^2 more focuses on the generalized predictive ability of the models rather than just the fitting competency. Comparatively, the previous two metrics, R^2 and Q^2, respectively, make full use of train and test data subsets, so that their effective combination can well quantify the model generalization capability and prediction performance. In this way, it overcomes the overfitting defect when only devoting one's attention to the prediction errors of train data set. Therefore, the critical phases should also be determined based on both of the two metrics. If and only if both R^2 and Q^2 are above the critical limits derived from F-test in linear regression, the corresponding phase is deemed to be critical to quality.

Commonly, when the number of observations is sufficient, the samples are split randomly into two parts: one used to calibrate the model (i.e., estimate the parameter values), and the other to validate the model (i.e., to test the predictive performance). In general, it is often recommended to use more data to calibrate than to validate; usually two-thirds to one-third split of the sample. The latter set is sometimes known as a holdout sample. After fitting the model using the data from the calibration subset, we can get the regression parameters, and then use them to calculate the predicted values for the observations in the holdout sample. Thus, we can more assess the generalized predictive accuracy of the model in the validation sample rather than the fitting ability in the calibration sample. Moreover, if the data set is too small to split the sample, an alternative is to use jackknife validation.

In addition, different process variables in critical phases may have different effects on the quality variation. The phase-specific regression coefficient $B_c(J \times 1)$, explores the different explanation abilities of different process variables to quality from an overall phase viewpoint. Generally, a larger absolute value of regression parameter denotes that the process variable is dominant in quality prediction. Meanwhile, a negative value indicates that the predictor changes in the opposite trend to response variable. Through detailed analysis of regression coefficients, one can clearly get the relationships between the explanatory variables and the explained variable from the prediction aspect.

Estimating missing measurements

Because the average of integrated phase-specific observations is required for prediction, it is inevitable to estimate the missing process observations for online quality predicting, since the current phase-specific measurements are not complete until the end of the phase. Nomikos and Macgregor[16] have developed three approaches to anticipate the unknown measurements: (1) fill the missing values with zero deviation; (2) fill the missing values with current deviation, and (3) fill the missing values using PCA projection. However, since the dynamic process evolvement is not taken into account, it is very difficult for the predicted trajectory to quickly catch up with the real process operation, potentially yielding undesirable quality predicting results. To improve the data filling performance, Cho and Kim[17] developed a new method to supplement the unknown part of the new batch, which makes extensive use of the history batch trajectories. The current monitored batch is compared with these reference trajectories at the passed time, where the most similar one is chosen to fulfill the unknown future data. In this way, the process dynamic relationships are preserved properly, and, thus, the process information can be clarified preferably. This method entails calculation of the distance similarity between the current evolving process and all history data at each time. However, in their method, the raw process variables are employed to calculate the distance dissimilarity. Since the "similar trajectory selection" method is a distance-based technique, it is easily susceptible to outliers and noise in the data. Moreover, for MPLS, the estimation of future measurements may bear a larger inaccuracy, especially during the initial period, because at that time most measurements are unknown. Therefore, large quantities of estimated data will distort the real process operation information, leading to poor quality predicting performance during the initial period of a batch. In addition, without distinguishing the different phase-specific behaviors, the distance calculation tends to be unable to reflect the similarity between new batch and reference batches reliably enough. The proper division of phases and checking of critical phases enable the complement of missing future data to be implemented, focusing on each critical phase, which can reduce the amount of unavailable data to be anticipated, and, meanwhile, take into account the phase-specific characteristics. Considering that the estimation accuracy of the missing part of the new batch is vital to the performance of quality prediction, it is significant to fulfill the current batch with dynamic trajectory sensitive enough to operation variation. The problem of unsatisfactory quality anticipating results, caused by poor unknown data estimation, can be overcome by tracking the dynamic pattern of the new batch. Despite increased computational load, the accurate estimation of unknown data enhances the reliability of the quality predicting results. Thus, the calculation burden and time consumption is deserved.

In this work, the missing observation supplement procedure is performed based on the distance similarity calculation, whose basic idea roots in the initial algorithm by Cho and Kim,[17] and further developed to be fit for our application situation. In our data estimation procedure, the distance similarity between the current new batch and the past batch trajectories are calculated based on the latent scores rather than the raw process variables. In this way, the outliers and noise consisting in process measurements are filtered, and, thus, the critical process features are emphasized. Moreover, the realtime average critical feature trajectories are employed in the distance calculation as the basis unit rather than raw sampling at each time.

As mentioned in the outline of the modeling procedure, the corresponding score matrix T_k, at each time for history batches have been steadily obtained by projecting time slice $\widetilde{X}_k(I \times J)(k = 1, 2, \ldots K_c)$ onto $R_c(J \times A_c)$, which represents most of the important process information relevant to the quality variation along time direction within each phase. First, for each time slice score matrix $T_k(I \times A_c)$, it is necessary to consider the different importance of each principal component $T_{k,j}(I \times 1)$, and give them different weights

$$\begin{aligned}\widetilde{T}_k &= \left[T_{k,1} \cdot g_{k,1}, T_{k,2} \cdot g_{k,2}, \ldots, T_{k,A_c} \cdot g_{k,A_c},\right] \\ &= T_k \cdot diag(g_{k,1}, g_{k,2}, \ldots, g_{k,A_c}) \end{aligned} \tag{11}$$

where $g_{k,j} = \lambda_k^j / \sum_{j=1}^{A_c} \lambda_k^j$ and λ_k^j is the variance of the associated jth principal component at each time k.

During the operation of each critical phase c, the weighted score row vector $\tilde{t}_k^i(1 \times A_c)(i = 1, 2, \ldots, I)$ covers the critical feature at time k. Then the corresponding average value of weighted scores up to the current time k from the beginning of the critical phase for all history batches $\bar{t}_k^i$, are calculated and stored in the history library

$$\bar{t}_k^i(1 \times A_c) = \frac{1}{k - K_{cs} + 1} \sum_{k \in c} \tilde{t}_k^i(1 \times A_c) \tag{12}$$

where superscript i denotes batch, and subscript k and c denote time and phase, respectively. K_{cs} is the starting time of phase c. Consequently, the real-time synthetical process variation information closely related with the final quality is represented naturally by $\bar{t}_k^i$, corresponding to each batch. So the history data library is built as the candidate trajectories to fill the incomplete new batch.

For the coming new batch, up to the current time k, process time is employed to check the location of new measurement x_k^{new}, so as to normalize the measurement adopting the corresponding mean and standard variance obtained from the modeling procedure. Moreover, process time can be used to identify phase completion, so as to check which phase-specific should be utilized to calculate the real-time score t_k^{new}. Then the weighted score $\tilde{t}_k^{\text{new}}$, can be readily obtained using Eq. 11. From Eq. 8, it can be revealed that for the new batch, the quality prediction performance depends directly on the credibility of average phase-specific score. Considering that the online quality prediction is conducted, based on regression models established, and based on phase-average process trajectory, therefore, the objective of data estimation is to anticipate the future observations in such a way that the obtained phase-specific average score at time k, $\bar{t}_{k,c}^{\text{new}}$, should be as close as possible to those that would have been computed, based on the real complete phase measurements. The real-time average form $\bar{t}_k^{\text{new}}$, can be calculated using Eq. 12 to reveal the critical feature derived from the new process trajectory up to the current time k. Therefore, the dissimilarity distance between history batches, and the current new batch is compared using $\bar{t}_k^i$ and $\bar{t}_k^{\text{new}}$. In this way, the use of average level of scores can overcome the influence of individual sampling variation. The Euclidean distance, the most popular metric, can be used to calculate the dissimilarity level between two features

$$d_k^i = (\bar{t}_k^{\text{new}} - \bar{t}_k^i) \cdot (\bar{t}_k^{\text{new}} - \bar{t}_k^i)^T \tag{13}$$

where larger d_k^i means larger dissimilarity between two critical feature $\bar{t}_k^i$ and $\bar{t}_k^{\text{new}}$. The least d_k^i indicates that history batch i has the most similar process feature to the current batch i, which will be employed to complete the missing data. Then by averaging all the scores including the known and estimated ones within phase c, the phase-specific average score $\bar{t}_{k,c}^{\text{new}}$ can be obtained naturally corresponding to each time k within the cth phase.

The basic procedure of data estimation method focusing on the cth phase can be summarized as follows:

Step 1: For the new batch, normalize the new measurement x_k^{new}, the same as the way in the modeling procedure, and calculate the weighted scores at each time k, $\tilde{t}_k^{\text{new}}$.

Step 2: Calculate the average score up to the current time k, $\bar{t}_k^{\text{new}}$, based on Eq. 12.

Step 3: Calculate the distance dissimilarities d_k^i, between the new batch and those in history score library using $\bar{t}_k^{\text{new}}$ and $\bar{t}_k^i$, based on Eq. 13.

Step 4: Find the most similar batch m to the current new batch with the least d_k^i, and complete the new batch by employing those scores of history batch m from k to the end of phase c.

Step 5: Corresponding to time interval k, the phase-specific average score $\bar{t}_{k,c}^{\text{new}}$, can be obtained by averaging all the score vectors within the cth phase, including real and estimated scores, which will be used for online quality predicting.

The procedure will be repeated at every current time k, and the most similar history trajectory is reselected continuously until reaching the terminal end of phase c. On the one hand, the use of real-time average scores can overcome the overdependence of estimation accuracy on every individual sampling, which tends to drift with the influence of individual process variation. Also it need not bear the heavy calculation burden and complexity, but can still ensure the reliable data filling precision. On the other hand, it allows estimation of unknown future observations located in each individual phase. The data outside the current phase will not impact the accuracy of data estimation. In this way, it weakens the influence of inaccurate data estimation caused by too many assumptions of incomplete measurements required by MPLS-based methods, where the measurements from the current up to the end of the entire batch cycle are unavailable.

In addition, there are several issues that should be investigated further. The missing measurement estimation performance would be directly affected by the performance and sufficiency of the history batch library. So, it would be necessary to gather as broad and abundant normal batch trajectories as possible from past batch runs. Moreover, updating batch library with the evolvement of operation is also an important issue, where new normal batches are added into the database continually, supplementing the backup trajectory information. In addition, because quality prediction is conducted in real-time, the computational time of estimating unknown measurements should also be taken into account. Considering the accurate estimation of missing data can greatly improve the desired accuracy of quality prediction, it deserves such computation devotion. In conclusion, one must reconcile the improvement of quality predicting performance vs. increased calculation complexity.

Online quality predicting

After missing data complement, the phase-specific average score of the new batch obtained at each time k, $\bar{t}_{k,c}^{\text{new}}$, can be directly employed to conduct the online quality predicting combined with the corresponding phase-representative model using Eq. 8. The predicted quality may vary with time within the same phase. The variation may be caused by measurement noises and data estimation errors, especially in the initial period as the phase-specific average trajectory at that time is obtained from few real observations and vast estimated data. With the development of process operation within the phase, more and more real observations will substitute the estimated unknown data, which makes the quality prediction more reliable and accurate. So, the final quality predicted at the ending of each specific phase $\hat{y}^c$, is defined as the end-of-phase prediction.

As mentioned by Lu and Gao,[7] quality variables in a batch run can be divided into two types: quality determined by only one specific phase, and quality determined by more than one phase. It is comprehended easily that only critical-phase PLS models can give reliable and stable quality prediction, while others are not directly related with the quality index, so there is no need to waste extra efforts to forecast the product in those unimportant time regions. Moreover, since different critical phases explain different parts of quality variations, a strategy will have to be implemented to stack cumulative effects of multiple phases on quality.

Without losing generality, assuming that quality variable has two critical phases, phase 1 and phase 2, the current online quality prediction can then be yielded as

$$\hat{y}_k = \begin{cases} \bar{t}_{k,c}^{\text{new}}(1 \times A_c) \cdot Q_c(A_c \times 1) & k \in \text{the two critical phases; } c = 1,2 \\ \text{null} & \text{others} \end{cases} \quad (14)$$

Weighted sum of end-of-phase predicted values is formulated as the final forecast quality

$$\hat{y} = w_1 \cdot \hat{y}^1 + w_2 \cdot \hat{y}^2 \quad (15)$$

where $\hat{y}^1$ and $\hat{y}^2$ are the end-of-phase quality prediction corresponding to phase 1 and 2, respectively; w_1 and w_2 are, respectively, weights of phase 1 and 2. Zhang et al.[18–21] have pointed out that stacking weights can be determined in a number of ways in their series of investigations about combining individual neural networks. A simple, but inappropriate approach is to give equal weights to each individual model. A second way is to obtain different weights through multiple linear regression. Here, since R^2, as the fitness of referential models, reveals the different significance to quality prediction, we can directly employ them to calculate the phase-specific weights in order to stack the cumulative, but different effects of each critical phase on the product quality: $w_c = R_c^2/(R_1^2 + R_2^2)$.

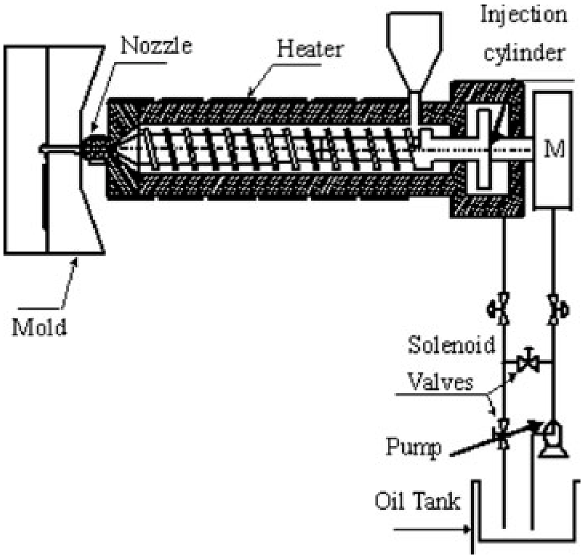

Figure 2. Injection molding machine.

Illustration and Discussion

Injection molding process description

Injection molding,[22,23] a key process in polymer processing, transforms polymer materials into various shapes and types of products. Figure 2 shows a simplified diagram of a typical reciprocating-screw injection molding machine with instrumentation.[22] A typical injection molding process consists of three 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.[22] It is a typical process for the application and verification of the proposed phase-based quality prediction algorithm, where process variables can be collected online with a set of sensors, while the quality variable is only available at the end of each batch run.

The material used in this work is high-density polyethylene (HDPE). The process variables and quality variable selected for modeling are listed in Table 1. In the simulation illustration, we focus on the prediction of product length, that is, the dimension quality, whose real measurements can be directly obtained offline by instruments. The operating

Table 1. Process and Quality Variables for Injection Molding Process

No.	Variable's descriptions	Unit
	Process variables	
1	Cavity Temperature (C. T.)	°C
2	Nozzle Pressure (N. P.)	Bar
3	Stroke	mm
4	Injection Velocity (I. V.)	mm/sec
5	Hydraulic Pressure (H. P.)	Bar
6	Plastication Pressure (P. P.)	Bar
7	Cavity Pressure (C. P.)	Bar
8	Screw Rotation Speed (S. R. S)	RPM
9	SV1 opening(SV1)	%
10	SV2 opening (SV2)	%
11	Barrel Temperature (B. T.)	°C
12	Mold Temperature (M. T.)	°C
	Quality variable	
	Length	mm

conditions are set as shown in Table 2. 33 normal batch runs are conducted under various operation conditions by DOE (design of experiment) method. Because of different filling time in the injection phase induced by different injection velocities, reference batch runs, therefore, have varying operation length. Using injection stroke as an indicator variable, we can fix the injection phase with unified duration by data interpolation. Moreover, control packing-holding and cooling time at 6 and 15 s, respectively, we can get the final data matrices $\overline{X}(33 \times 12 \times 1300)$ and $Y(33 \times 1)$, among which the first 25 batches are used for modeling, post-batch process analysis and knowledge extraction, while the other eight cycles are used for model validation.

Phase-based process analysis

The weighted loading matrices calculated from PLS on the time slice data sets are fed to the clustering algorithm. Moreover, to reveal the influence of threshold value on clustering result, phase division is performed adopting two different thresholds, 0.03 and 0.01. The results are contrastively shown in Table 3, where the whole process trajectory is automatically and generally divided into four main phases. Corresponding to the two thresholds, it is generally similar for the first two main phases (phase 1 and phase 2), and the last main phase (phase 4); but with the smaller threshold, phase 3 obtained from the larger threshold is subdivided into two subphases, phase 3-1 and 3-2. Combined with the indicator variable technique and prior process experience, it is not difficult to find out that the two subphases indeed correspond to plastication phase. To evaluate the performance of the two kinds of phase partition results, they are both put into modeling and online quality prediction. Table 3 also lists the analysis result of model fitness over different phases. The larger R^2, the better fitness of the corresponding phase-specific PLS model. Moreover, to further affirm the reasonability of the earlier analyses, multiple coefficient of determination Q^2 for test batches are also evaluated comparatively, which are generally smaller than R^2. From the fitness analysis in Table 3, it reveals that the subdivision of plastication phase cannot improve the performance of quality prediction. Meanwhile, for the other three main phases, it yields similar prediction performance based on the fitness evaluation. Therefore, the phase division with 0.03 as threshold is chosen finally, and the following process analysis is performed on it to illustrate the quality prediction algorithm, based on phase-specific average trajectory.

Based on the selected clustering result, it clearly shows that without using any prior process knowledge, the trajectory of the injection molding can be automatically divided into seven phases, among which four long phases agree well with four physical operation phases of the process, that is, injection, packing-holding, plastication and cooling phases, plus a few short transitional time periods. These temporary time regions, corresponding to the dynamic transition period between main phases with unstable process states, form individual time regions, which have little impact on quality prediction. Dividing in detail a batch process into "steady" and transient phases cannot only improve quality prediction performance, but enhance process analysis and understanding. Without losing generality, each phase hints different effects on the final product, as well as different correlations between process variables and product quality. According to the quantitative evaluation of fitness metrics shown in Table 3, it indicates that phase 2 and 3, especially phase 2, the values are above the critical point derived from Eq. 10, which can be deemed to be critical-to-quality phases. In fact, no matter which of the two threshold values 0.03 or 0.01, is selected, the critical-to-quality phases are identified the same. It demonstrates from another aspect that it is reasonable to conduct quality prediction located in different phases. Normally, critical-to-quality phase models are more accurate and reliable for the prediction of quality variable. Therefore, in critical-to-quality phases one can expect more accurate quality prediction results.

In critical-to-quality phase 2 and 3 (i.e., the packing-holding and plastication phases), the phase regression coefficients of process variables are plotted in Figure 3a and b, respectively, which can numerically explain how these variables will affect the product quality. Commonly, pressure variables have positive relation with the quality, while temperature variables are negatively correlated with the quality, which generally tells that higher pressures and lower temperatures result in larger product length. Also longer screw displacement results in more material crammed into the cavity, which obviously generates a longer product. For example, for phase 2, the packing-holding phase, variables no 2, 3, 5, 7 and 9, including pressure variables (nozzle pressure, hydraulic pressure, and cavity pressure), and displacement variable (stroke),

Table 2. Operation Condition Setting for Injection Molding Process

Operating conditions	Setting values
Material	High-density polyethylene(HDPE)
Injection velocity	22~26 mm/sec
Packing pressure	150, 300, 450 bar
Barrel temperature	180, 200, 220°C
Mold temperature	15, 35, 55°C
Packing-holding time	6 sec
Cooling time	15 sec

Table 3. Phase Partition Analysis with Two Different Parameters

	Threshold = 0.01				Threshold = 0.03		
	Duration	R^2	Q^2	Critical value (95%)	Duration	R^2	Q^2
	1~41	–	–	–	1~41	–	–
I	42~244	0.4741	0.4011	0.7288	42~233	0.5743	0.4103
	245~255	–	–	–	234~246	–	–
II	256~557	0.9453	0.7423	0.7288	247~555	0.9314	0.7421
	558~576	–	–	–	556~582	–	–
III-1	577~781	0.8988	0.7123	0.7288	583~908	0.9092	0.7312
III-2	782~926	0.83726	0.7002				
IV	927~1300	0.6481	0.5304	0.7288	909~1300	0.6301	0.5116

which have higher positive regression coefficients, indicating that they are quality-correlated variables. Moreover, they are all positive related with the quality. Variables no 1, 11, 12, i.e., temperature variables are negatively related with the quality. This explores the different relations of process variables with quality prediction, implying that larger pressures and lower temperatures may result in larger product length. That is, in the packing-holding phase, impacted by the corresponding pressures, the melted materials in the cavity are further cooled, pressed, and supplemented to be more compact, which may result in a longer product. Without the use of prior process knowledge, the aforementioned phase-based analyses agree well with the real physical process, which can be useful for the improvement of quality prediction.

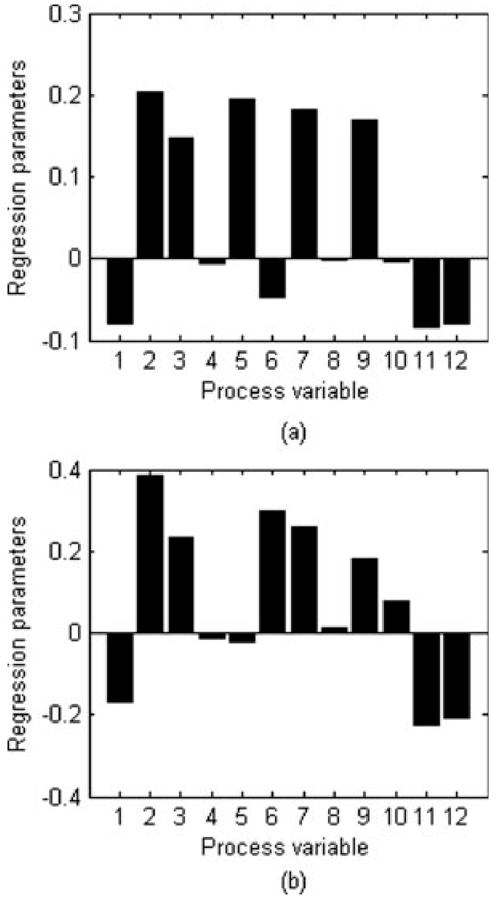

Figure 3. Regression parameters of process variables for (a) phase 2, and (b) phase 3.

Performance illustration of quality prediction

According to the aforementioned analysis and understanding, phase 2 and 3 are indicated as critical phases. The product length has close relations with both phases. Based on the aforementioned unknown measurement filling method in this work, the real-time phase-specific score average trajectories in phase 3 for one test batch are illustrated in Figure 4.

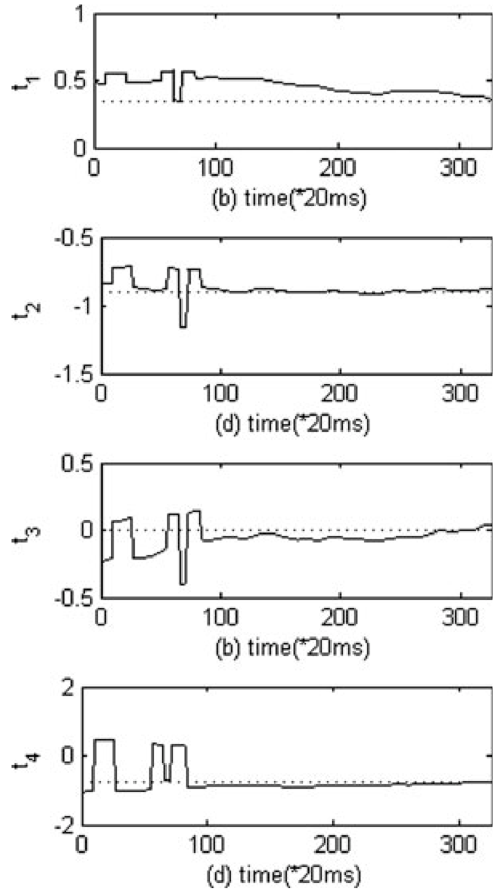

Figure 4. Online estimation of average score trajectory in phase 3 for test batch 4 (dash line, real phase-based average trajectory; sold line, online estimation values).

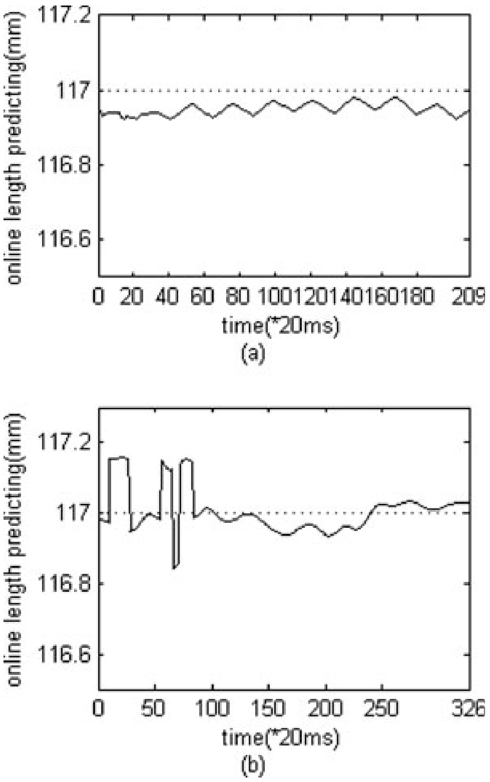

Figure 5. Online quality prediction for one test batch in critical phases (dash line, real measurement of final quality; sold line, online prediction).

Taking the first four scores as example, it can be seen that the estimated average scores can better track the real phase-specific average trajectories dynamically with a satisfying trend, especially when approaching the phase terminal. It results from the fact that with time evolving, the known process measurements become more and more, providing more abundant key process information denoted with scores, and, thus, the estimation precision will be more accurate. Then, the online quality prediction is performed at each sampling time of critical phase 2 and 3, respectively, as shown in Figures 5a and b for the test batch. The maximum online predicted error rates are less than 0.07% corresponding to phase 2 and 0.15% in phase 3, respectively, which are a well acceptable prediction precision in industry. The accurate quality prediction can successfully demonstrate the phase-specific effects of process variables on quality, explore process operating information, and evaluate product quality performance in advance. Here it should be noted that since it is inevitable to infer the unknown future process measurements, therefore, the accurate estimation of missing data is important for reliable quality prediction. However, due to the partition of different phases along process operation, whenever the process enters a new phase, the amount of available known measurements becomes little in the current phase, thus, providing less useful information for the accurate supplement of missing data. Therefore, it may be reluctant to get a credible quality prediction at the beginning of a new phase, which consequently requires caution during the period when distinct phases are transiting between each other. With process development, the real observations become more apparent for the current phase, and it will be easier and more reliable to catch up with the real trajectory of the new phase, and, thus, get a more accurate quality prediction.

The final predicted quality is obtained by taking into account the cumulative effects over the two phases. Figures 6 and 7 deliver a comparison of offline prediction for training and testing batches, respectively, using the proposed method, stage-based sub-PLS method by Lu[7] and conventional MPLS method. From Figure 6, it is general that conventional MPLS model has the superiority for fitting training batches, where the predicted quality fitted well with the real quality measurement. In contrast, from Figure 7, it can be seen that for test batches, especially batches 1, 3 and 7, the trained MPLS regression model fails to give a reliable enough quality prediction result. The comparison may illustrate the disadvantages of conventional MPLS algorithm. Although MPLS is well-known as an effective data compression and feature extraction technique, one cannot expect it to

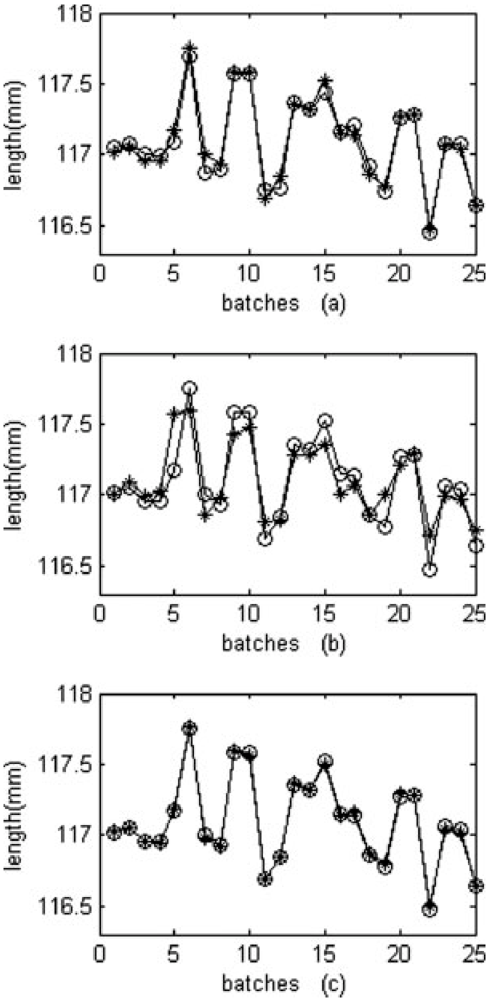

Figure 6. Offline quality prediction results for the reference batches using (a) proposed method; (b) Lu's stage-based sub PLS method, and (c) MPLS method.

(-○-, real measurements; -∗-, predicted measurements).

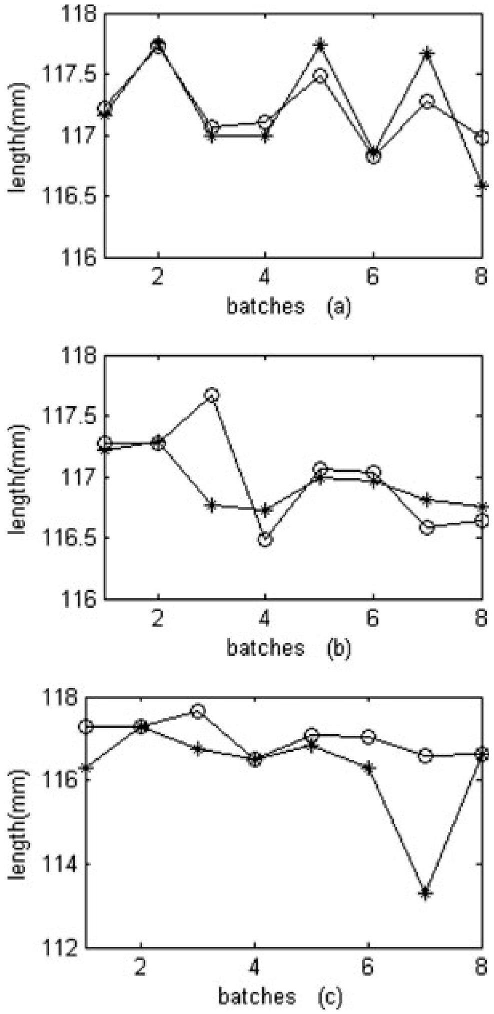

Figure 7. Offline quality prediction results for the test batches using (a) proposed method; (b) Lu's stage-based sub PLS method, and (c) MPLS method.

(-○-, real measurements; -∗-, predicted values).

pick up pivotal information from a great capacity of redundant candidate data information. Thus, conventional MPLS may not clearly distinguish the system information from normal stochastic noises, which have little relation with the final quality. Therefore, the over-fitting to common-caused process variations may spoil the models' ability to capture those useful predictor information, and, thus, loses the favorable adaptability to new batches with normal batch-wise stochastic dynamics. For batch 1, 3 and 7, resulting from different normal stochastic changes in operation trajectory from those training batches, which is often the general case, they go beyond the competence of trained regression model. Lu's sub-PLS method also compromises the reliability of predicted quality since their phase-based regression relationships are extracted, only focusing on each individual time-slice, which may not be able to catch stable enough prediction information. By comparison, the superiority of the proposed method over the other two methods is obvious for both training and testing batches, which gives a satisfying overall prediction trend, demonstrating the models' fitness ability and prediction adaptability.

Based on the earlier illustration and discussion, conclusion can be drawn that the proposed method can give reasonable predictions for multiphase batch processes where there are time-cumulative effects of process variables on quality within each phase. Moreover, some factors deserve special consideration when applying the proposed method to multiphase processes for online quality prediction. To separate the process into different phases, the process correlations more relevant to quality are analyzed and employed in the clustering algorithm, where similar process patterns are grouped into the same phase and process correlations change over different phases. Thus, phases can be recognized for their different underlying process characteristics rather than physical operations. That is, one physical operating stage can be also divided into several different subphases if inherent process characteristics change within the stage. Thus, the clustering algorithm can well apply to multiphase batch processes, no matter whether phases result from physical operation or phenomenological changes, which both reveal phase-specific underlying process behaviors. Therefore, although the illustration case in this work is a typical process consisted of distinct physical phases, and, thus, obvious phase separation, it is readily to deduce that the clustering algorithm has the ability to capture these phenomena driven phases, and can well work in processes where only phenomenological changes govern the operation. It would be more convincing to demonstrate this algorithm using more illustrative examples, which is, however, beyond this work due to the lack of space. The extensive illustrations under various real circumstances are significant and worthy of further investigation in the future.

Conclusions

A quality prediction method has been proposed for the improvement of quality inferential performance in multiphase batch processes. Based on the clustering algorithm, it allows us to automatically divide a batch process into different phases, check the critical ones, and develop phase-based PLS regression models with a simpler structure using phase-specific average process trajectory. Therefore, we can conduct quality prediction located in those critical phases, which may provide stabler and more reliable phase-specific predicting relationship. The proposed scheme cannot only give a valid quality predicting result earlier, but also help to better understand the phase-specific accumulative effects of process trajectory on quality, and find out the critical factors to the concerned quality. All the aforementioned provide the potential for the improvement of quality prediction. The application to injection molding process illustrates that the proposed method is effective. Especially, if the proposed method is applied to multiphase batch processes bearing obvious time-cumulative effects on quality with process evolving, its superiority will appear more clearly.

Acknowledgments

The authors would like to acknowledge Professor Furong Gao's group at Hong Kong University of Science and Technology (HKUST) for providing the injection molding data for testing. The project was supported in part by the National Natural Science Foundation of China (No. 60374003 and No. 60774068) and Project 973 (No. 2002CB312200), China.

Literature Cited

1. Jackson JE. *A User's Guide to Principal Components.* Wiley: New York: John Wiley & Sons, Inc.; 1991.
2. Geladi P, Kowalshi B. Partial least squares regression: A tutorial. *Anal Chim Acta.* 1986;185:1–17.
3. Nomikos P, MacGregor JF. Monitoring batch processes using multi-way principal component analysis. *AICHE J.* 1994;40:1361–1375.
4. Nomikos P, MacGregor JF. Multi-way partial least squares in monitoring batch processes. *Chemom Intell Lab Syst.* 1995;30:97–108.
5. Duchesne C, MacGregor CD. Multivariate analysis and optimization of process variable trajectories for batch processes. *Chemom Intell Lab Syst.* 2000;51:125–137.
6. Chu YH, Lee YH, Han C. Improved quality estimation and knowledge extraction in a batch process by bootstrapping-based generalized variable selection. *Ind Eng Chem Res.* 2004;43:2680–2690.
7. Lu N, Gao F. Stage-based process analysis and quality prediction for batch processes. *Ind Eng Chem Res.* 2005;44:3547–3555.
8. Lu N, Gao F. Stage-based online quality control for batch processes. *Ind Eng Chem Res.* 2006;45:2272–2280.
9. Lu N, Gao F, Wang F. A sub-PCA modeling and on-line monitoring strategy for batch processes. *AIChE J.* 2004;50:255–259.
10. Zhao SJ, Zhang J, Xu YM. Performance monitoring of processes with multiple operating modes through multiple PLS models. *J of Process Control.* 2006;16:763–772.
11. Dayal BS, MacGregor JF. Improved PLS algorithms. *J Chemometrics.* 1997;11:73–85.
12. Ündey C, Cinar A. Statistical monitoring of multistage, multiphase batch processes. *IEEE Contr Syst Mag.* 2002;22:40–52.
13. David G.K, Lawrence LK, Keith EM, Azhar N. *Applied Regression Analysis and Other Multivariable Methods.* 3rd ed. Beijing: China Machine Press; 2003.
14. Michael HK, Christopher JN, John N. *Applied Linear Regression Models.* 4th ed. Beijing: Higher Education Press; 2005.
15. Wang H. Partial Least-Squares Regression-Method and Applications. Beijing: National Defence Industry Press; 1999.
16. Nomikos P, MacGregor JF. Multivariate SPC charts for monitoring processes. *Technometrics.* 1995;37:41–59.
17. Cho HW, Kim KJ. A method for predicting future observations in the monitoring of a batch process. *J Qual Technol.* 2003;35:59–69.
18. Zhang J, Martin EB, Morris AJ, Kiparissies C. Inferential estimation of polymer quality using stacked neural networks. *Comput Chem Eng.* 1997;21:1025–1030.
19. Zhang J, Martin EB, Morris AJ, Kiparissies C. Prediction of polymer quality in batch polymerization reactors using robust neural networks. *Chem Eng J.* 1998;69:135–143.
20. Zhang J, Martin EB, Morris AJ, Kiparissies C. Developing robust non-liner models through bootstrap aggregated neural networks. *Neurocomputing.* 1999;25:93–113.
21. Zhang J, Martin EB, Morris AJ, Kiparissies C. Inferential estimation of polymer quality using bootstrap aggregated neural networks. *Neural Networks.* 1999;12:927–938.
22. Yang Y, Gao F. Cycle-to-cycle and within-cycle adaptive control of nozzle pressures during packing-holding for thermoplastic injection molding. *Polym Eng Sci.* 1999;39:2042–2064.
23. Yang Y, Gao F. Adaptive control of the filling velocity of thermoplastics injection molding. *Control Eng Practice.* 2000;8:1285–1296.

Appendix

Modified k-means clustering algorithm

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

Outputs: the number of clusters C, the cluster centers $\{W_1, W_2, \ldots, W_C\}$, and the strict membership of $\breve{P}_k$, 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 $W_c^0(\mathrm{c} = 1, 2, \ldots, C^0)$, 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 $\breve{P}_k$ to all of the centers $dist(\breve{P}_k, W_c^{i-1})$, assign to the nearest center W_{c*}^{i-1}, and denote its membership 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, recompute the new cluster centers $W_c{}^i(c = 1, 2, \ldots, C^{\mathrm{i}})$, 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.

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

Manuscript received May 19, 2007, revision received Sept. 19, 2007, and final revision received Nov. 20, 2007.