Front. Electr. Electron. Eng. China (2006) 2: 211–215
DOI 10.1007/s 11460-006-0010-7

RESEARCH ARTICLE

LI Chun-fu, ZHANG Jie, WANG Gui-zeng

Adaptive quality prediction of batch processes based on PLS model

Abstract There are usually no on-line product quality measurements in batch and semi-batch processes, which make the process control task very difficult. In this paper, a model for predicting the end-product quality from the available on-line process variables at the early stage of a batch is developed using partial least squares (PLS) method. Furthermore, some available mid-course quality measurements are used to rectify the final prediction results. To deal with the problem that the process may change with time, recursive PLS (RPLS) algorithm is used to update the model based on the new batch data and the old model parameters after each batch. An application to a simulated batch MMA polymerization process demonstrates the effectiveness of the proposed method.

Keywords Batch processes, PLS, RPLS, Modeling

1 Introduction

Batch and semi-batch processes have been increasingly used in the production of low volume and high value-added products. Usually there are no on-line product quality measurements and only on completion of the batch, a range of quality measurements is made at the quality-control laboratory on a sample of the product. In order to make the product quality better controlled, it is important to develop models to predict the end-product quality from the readily available on-line measurements, especially at the early stage of the batch. Another problem is that batch processes generally exhibit some batch-to-batch variations. Inevitably, there exist some mismatches between the already developed model and the new batch run. So it is desirable to update the model based on the newly obtained information in order to adapt to the process variations.

In this paper, a model for predicting the end-of-batch product quality from on-line measurements of process variables at the early stage of a batch is developed using the PLS method. When some quality measurements are available at one or more time instants during a batch, they are used to rectify the final prediction results. After each batch, the PLS model is updated based on the new batch data and old model parameters to adapt to the process variations.

The remaining sections of the paper are organized as follows. Sect. 2 presents the PLS modelling methodology of batch processes. The model update is given in Sect. 3. An application to a simulated batch MMA polymerization reactor is given in Sect. 4. Sect. 5 concludes this paper.

Translated from *Jorunal of Tsinghua Science and Technology(Natural Science Edition),* 2004, 44(10): 1360-1363 (in Chinese)

LI Chun-fu(✉),WANG Gui-zeng
Department of Automation, Tsinghua University, Beijing 100084, China
E-mail:lichunfu99@ tsinghua.org.cn
ZHANG Jie
School of Chemical Engineering and Advanced Materials, University of Newcastle, Newcastle upon Tyne NE1 7RU, U.K.

2 Modelling of batch processes

2.1 PLS model

The prediction of end-product quality is carried out using a PLS model. PLS regression is employed because the measured process variables are usually highly correlated in batch processes. PLS overcomes this problem by first projecting the input and output data down onto a subspace of orthogonal latent variables and then regressing each pair of corresponding input and output score vectors by univariate regression. Suppose $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the input and output matrices respectively. The PLS regression builds a linear model by decomposing $\boldsymbol{X}$ and $\boldsymbol{Y}$ into bilinear terms as follows:

$$\boldsymbol{X} = \boldsymbol{t}_1 \boldsymbol{p}_1^{\mathrm{T}} + \boldsymbol{E}_1 \quad (1)$$

$$\boldsymbol{Y} = \boldsymbol{u}_1 \boldsymbol{q}_1^{\mathrm{T}} + \boldsymbol{F}_1 \quad (2)$$

where $\boldsymbol{t}_1$ and $\boldsymbol{u}_1$ are score vectors of the first PLS factor, $\boldsymbol{p}_1$ and $\boldsymbol{q}_1$ are the corresponding loading vectors, and $\boldsymbol{E}_1$ and $\boldsymbol{F}_1$ are the residual matrices. The scores vectors $\boldsymbol{t}_1$ and $\boldsymbol{u}_1$ are such determined that they have the largest covariance

among all the following $\boldsymbol{t}$ and $\boldsymbol{u}$ vectors: $\boldsymbol{t}=\boldsymbol{Xw}$, $\boldsymbol{u}=\boldsymbol{Yc}$, $\|\boldsymbol{w}\|=\|\boldsymbol{c}\|=1$, where $\boldsymbol{w}$ and $\boldsymbol{c}$ are the weight vectors for the input and output matrices, respectively. The score vectors can be calculated by using the nonlinear iterative partial least squares (NIPALS) algorithm [1–2] or singular value decomposition (SVD) [3–4].

Equations (1)-(2) formulate a PLS outer model. After the outer calculation, the score vectors are related by a univariate linear regression model as follows:

$$\boldsymbol{u}_1 = b_1\boldsymbol{t}_1 + \boldsymbol{\varepsilon}_1 \tag{3}$$

where b_1 is the regression coefficient and $\boldsymbol{\varepsilon}_1$ the residual. b_1 can be solved by the least squares method and the solution has the following from:

$$b_1 = \frac{\boldsymbol{u}_1^{\mathrm{T}}\boldsymbol{t}_1}{(\boldsymbol{t}_1^{\mathrm{T}}\boldsymbol{t}_1)} \tag{4}$$

After going through the above calculations, the residual matrices are calculated as follows:

$$\boldsymbol{E}_1 = \boldsymbol{X} - \boldsymbol{t}_1\,\boldsymbol{p}_1^{\mathrm{T}} \tag{5}$$

$$\boldsymbol{F}_1 = \boldsymbol{Y} - b_1\,\boldsymbol{t}_1\,\boldsymbol{q}_1^{\mathrm{T}} \tag{6}$$

Then the second factor is calculated by decomposing the residuals $\boldsymbol{E}_1$ and $\boldsymbol{F}_1$ using the same procedure as for the first factor. This procedure is repeated until all specified a factors are calculated and the residual matrices $\boldsymbol{E}_a$ and $\boldsymbol{F}_a$ almost has no useful information for regression. The appropriate number of latent variables to retain in the model is generally determined by means of cross-validation[2]. Let $\boldsymbol{T}=[\boldsymbol{t}_1, \boldsymbol{t}_2, \cdots, \boldsymbol{t}_a]$, $\boldsymbol{P}=[\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_a]$, $\boldsymbol{Q}=[\boldsymbol{q}_1, \boldsymbol{q}_2, \cdots, \boldsymbol{q}_a]$ and $\boldsymbol{B}=\mathrm{diag}\{b_1, b_2, \cdots, b_a\}$, the decomposition of $\boldsymbol{X}$ and $\boldsymbol{Y}$ can be written as:

$$\boldsymbol{X} = \boldsymbol{T}\boldsymbol{P}^{\mathrm{T}} + \boldsymbol{E}_a \tag{7}$$

$$\boldsymbol{Y} = \boldsymbol{T}\boldsymbol{B}\boldsymbol{Q}^{\mathrm{T}} + \boldsymbol{F}_a \tag{8}$$

The final prediction from the PLS model can be rearranged and expressed in linear regression form as:

$$\hat{\boldsymbol{y}}^{\mathrm{T}} = \boldsymbol{x}^{\mathrm{T}}\,\boldsymbol{\beta} \tag{9}$$

where $\boldsymbol{\beta}$ is the PLS regression coefficient matrix, which can be calculated from the model parameters $\boldsymbol{W}$, $\boldsymbol{P}$, $\boldsymbol{B}$ and $\boldsymbol{Q}$ [5].

2.2 Prediction model of batch processes

To make the product quality better controlled in batch processes, it is better that the end-product quality can be predicted at the early stage of the batch. Taking the process variables in the early stage of each batch as the model input (noted as $\boldsymbol{x}_{\mathrm{I}}$) and the final product quality variable as the model output (noted as $\boldsymbol{y}_{\mathrm{end}}$), a PLS model (Model I) can be built as illustrated in Sect. 2.1 and represented as:

$$\hat{\boldsymbol{y}}_{\mathrm{end}}^{\mathrm{T}} = \boldsymbol{x}_{\mathrm{I}}^{\mathrm{T}}\,\boldsymbol{\beta}_{\mathrm{I}} \tag{10}$$

where $\boldsymbol{\beta}_{\mathrm{I}}$ is the regression coefficient matrix of Model I.

If some quality measurements are available at one or more discrete times (θ_i, where i=1, 2, ⋯) during the batch, they can be used to correct the final prediction results. At each time θ_i, another two PLS models are built, noted as Model II and Model III, respectively. Model II is used for predicting the quality variables at time θ_i. The predictor variables include process variables and available quality variables measurements up to time θ_i (exclusive of time θ_i). Model II has the following form:

$$\hat{\boldsymbol{y}}(\theta_i)^{\mathrm{T}} = \boldsymbol{x}_{i,\mathrm{II}}{}^{\mathrm{T}}\boldsymbol{\beta}_{i,\mathrm{II}} \tag{11}$$

where $\boldsymbol{x}_{i,\mathrm{II}}{}^{\mathrm{T}} = [\boldsymbol{x}_{\mathrm{I}}^{\mathrm{T}}, \boldsymbol{y}(\theta_1)^{\mathrm{T}}, \cdots, \boldsymbol{y}(\theta_{i-1})^{\mathrm{T}}]$ and $\boldsymbol{\beta}_{i,\mathrm{II}}$ is the regression coefficient matrix of Model II.

Model III is developed for predicting the end-of-batch product quality. The predictor variables for Model III include those in Model II plus the quality variables measurements at time θ_i. Model III has the following form:

$$\hat{\boldsymbol{y}}_{\mathrm{end}}{}^{\mathrm{T}} = \boldsymbol{x}_{i,\mathrm{III}}{}^{\mathrm{T}}\boldsymbol{\beta}_{i,\mathrm{III}} \tag{12}$$

where $\boldsymbol{x}_{i,\mathrm{III}}{}^{\mathrm{T}}=[\boldsymbol{x}_{i,\mathrm{II}}{}^{\mathrm{T}}, \boldsymbol{y}(\theta_i)^{\mathrm{T}}]$, $\boldsymbol{\beta}_{i,\mathrm{III}}$ is the regression coefficient matrix of Model III. Since more process variables and some mid-course quality variables are used in Model III, its performance is expected to be better than that of Model I.

Suppose Models I, II and III have been built from historical data on a set of training batches. For a new batch, the prediction procedure is formulated as follows:

1) The process variables at the early stage of the batch are gathered and end-of-batch product quality variables are predicted from Model I.
2) When measurements of quality variables are obtained at time θ_i, they are compared with the prediction results from Model II:
   a) If the prediction error is large, then there are significant changes in the process operating conditions and the prediction errors for the final product quality from Model I would be very large. Model III should be then used to improve the prediction of the final product quality. If the predicted final product quality differs significantly from the target, then the mid-course correction strategy [6-7] can be applied to bring the product quality back to target.
   b) If the prediction error is small, then Model I is likely to give accurate predictions of the final product quality and Model III may not be required.
3) At each time instant θ_i, i=1, 2, ⋯, repeat Step 2).

3 Model update

Real processes often experience time-varying changes. In this situation, the original PLS model will be unfit and the prediction results will be poor. To overcome this limitation, batch-to-batch model update is applied. After each batch, the model is updated by using the newly obtained data. Suppose the input and output data for the kth batch are $\boldsymbol{x}^{(k)}$ and $\boldsymbol{y}^{(k)}$, respectively. After batch k, the model update can be achieved by simply augmenting the original $\boldsymbol{X}$ and $\boldsymbol{Y}$ matrices with the new data and refitting the PLS model as follows:

$$\boldsymbol{X}^{(k)}=\begin{bmatrix}\boldsymbol{X}^{(k-1)}\\ \boldsymbol{x}^{(k)\mathrm{T}}\end{bmatrix},\ \boldsymbol{Y}^{(k)}=\begin{bmatrix}\boldsymbol{Y}^{(k-1)}\\ \boldsymbol{y}^{(k)\mathrm{T}}\end{bmatrix} \tag{13}$$

$$\left\{\boldsymbol{X}^{(k)},\boldsymbol{Y}^{(k)}\right\}\xrightarrow{\text{PLS}}\left\{\boldsymbol{P}^{(k)},\boldsymbol{B}^{(k)},\boldsymbol{Q}^{(k)},\boldsymbol{\beta}^{(k)}\right\} \tag{14}$$

The updated model is represented as:

$$\hat{\boldsymbol{y}}^{\mathrm{T}}=\boldsymbol{x}^{\mathrm{T}}\boldsymbol{\beta}^{(k)} \tag{15}$$

One drawback of the above method is that the PLS model has to be rebuilt based on merging the new data and old data. So the old data are used repeatedly, which leads to computational inefficiency. Besides, when new batch data are gathered continuously, this method will become more and more inefficient. To deal with this problem, RPLS algorithm [8] can be used to update the model based on the new data and the parameters of the old model. The following theorem was given in Ref. [8].

Theorem 1 *Given a PLS model:*

$$\{\boldsymbol{X},\boldsymbol{Y}\}\xrightarrow{\text{PLS}}\{\boldsymbol{P},\boldsymbol{B},\boldsymbol{Q},\boldsymbol{\beta}\}$$

and a new data pair $\{\boldsymbol{X}_1,\boldsymbol{Y}_1\}$*, performing PLS regression on data pair*

$$\left\{\begin{bmatrix}\boldsymbol{P}^{\mathrm{T}}\\ \boldsymbol{X}_1\end{bmatrix},\begin{bmatrix}\boldsymbol{B}\boldsymbol{Q}^{\mathrm{T}}\\ \boldsymbol{Y}_1\end{bmatrix}\right\}$$

results in the same regression model as performing PLS *regression on data pair*

$$\left\{\begin{bmatrix}\boldsymbol{X}\\ \boldsymbol{X}_1\end{bmatrix},\begin{bmatrix}\boldsymbol{Y}\\ \boldsymbol{Y}_1\end{bmatrix}\right\}$$

The derivation of Theorem 1 was also given in Ref. [8]. For batch process, suppose the parameters of the old PLS model until batch $k-1$ are $\{\boldsymbol{P}^{(k-1)},\ \boldsymbol{B}^{(k-1)},\ \boldsymbol{Q}^{(k-1)},\ \boldsymbol{\beta}^{(k-1)}\}$. The update of the model after batch k using RPLS algorithm is formulated as follows:

$$\boldsymbol{X}^{(k)}=\begin{bmatrix}\boldsymbol{P}^{(k-1)\mathrm{T}}\\ \boldsymbol{x}^{(k)\mathrm{T}}\end{bmatrix},\ \boldsymbol{Y}^{(k)}=\begin{bmatrix}\boldsymbol{B}^{(k-1)}\boldsymbol{Q}^{(k-1)\mathrm{T}}\\ \boldsymbol{y}^{(k)\mathrm{T}}\end{bmatrix} \tag{16}$$

$$\left\{\boldsymbol{X}^{(k)},\boldsymbol{Y}^{(k)}\right\}\xrightarrow{\text{PLS}}\left\{\boldsymbol{P}^{(k)},\boldsymbol{B}^{(k)},\boldsymbol{Q}^{(k)},\boldsymbol{\beta}^{(k)}\right\} \tag{17}$$

Usually the size of the parameters of the old model is smaller than the ever-increasing data sample number. Therefore the computation will be greatly reduced. In order to weaken the influence of the old data, the forgetting factor approach can be adopted. First, old parameter matrices are weighted by a forgetting factor λ, where $0<\lambda\le 1$. Then they are combined with new data matrices as follows:

$$\boldsymbol{X}^{(k)}=\begin{bmatrix}\lambda\boldsymbol{P}^{(k-1)\mathrm{T}}\\ \boldsymbol{x}^{(k)\mathrm{T}}\end{bmatrix},\ \boldsymbol{Y}^{(k)}=\begin{bmatrix}\lambda\boldsymbol{B}^{(k-1)}\boldsymbol{Q}^{(k-1)\mathrm{T}}\\ \boldsymbol{y}^{(k)\mathrm{T}}\end{bmatrix} \tag{18}$$

Finally, PLS regression is performed on the data pair $\{\boldsymbol{X}^{(k)},\boldsymbol{Y}^{(k)}\}$ and the updated model is obtained. The smaller λ is, the faster the model will adapt new data and forget old data. When the process changes slowly, λ can be assigned a value slightly less than 1. When the process changes fast, λ can be assigned a smaller value to discard the old data and adapt the changes faster.

After each batch, Model I, II and III are updated using the above method and then used for prediction of the next batch.

4 Application to a simulated batch MMA polymerization reactor

The batch polymerization reactor studied in this paper is a pilot scale polymerization reactor installed at the department of chemical engineering, Aristotle University of Thessaloniki, Greece, and is shown schematically with its control system in Fig.1. The reaction is the free-radical solution polymerization of MMA with a water solvent and benzoyl peroxide initiator. The jacketed reactor is provided with a stirrer to ensure thorough mixing of the reactants. Heating and cooling of the reaction mixture is achieved by circulating water at an appropriate temperature through the reactor jacket. The reactor temperature is controlled by a cascade control system consisting of a primary PID and two secondary PID controllers. The reactor temperature is fed back to the primary controller whose output is taken as the setpoint of the two secondary controllers. The manipulated variables for the two secondary controllers are the hot and cold water flow rates. The hot and cold water streams are mixed before entering the reactor jacket and provide heating or cooling for the reactor. The jacket outlet temperature is then fed back to the two secondary controllers. A detailed mathematical model covering reaction kinetics and heat and mass balances has been developed and comprehensively validated using the real reactor operation data. Based on this model, a rigorous simulation program was developed and used to generate polymerization data under different batch operating conditions. These data were used to build and validate the PLS model.

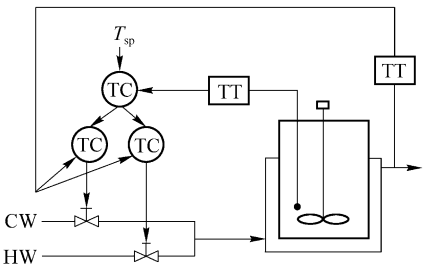

Fig. 1 The pilot batch polymerization reactor

In this study, the batch time of this reactor is set to 180 min. The quality variables to be predicted are the monomer conversion X, the number average molecular weight M_n, and the weight average molecular weight M_w at the end of the batch. The predictor variables of Model I include reactor temperature, reactor jacket inlet and outlet temperatures, and coolant flow rate, measured at 10, 15, 20, 25, and 30 min from the batch start. A mid-course measurement of X at 60 min is available and used in Model II as response variable and in Model III as predictor variable.

In order to simulate the building of the model, 30 batches

were simulated with controls generated from Monte-Carlo simulation. Models I, II and III were built based on these data. Cross-validation method was used to select the appropriate number of latent variables. After the models had been built, another 10 batches were generated under the same initial batch condition to test their performance. The prediction results of X at 60 min from Model II are shown in Fig. 2 and those of end-product quality from Model I and III are shown in Fig. 3. It can be seen that all the models have achieved good performance, and the prediction results from Models I and III are almost the same.

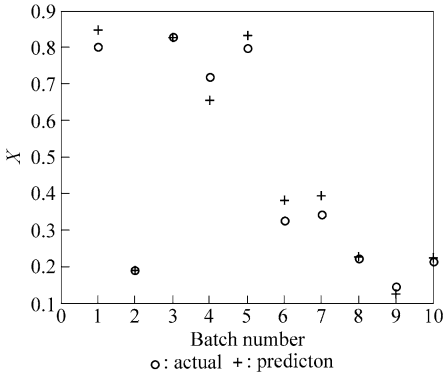

Fig. 2 Prediction results of X at 60 min from Model II

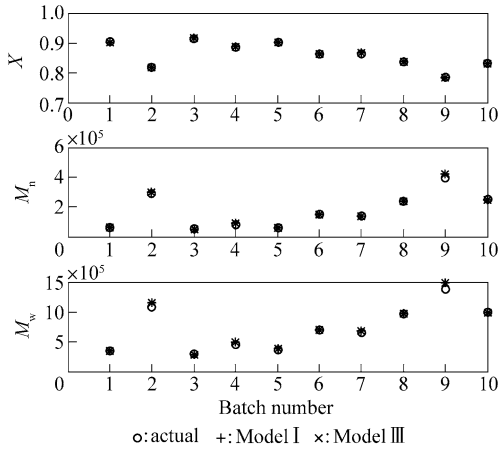

Fig. 3 Prediction results of end-product quality from Model I and III

To test the performance of the proposed model update method, the amount of initial initiator is reduced from its nominal value of 2.5 g to 2.0 g to simulate the presence of 0.5 g of reactive impurities. Recursive PLS algorithm with a forgetting factor of 0.95 is used to update the model after each batch. The prediction results from Model III with model update from batch to batch are shown in Fig. 4. It can be seen that the performance of the model is significantly improved in the second batch. From that batch, the model has achieved quite good prediction performance. To compare the performance of the model before and after model update, relative root-mean-square errors (RRMSEs) are shown in Fig. 5. E_{RRMSE} is defined as:

$$ER_{\mathrm{RMSE}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\hat{y}_i - y_i}{y_i}\right)^2} \tag{19}$$

where y_i and $\hat{y}_i$ are the actual and prediction quality variables, respectively. The results also demonstrate that the performance of the model is greatly improved after model update.

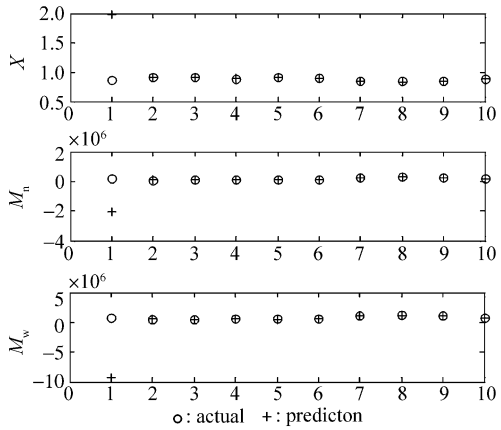

Fig. 4 Prediction results from Model III after model update

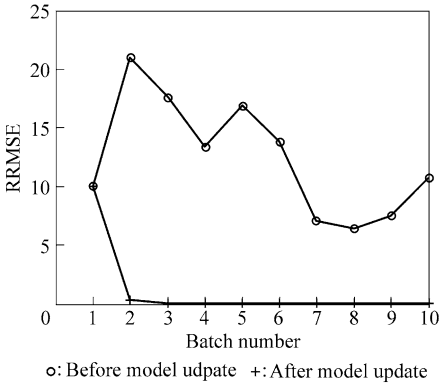

Fig. 5 Comparison of E_{RRMSE} before and after model update

5 Conclusions

To make the product quality better controlled, a PLS model for predicting the end-product quality from the readily available on-line process variables at the early stage of a batch is developed. Besides, additional PLS models are developed by using the available mid-course quality measurements to rectify the final prediction results. To deal with the problem that the process may change with time, after each batch the model is updated based on the new batch data and the old model parameters using RPLS algorithm. An application to a simulated batch MMA polymerization process demonstrates that the proposed method can overcome the problem of process changes and predict the end-of-batch product quality effectively.

Acknowledgements The authors gratefully acknowledge the support of the UK EPSRC (Grant GR/N13319) and thank Prof. C. Kiparissides of the Aristotle University of Thessaloniki, Greece, for providing the polymerization reactor model and the simulation program.

References

1. Wold H., Nonlinear estimation by iterative least squares procedures, In: David F. (ed) Research Papers in Statistics, New York: Wiley, 1966
2. Geladi P., Kowalski B. R., Partial least squares regression: A tutorial, Analytica. Chim. Acta, 1986, 185: 1–17
3. Lorber A., Wangen L. E., Kowalski B. R., A theoretical foundation for the PLS algorithm, J. Chemom., 1987, 1: 19–31
4. Kaspar M., Ray W. H., Partial least squares modeling as successive singular value decompositions, Comput. Chem. Eng., 1993, 17: 985–989
5. Höskuldsson A., PLS regression methods, J. Chemom., 1988, 2: 211–228
6. Flores-Cerrillo J., MacGregor J. F., Control of particle size distributions in emulsion semibatch polymerization using mid-course correction policies, Ind. Eng. Chem. Res., 2002, 41: 1805–1814
7. Yabuki Y., MacGregor J. F., Product quality control in semibatch reactors using midcourse correction policies, Ind. Eng. Chem. Res., 1997, 36: 1268–1275
8. Qin S. J., Recursive PLS algorithms for adaptive data modelling, Comput. Chem. Eng., 1998, 22(4/5): 503–514