Proceedings of the 9th International Symposium on
Dynamics and Control of Process Systems (DYCOPS 2010),
Leuven, Belgium, July 5-7, 2010
Mayuresh Kothare, Moses Tade, Alain Vande Wouwer, Ilse Smets (Eds.)

Latent Variable Modeling of Batch Processes for Trajectory Tracking Control

Masoud Golshan*, John F. MacGregor**

**Department of Chemical Engineering, McMaster University*
Hamilton, Canada, golsham@mcmaster.ca
***Department of Chemical Engineering, McMaster University*
Hamilton, Canada, macgreg@mcmaster.ca

Abstract: Latent Variable Modeling (LVM) of batch processes is explored from the view point of its application to trajectory tracking model predictive controller design. The ability of the models to capture nonlinearity and time-varying properties of batch processes and to provide a well-behaved description of the process are important characteristics to be considered. Furthermore, the importance of requiring as few batches as possible in the modeling step is considered in the discussion of different models. Two previously proposed approaches for batch process modeling (Golshan et al., 2009b) are investigated from the above points of view and benefits of them as well as their drawbacks are specified. Then, a new approach is proposed to overcome the major shortcoming of each previous approach while capturing their major benefits. The impact of the different latent variable modeling approaches on MPC for trajectory tracking is illustrated using a simulation of a Nylon polymerization process.

Keywords: Batch Processes, Latent Variable Modeling, Latent Variable MPC (LV-MPC), Principal Component Analysis (PCA)

1. INTRODUCTION

Batch processes pose several characteristics that make the identification and control studies of batch systems different from those of continuous processes. Nonlinear and time-varying behavior, finite duration, and no equilibrium operating point are among the most important differences between batch and continuous processes. One of the main bottlenecks in the application of advanced control algorithms is the process model. Nonlinear mechanistic models have been used for the control of batch processes (e.g. Clarke-Pringle and MacGregor (1997) and Kravaris et al. (1989)). However, due to the difficulties associated with the development of mechanistic models for real batch processes, empirical models are appealing.

There are two main problems in batch control: (i) the high level problem of controlling the final product quality and (ii) the lower level problem of trajectory tracking. The concern of the former problem is only the product quality at the very end of the batch, while the latter problem involves control over the local batch behavior at every time point throughout the duration of the batch. As a result the requirements for any model for the two problems are very different. In the latter problem the sampling and control frequencies are generally much higher than for the first problem. In this paper we are only concerned with models for the lower level trajectory control problem and the results presented in the paper are only relevant to this problem.

Flores-Cerrillo and MacGregor (2005) proposed the idea of Model Predictive Control over batch trajectories based on Latent Variable models. Golshan et al. (2009a) made improvements to this LV-MPC methodology to allow for offset free tracking and non-stationary disturbances. Their approach is based on a principal component analysis (PCA) model. Golshan et al. (2009b) compared the performance of LV-MPC for batch processes for two different LVMs.

In this paper those two LVMs are investigated in more detail, problems associated with each approach are highlighted and a new modeling approach that avoids the major problems of each of these modeling approaches while retaining the important benefits of both of them is proposed.

2. LATENT VARIABLE MODELING AND IDENTIFICATION

2.1 Rearrangement of Batch Dataset

The structure of the data collected from a batch process is a cube as shown in fig. 1a. There are different approaches for rearranging these data for analysis (Nomikos and MacGregor, 1994, 1995 and Louwerse and Smilde, 2000). The main difference among different unfolding approaches comes from the way they construct a 2-dimensional array (a matrix) from the three dimensional cube of the data set.

2.1.1 *Batch-wise unfolding (BWU)*. Nomikos and McGregor (1995) suggested many possibilities, but proposed the batch-wise unfolding approach shown in Fig.1b as the most logical way for modeling differences among batches. In this approach all the variables at different sample times are put beside each other and each batch constitutes one observation or row in the unfolded matrix. Applying PCA or PLS to this unfolded matrix allows for modeling the time varying and nonlinear behavior of the batches as a locally linear model at every point in time.

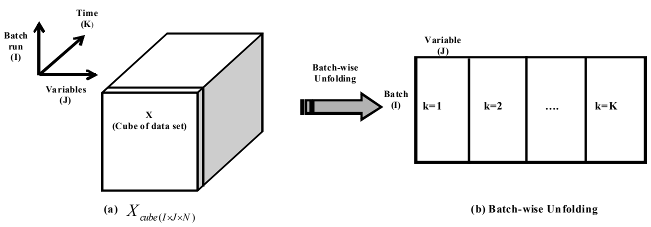

Fig. 1. Batch process dataset and batch-wise unfolding

2.1.2 *Observation-wise unfolding (OWU).* Nomikos and MacGregor (1994, 1995) and Wold et al. (1998) introduced an observation-wise unfolding approach in which variables of each sample time are considered as an independent observation. The schematic diagram is the same as fig. 1b but the time slices are arranged underneath each other instead of beside each other. The underlying assumption behind this approach is the fact that the correlation structure among the dataset does not vary with time and a static average model is enough to explain the process. Hence, one can build a LV model on the OWU matrix using as few as 1-3 batch runs by considering each time step during a batch as an observation. However, the time varying nature of most batch processes is ignored by this approach.

2.1.3 *Observation-wise with time-lag unfolding(OWTU).* To overcome this lack of dynamic modeling ability of the OWU approach a modification inspired by finite time series modeling (Flores-Cerrillo and MacGregor,2005 and Ferrer et al. 2008) has been proposed to include time lags in the observation-wise unfolded batch dataset. This approach is similar to using an ARX model at all time periods during the batch. The resulting model is still an average dynamic model over the whole batch, but it is not a static model. A schematic of the observation-wise unfolding with time-lagging is illustrated in Fig. 2 where *ph* and *fh* are the past and future number of lags taken about each time point.

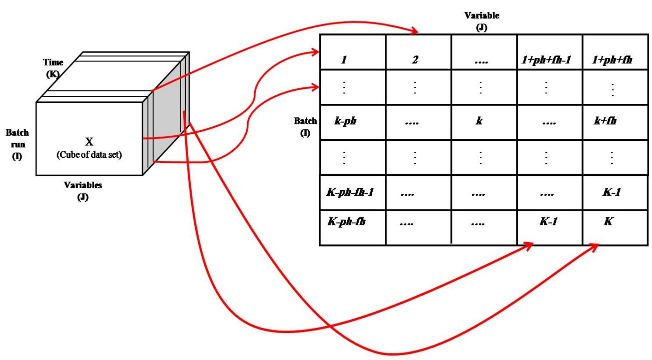

Fig. 2. Observation-wise with time-lag unfolding approach

The parameter "*ph"* should be selected large enough so that the missing data imputation algorithm yields satisfactory performance, and the parameter "*fh"* reflects the prediction horizon.

2.2 Comparison of batch-wise and observation-wise with time-lag unfolding approaches

Batch-wise unfolding puts all the variables at all time lags into one row and then mean centers to remove the average trajectories. A PCA or PLS model then provides different loadings or weights for every deviation variable at every time point throughout the duration of the batch. As a result the latent variable model captures the time-varying nonlinearities throughout the batch as a locally linear model at every time point. Therefore, this BWU approach offers the considerable advantage of capturing the time varying nonlinearities that can be useful for control. However, the data requirements to identify these BWU latent variable models at the high sampling frequencies required for trajectory tracking control present a problem. The number of loading parameters required to capture the locally varying dynamic effects is large and the number of multivariate observations (batches or rows) available for the model building is usually not large. The variance of the resulting loadings is therefore large thereby making the PCA model non-smooth and leads to non-smooth trajectory tracking. Although the use of many observations (batches) smoothens these local effects and makes the aforementioned problem minimal, with the normal number of observations this problem exists. Thus, the batch-wise unfolding approach needs a large number of batches to build a PCA model. This requirement is the biggest bottleneck in modeling batch processes using batch-wise unfolding approach for trajectory tracking control. (Note that this is usually not a problem for the traditional batch data anlaysis, monitoring and end-point control problems where the sampling and control frequencies are much lower).

On the other hand, in the observation-wise with time-lag unfolding approach one gets a huge number of observations using even 1 batch and one only has to identify an average finite ARX model for the whole batch . As shown in figure 2, the total number of observations resulting from each batch is *K-ph-fh.* For example, for a batch with 300 sample times and using a typical value for *ph* and *fh* which is 20 each, one batch results in 260 observations which are much more than enough for building a PCA model. The main drawback of this algorithm is that it is an average model for the batch and cannot handle time varying, nonlinear behavior.

2.3 Regularized batch-wise unfolded models (RBWU)

In this section a third modeling approach is proposed that tries to capture the major benefits of the above two modeling approaches, while avoiding the problems of each one. The new unfolding approach uses elements from both of the preceding approaches. It unfolds batch-wise but also repeats each batch row *L* times each time shifted by one sampling interval. A schematic of the batch-wise with time-shifting approach is shown in figure 3. The parameter *L* is the number of time shifts used. It can be thought of in two ways. One can start with BWU shown in Fig. 1 and then replicate each row *L* times while shifting it by one interval in each case.

Alternatively one can start with OWU with time lags shown in Fig. 2 and use the past and future horizons (ph, fh) to cover the (K-L) time steps of the batch in each row and use only L block rows.

If L=0 (no shifting) this unfolding is simply BWU. But if a small number of shifts (L>0) are used this approach will retain most of the advantages of the BWU approach (capturing time-varying, non-linear behavior), but the model at each time interval will be averaged over L time periods thereby having some of the advantages of the OWTU. For example, if L is small (e.g. L=5) and the number of time intervals K>300, this will not seriously affect the capturing of any time-varying behavior, but it will reduce the variance of the latent variable model loadings (by a factor of L) since these loadings at each time point will now be averaged over L local time periods. The time shifting effectively provides a regularized BWU latent variable model where the loading estimates are effectively averaged over a window of L local time periods. The resulting model will therefore have a smoother variation in the loading coefficients with time. This regularized BWU latent variable model should thus provide smoother model predictive trajectory tracking control. A much similar regularization of the latent variable model could be achieved by performing PCA or PLS with constraints on the rate of change or smoothness of the loadings from one time interval to the next. However, this would lead to a non-linear modification of the latent variable estimation algorithms with essentially the same result as achieved by using this simple time shifting with the standard algorithms.

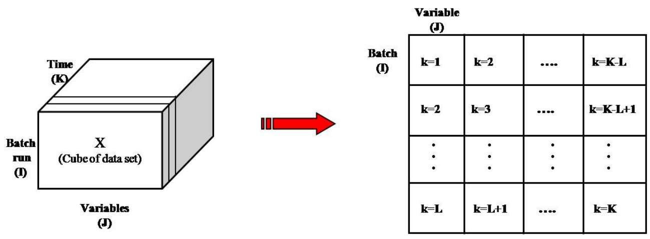

Fig. 3. Regularized Batch-wise unfolding approach

To show how the batch-wise with time-lag unfolding approach leads to regularized PCA, the batch-wise unfolded matrix in the Fig.1 is defined as:

$$X_1 = [a_1, a_2, \dots, a_K] \quad (1)$$

Where a_1,..a_K are matrices of measured variables at sample times $1,\dots K$ (blocks in Figs. 1b and 3, $a_i \in \mathbb{R}^{I \times J}$). Then, the corresponding Batch-wise with Time-lag unfolding approach can be shown as:

$$X_2 = \begin{bmatrix} a_1 & a_2 & \cdots & a_{K-L} \\ a_2 & a_3 & \cdots & a_{K-L+1} \\ & \vdots & & \\ a_L & a_{L+1} & \cdots & a_K \end{bmatrix} \quad (2)$$

$$Cov(X_2) = \frac{1}{LI} X_2^T X_2 = \frac{1}{LI} \begin{bmatrix} a_1^T & a_2^T & \dots & a_L^T \\ a_2^T & a_3^T & & a_{L+1}^T \\ \vdots & \vdots & & \vdots \\ a_{K-L}^T & a_{K-L+1}^T & & a_K^T \end{bmatrix} \begin{bmatrix} a_1 & a_2 & \dots & a_{K-L} \\ a_2 & a_3 & \dots & a_{K-L+1} \\ & & \vdots & \\ a_L & a_{L+1} & \dots & a_K \end{bmatrix} \quad (3)$$

Using the outer product definition of the matrices:

$$Cov(X_2) = \frac{1}{LI} \left\{ \begin{bmatrix} a_1^T \\ a_2^T \\ \vdots \\ a_{K-L}^T \end{bmatrix} \begin{bmatrix} a_1 & a_2 & \cdots a_{K-L} \end{bmatrix} + \begin{bmatrix} a_2^T \\ a_3^T \\ \vdots \\ a_{K-L+1}^T \end{bmatrix} \begin{bmatrix} a_2 & a_3 & \cdots a_{K-L+1} \end{bmatrix} + \cdots + \begin{bmatrix} a_L^T \\ a_{L+1}^T \\ \vdots \\ a_K^T \end{bmatrix} \begin{bmatrix} a_L & a_{L+1} & \cdots a_K \end{bmatrix} \right\}$$

Hence,

$$Cov(X_2) = \frac{1}{L} \left(Cov(X_{1,1:K-L}) + Cov(X_{1,2:K-L+1}) + \dots + Cov(X_{1,L:K}) \right) \quad (4)$$

Where $X_{1,i:K-j}$ means the blocks "i" to "K-j" of matrix X_1. The above equation shows that the covariance of the resulting matrix from the batch-wise with time-shifting approach is an average of the original covariance matrix over L sample times. Thus, the resulting covariance matrix is a regularized of the original covariance matrix.

A benefit of the observation-wise with time-lag unfolding approach was that using this type of unfolding the batch can be modeled using only 1-3 batch runs (although one only obtains an average model for the batch). However, with this regularized approach we get some of the same benefit without the liability. For example by using 20 batches and L= 5-10 one can get 100-200 observations (rows) in the unfolded matrix which is perfect for implementing Multiphase LV-MPC (MLV-MPC) proposed in Golshan et al. (2009a). In general the choice of the number of time shifts (L - the regularization parameter) will depend upon the rapidity of the time varying behavior of the batch versus the number of batches available for the model identification.

2.4 Multi-phase Modeling

LV modeling of batch processes for trajectory tracking control using the batch-wise unfolding approach leads to a very large global LV model because of the large number of time intervals over the batch duration. This is not desirable as it requires many latent variables (which implies many batches may be needed in the training set), it leads to ill-conditioned matrices in the model used during the control computations, and does not focus on the local behavior of the trajectories. Therefore, utilization of multi-phase LV models, as presented in Golshan et al. (2009a) is necessary or at least preferred. The multiphase modeling approach is based on identifying multiple phases within the batch, partitioning of the dataset according to phases, considering overlap between two adjacent phases, and building PCA models for each phase.

The same multiphase modeling approach is applicable to a batch-wise with time-shifting unfolded dataset (regularized BWU model). The resulting matrix from batch-wise unfolded model with time-shifting is considered as a new batch-wise unfolded matrix and the multiphase modeling is performed.

However, for the observation-wise with time-lag unfolding approach multiphase modeling has both good and bad effects and application of multiphase models may not always improve the results. The observation-wise with time-lag unfolding approach by its definition is applicable to the processes that are not highly time-varying and whose behavior can be modeled by an average dynamic model. When the process is time-varying, the observation-wise with time-lag unfolding approach can be applied in multi-phase framework. However, as explained in Golshan et al. (2009b), it results in switching between different local models. This switching manner may cause some inconsistencies at the switching times that lead to performance deterioration for a transient period.

2.5 Identification

The training data can be the data from the previous batches run under normal conditions augmented with additional batches executed according to identification experiments to provide information on the causal relationships between the manipulated variables and the controlled variables at every time interval throughout each phase. The direct identification approach based on closed loop data is used in this study. Closed loop identification is preferred over open loop identification for batch processes in order to maintain the process close to its desired trajectories and to minimize the final product quality variations. A Random Binary Signal (RBS) is added on top of the manipulated variable trajectories coming from an existing controller (PID) to provide some additional excitation of the process. The RBS dither signal is chosen to have its switching frequency in a suitable range (~ 1/6-1/3 of the dominant time constant of the process). The closed-loop design of identification experiments for identifying models of time varying, finite duration batch systems has not been discussed in the literature, but is the topic of current research by the authors. Therefore, we do not go into the identification issues in this paper except to note a few qualitative issues and some observations from the simulation studies presented below. In this study it was found that if the observation-wise with time-lag unfolding approach is used, models were identifiable from historical batches under pure feedback control with no additional excitation. This somewhat surprising result might be explained by analogy with the closed-loop identification of linear time-invariant systems (Ljung, 1999). There it has been shown that closed-loop identifiability is satisfied if one switches between a sufficient number of linear controllers or if the control is nonlinear. Here the batch system is non-linear and time varying and so with a fixed PID controller it is analogous to the controller being time varying or nonlinear for a linear system. Furthermore, there are time-varying setpoints in the training data generation that also helps to satisfy the identifiability conditions. But, this is for the modeling approach based on observation-wise with time-lag unfolding approach which is a linear time-invariant model over an entire batch phase. However, for the other 2 modeling approaches (based more on BWU) where the time-varying behavior is being modeled, these conditions may not be enough to ensure identification of an adequate model (with no fixed underlying true model identifiability conditions are not clearly defined). The historical batch data are very important for providing models for the effects of inherent disturbances in the batch process and their influence on the behavior of the evolving trajectories. This information is essential for the prediction of the future trajectories and ensuring no steady state offset in the control as discussed in the next section.

Another important issue in the selection and design of identification experiments is the inclusion of experiments using somewhat different set-point trajectories. This is mainly of importance if the MPC is to be required to track a range of set-point trajectories such as might be needed for achieving different grades of the product. In practice, historical batch data would usually be available on these different grades and could be included in the training data.

3. CONTROL METHODOLOGY

In order to assess the performance of the aforementioned modeling approaches, they are used in the course of a trajectory tracking control problem. The selected control algorithm is the recently proposed Latent Variable Model Predictive Control (LV-MPC) (Golshan et al. 2009a). The Control objective is to find an optimal manipulated variable path to make the controlled variable track its desired trajectory. Two control formulations are proposed in Golshan et al. (2009a). However, for the sake of briefness, one of them that the three modeling approaches are tested by is briefly explained in this paper.

Assume the control algorithm is in the middle (sample time k) of a new batch and ζ_k is defined by equation (5).

$$\zeta_k^T = [x_{me,k}^T, y_{cv,k}^T, u_{c,k}^T, y_{sp,k}^T] \qquad (5)$$

Where x_{me}, y_{cv}, u_c, and y_{sp} are measured variables, controlled variables, manipulated variables, and set point variables respectively. The existing information in the current batch can be rearranged as follows:

$$\begin{aligned} x_k^T &= [\zeta_j^T |_{j=1:k-1}, x_{me,k}^T, y_{cv,k}^T, y_{sp,k}^T \qquad y_{sp,j}^T |_{j=k+1,\ldots,K} \\ &\quad u_{c,k}^T, u_{c,j}^T |_{j=k+1,\ldots,K-1}, x_{me,j}^T |_{j=k+1,\ldots,K} \qquad y_{cv,j}^T |_{j=k+1,\ldots,K}] \\ &= [x_{P1,k}^T, x_{P2,k}^T; x_{f1,k}^T, x_{f2,k}^T] \end{aligned} \qquad (6)$$

The corresponding loadings, P matrix in the PCA model, can also be separated in the same way. The past data can be used to estimate the score of the current batch, τ_k, which summarizes the current position of the batch using missing data imputation methods (Arteaga and Ferrer, 2002). Then a correction to the score, $\Delta\hat{\tau}_k$ can be computed to bring the batch trajectories closer to their desired values by optimizing the following quadratic objective:

$$\min_{\Delta\hat{\tau}_k} \frac{1}{2}\left(\hat{y}_{cv}-y_{sp}\right)^T V_1\left(\hat{y}_{cv}-y_{sp}\right)+\hat{u}_f^T V_2\hat{u}_f \qquad (7)$$
$$= \frac{1}{2}\left(\hat{x}_{f2}-x_{P2}\right)^T V_1\left(\hat{x}_{f2}-x_{P2}\right)+\hat{u}_f^T V_2\hat{u}_f$$

Comparing equations (7) and (6), y_{cv} and y_{sp} correspond to x_{f2} and x_{p2} respectively. Using the PCA model it can be shown that x_{f2} and u_f can be written as a function of the decision variable, $\Delta\hat{\tau}_k$ (Golshan et al., 2009a):

$$\hat{x}_{f2} = P_{f2}\left(P_f^T P_f\right)^{-1}\left(\hat{\tau}_k + \Delta\hat{\tau}_k - P_p^T x_{p,k}\right) \qquad (8)$$

$$\hat{u}_f = P_{uf}\left(\hat{\tau}_k + \Delta\hat{\tau}_k\right) \qquad (9)$$

Combining equations (7), (8), and (9) and following optimization procedures, one can obtain the optimum $\Delta\hat{\tau}_k$. $\Delta\hat{\tau}_k$ contains information on the adjustments to all future inputs till the end of the batch ("infinite" horizon control Golshan et al. ,2009a). The corresponding u_f can be computed using the PCA model relationship in equation (9). Then according to MPC algorithm its first element is implemented to the process. At the next sample time the same procedure will be repeated.

It should be noted that in this control method, it is possible to either solve the optimization problem analytically, if there is no hard constraint, or solve it by numerical optimization methods, in case of existence of hard constraints. However, constraints are generally much less of a problem in batch processes.

In the above control formulation the LQ matrices (V_1 and V_2) should be chosen carefully. V_1 is a diagonal matrix that can be exponentially weighted to put stress on the early future values rather than the far values. However, V_2 matrix should be a derivative matrix to penalize the changes in the MV's.

4. SIMULATION RESULTS

The case study presented in this paper is a MIMO control problem in which temperature and pressure of a Nylon 6,6 polymerization reactor is controlled using the pressure of steam flow in the jacket and vent rate through the valve on top of the reactor. This is a constrained problem in which the vent rate cannot be less than zero and the steam pressure cannot be less than 4 psi and more than 52 psi. A schematic of these case studies are shown in Fig.5. For detailed information see Russell et al. (1998).

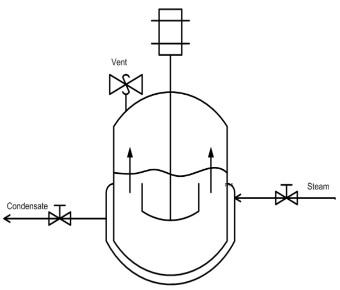

Fig. 5. Schematic Diagram of the Nylon 6,6 problem

In order to explore the full comparison of the modeling approaches many different scenarios have been tested. Though, Due to the space limitation only one example from each modeling approach is presented in this paper. The results of applying the LV-MPC to this case study are sorted in Figs. 6-8 and table 1. In all cases the model identification was performed using closed-loop data.

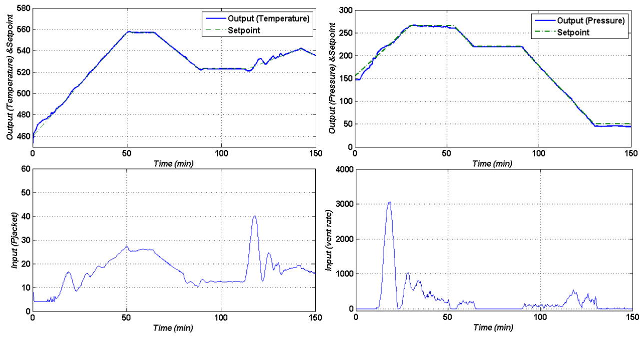

Fig.6. Control based on batch-wise unfolding modeling approach

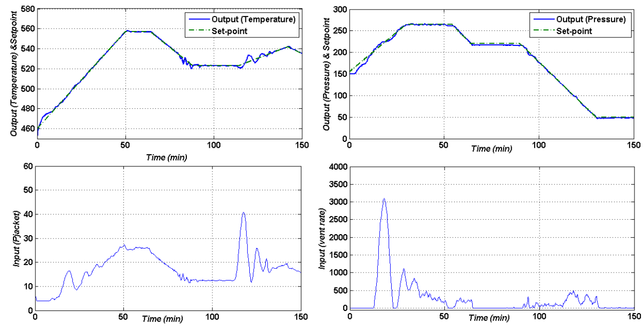

Fig.7. Control Based on Observation-wise with Time-lag unfolding modeling approach

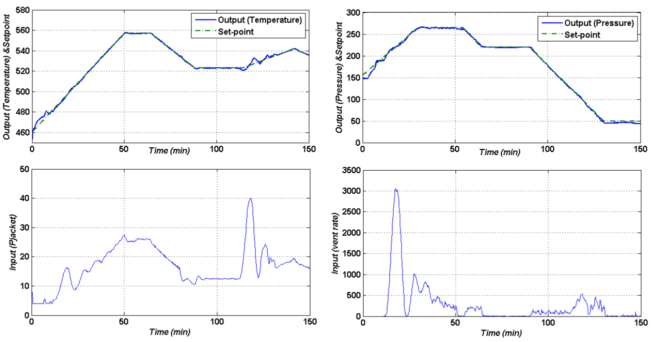

Fig. 8. Control Based on Batch-wise with Time-shift unfolding modeling approach

Table 1. LVMPC on the MIMO Nylon 6,6 Process based on three modeling approaches

Performance Index	BWU	OWTU	RBWU
RMSE(y_T-$y_{sp,T}$)	1.0417	1.2043	1.0982
STD(Δu_T)	0.4667	0.4709	0.4425
RMSE(y_p-$y_{sp,p}$)	2.9918	2.8300	2.8214
STD(Δu_p)	43.9659	43.2539	43.3021

Where RMSE is the abbreviation for root mean square error and STD represents the standard deviation. Fig. 6 shows that the multi-phase batch-wise unfolding approach with 10 phases of equal size and 30 batches in the training dataset leads to a good tracking of the reference trajectory. However, as mentioned before, too many batches need to be used for identification which may not exist in practical situations.

Inadequate number of batches may lead the practitioner to use the observation-wise with time-lag unfolding approach. The performance of the observation-wise with time-lag unfolding approach using one phase throughout the batch and 2 observations in the training dataset is shown in Fig.7. The tracking quality of the temperature loop is slightly worse than that in Fig. 6. More oscillations in tracking may result because of the average time invariant model for the whole batch that this method leads to. The case study is a time varying process, and one phase may not be sufficient to model this process. Though, as explained before, the inconsistencies caused by switching between different models on this specific process decrease the quality of the LV-MPC performance. After running a few batches, one may switch to BWU or regularized BWU. However, this modeling approach has the least data requirements for modeling. Moreover, it requires only the data of the current PI controller that is running the process without any additional excitation by dither signal (only historical batches).

Fig.8 illustrates the performance of the LV-MPC based on the batch-wise with time-shift unfolding approach using 10 phases, 15 observations, and 5 time shift units (L=5). It tries to compensate the shortcomings of both previous modeling approaches. It requires fewer observations and produces a smoother tracking as compared to the BWU modeling approach. In fact, when the number of observations is large or the process does not have large noises (as is the case for this process) the difference between the batch-wise and batch-wise with time-shift unfolding approaches becomes minimal as explained in section 2.2. However, in the above examples, regularized BWU needs half of the observations (batch runs) used in the training dataset of BWU, but gives better trajectory tracking.

5. CONCLUSIONS

Modeling of batch processes from the view point of its application for trajectory tracking control is scrutinized. In this paper, the previously proposed modeling approaches (Golshan et al., 2009b) are investigated in more details. BWU is more suitable for modeling the nonlinearity and time-varying characteristics of batches, but needs a large number of batch runs in the training dataset. Otherwise, noisy trajectory tracking is obtained. On the other hand, the OWU with time-lag unfolding approach requires as few as 2 batches in the training dataset and yields a smooth PCA model. However, it leads to modeling an average process dynamics.

The proposed regularized batch-wise modeling approach tries to capture the advantages of BWU and OWTU approaches while avoiding their disadvantages. In this paper, the BWU, OWTU, and Regularized BWU are tested in the course of the recently proposed LV-MPC methodology (Golshan et al., 2009a) on a MIMO case study. The results show that the regularized BWU modeling approach leads to superior trajectory tracking. This approach leads to a smooth (regularized) PCA model which in turn leads to a smoother trajectory tracking. Furthermore, it requires fewer batch runs in the dataset as compared with BWU modeling approach.

REFERENCES

Arteaga, F., Ferrer, A. (2002). Dealing with missing data in MSPC: several methods, different interpretations, some examples, *J. Chemom*. 16, 408-418.

Clarke-Pringle, T., and MacGregor, J. F. (1997). Nonlinear adaptive temperature control of multi-product, semi-batch polymerization reactors. *Comput. Chem. Eng.* 21 (12), 1395–1409.

Ferrer, A., Aguado, D., Vidal-Puig, S., Prats, J. M., Zarzo, M. (2008). PLS: a versatile tool for industrial process improvement, and optimization, *Appl. Stochastic Models Bus. Ind., 24,* 551–567

Flores-Cerrillo, J., and MacGregor, J. F. (2005). Latent Variable MPC for trajectory tracking in batch processes. *J. Proc. Cont.* 15, 651-663.

Garcia-Munoz, S., MacGregor, J. F. (2004), Model predictive monitoring of batch processes, *Ind. Eng. Chem. Res.,* 43, 5929-5941.

Golshan, M., MacGregor, J. F., Bruwer, M. J., Mhaskar, P. (2009a). Latent Variable Model Predictive Control for trajectory tracking in batch processes, Submitted for publication.

Golshan, M., MacGregor, J. F., Bruwer, M. J., Mhaskar, P. (2009b). Latent Variable MPC for trajectory tracking in batch processes: role of model structure, *Proceedings of American Control Conference*, St. Louis, U.S.A., June 10-12, 4779-4784.

Kravaris, C., Wright, R.A., and Carrier, J.F. (1989). Nonlinear controllers for trajectory tracking in batch processes. *Comput. Chem. Eng.* 13, 73-82.

Ljung, L.(1999). *System Identification - Theory For the User*, 2nd ed, PTR Prentice Hall, Upper Saddle River, N.J., 1999.

Louwerse, D. J., and Smilde, A. K. (2000). Multivariate statistical process control of batch processes based on three-way models. *Chem. Eng. Sci.* 55, 1225-1235.

Nomikos, P., MacGregor, J. F. (1994). Monitoring batch processes using multiway principal component analysis. *AIChE J.*, 40, 1361-1375.

Nomikos, P., and MacGregor, J.F. (1995). Multivariate SPC charts for catch processes. *Technometrics,* 37, 41-59.

Russell, S. A., Robertson, J. H., Lee, J. H., and Ogunnaike, B. A. (1998). Control of product quality for batch nylon 6,6 autoclaves. *Chemical Engineering Science , 53* (21), 3685-3702.

Wold, S., Kettaneh, N., Fridén, H., and Holmberg, A. (1998). Modelling and diagnostics of batch processes and analogous kinetic experiments. *Chemom. Intell. Lab. Syst.* 44(1-2), 331-340.