On the Use of Partial Least Squares (PLS) and Balancing for Nonlinear Model Reduction

Chuili Sun and Juergen Hahn

Abstract—**Model reduction is an important technique to reduce the complexity of nonlinear process models for controller design. The goal is to approximate the model as accurate as possible while at the same time achieve a speedup in computation time. The technique presented in this paper combines balancing with partial least squares (PLS) for achieving a small, control-relevant, reduced-order model. Two specific methods for using PLS are considered: one is for the balanced residualization for ODE systems and the other is for reducing the complexity of algebraic equations in DAE systems. Corresponding to these two cases, a fixed bed reactor (ODE) and a distillation column model (DAE) are studied to illustrate the use of this balancing/PLS combination for model reduction.**

I. Introduction

Nonlinear process control has become increasingly popular in chemical process industries over the past few years [1]. However, due to the extensive computational effort, complex nonlinear models are difficult to implement for online process control. To address this issue, model reduction techniques are used to reduce the size of the model while retaining control-relevant information.

Research on nonlinear model reduction not only dealt with systems described by ordinary differential equations (ODE) [2]-[3], but also systems described by partial differential equations (PDE) [4] and differential-algebraic equations (DAE) [5]. The goal of each procedure is to replace a high-dimensional description of a process with a model of lower order that approximates the behavior of the system. This approximation can be derived based upon knowledge about the original system as is the case in balanced reduction or via system identification, e.g. artificial neural networks (ANN). For nonlinear systems with a low to medium degree of nonlinearity, some linear methods, e.g. partial least squares (PLS) as introduced by Wold [6], can also provide excellent performance for multivariate approximations [7]. However, only using a linear system identification technique can lead to problems for models with a higher degree of nonlinearity.

In this paper, a combination of balancing and PLS is investigated for nonlinear model reduction of ODE and DAE systems. By combining a model-based method, i.e. nonlinear balancing, with a linear statistical approach, i.e. PLS, it is possible to reduce both size and complexity of a strongly nonlinear model while retaining the control-relevant input-output behavior. The outline of this paper is given as follows. Preliminary work is briefly reviewed in Section II, where the basic ideas of nonlinear model reduction and of PLS are presented. The following section discusses the new technique for model reduction of ODE and DAE systems. Two case studies are presented in Section IV and Section V contains the conclusions.

Manuscript received February 4, 2005.

C. Sun is with Dept. of Chemical Engineering, Texas A&M University, College Station, TX 77843 USA (email: treacysun@tamu.edu).

J. Hahn is with Dept. of Chemical Engineering, Texas A&M University, College Station, TX 77843 USA (phone: 979-8453568; fax: 979-8456446; e-mail: hahn@tamu.edu).

II. Preliminaries

A. Model Reduction of ODE Systems

ODE systems of the following form

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x) \end{aligned} \tag{1}$$

will be investigated for model reduction. A commonly used reduction procedure is balanced truncation via empirical gramians [3] or covariance matrices [8]. This is achieved by computing a transformation

$$\bar{x} = Tx \tag{2}$$

that transforms the original system into a balanced form, where the singular values are ordered from large to small in the balanced gramians or covariance matrices. These singular values provide a measure for the importance of specific states to the input-output behavior. The less important states can then be reduced by balanced truncation

$$\begin{aligned} \dot{\bar{x}}_1 &= PTf(T^{-1}\bar{x},u) \\ \bar{x}_2 &= \bar{x}_{2,ss} \qquad \text{where } \bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} \\ y &= h(T^{-1}\bar{x}) \end{aligned} \tag{3}$$

$P = [I_{k\times k} \quad 0]$, k: number of retained states

One drawback of this technique is that the steady state behavior is not retained in the reduced system. To overcome this disadvantage, balanced residualization has been introduced [2] where the derivatives of the less important

states are set equal to zero

$$\begin{aligned} \dot{\bar{x}}_1 &= P_1 T f(T^{-1}\bar{x}, u) \\ 0 &= P_2 T f(T^{-1}\bar{x}, u) \qquad \text{where } \bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} \\ y &= h(T^{-1}\bar{x}) \end{aligned} \tag{4}$$

$$P_1 = [I_{k\times k} \quad 0], \quad P_2 = [0 \quad I_{(n-k)\times(n-k)}]$$

n: number of states, k: number of retained states

which can be rewritten as

$$\begin{aligned} \dot{\bar{x}}_1 &= \bar{f}_1(\bar{x}_1, \bar{x}_2, u) \\ 0 &= \bar{f}_2(\bar{x}_1, \bar{x}_2, u) \\ y &= \bar{h}(\bar{x}_1, \bar{x}_2) \end{aligned} \tag{5}$$

While a residualized system will have better steady-state performance than a truncated system, it is only a reduced model in the sense that some differential equations have been replaced by algebraic ones. Further reduction of this type of system will be presented in Section III.

B. Model Reduction of DAE Systems

DAE systems under investigation are of the form

$$\begin{aligned} \dot{x} &= f(x, z, u) \\ 0 &= g(x, z) \\ y &= h(x) \end{aligned} \tag{6}$$

where the vector x represents the dynamic states and z the algebraic variables.

The transformation for the dynamic states, T_1, is computed in the same way as for ODE systems and the transformation for the algebraic variables, T_2, is obtained by singular value decomposition of a covariance matrix describing the correlation of the algebraic variables along system trajectories [5]. Using these transformations

$$\bar{x} = T_1 x, \quad \bar{z} = T_2 z \tag{7}$$

the transformed system is given by

$$\begin{aligned} \dot{\bar{x}} &= T_1 f(T_1^{-1}\bar{x}, T_2^{-1}\bar{z}, u) = \bar{f}(\bar{x}, \bar{z}, u) \\ 0 &= g(T_1^{-1}\bar{x}, T_2^{-1}\bar{z}) = \bar{g}(\bar{x}, \bar{z}) \\ y &= h(T_1^{-1}\bar{x}) = \bar{h}(\bar{x}) \end{aligned} \tag{8}$$

Some states or variables in (8) are of lesser importance and can be reduced by truncation, resulting in the following system

$$\begin{aligned} \dot{\bar{x}}_1 &= P_1 \bar{f}(\bar{x}, \bar{z}, u) \\ \bar{x}_2 &= \bar{x}_{2,ss} \\ 0 &= P_2 \bar{g}(\bar{x}, \bar{z}) \\ \bar{z}_2 &= \bar{z}_{2,ss} \\ y &= \bar{h}(\bar{x}) \end{aligned} \tag{9}$$

where

$$\bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix}, \ \bar{z} = \begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \end{bmatrix}, \ P_1 = [I_{k\times k} \quad 0], \ P_2 = [I_{m\times m} \quad 0],$$

k is the number of retained differential equations and m refers to the number of retained algebraic variables.

By a change of notation, (9) can be rewritten as

$$\begin{aligned} \dot{\bar{x}}_1 &= \hat{f}(\bar{x}_1, \bar{z}_1, u) \\ 0 &= \hat{g}(\bar{x}_1, \bar{z}_1) \\ y &= \hat{h}(\bar{x}_1) \end{aligned} \tag{10}$$

The system in (10) is a special case of the system described by (5). Therefore, similar techniques can be applied to further reduce this type of system.

C. Partial Least Squares

PLS is a linear system identification method that determines a latent space with orthogonal principle factors to approximate the original input-output space. If an independent matrix X and a dependant matrix Y are given then a PLS model of the form $X = TP^{\mathrm{T}} + E$, $Y = UQ^{\mathrm{T}} + F$, can be constructed where T and U are the score matrices, P and Q the loading matrices, and E and F are the residual matrices of X and Y, respectively. PLS extracts latent variables (also called principle factors) by analyzing the sample covariance matrix $(X^{\mathrm{T}}Y)(Y^{\mathrm{T}}X)$. These latent variables can not only capture the variance of X, but also maximize the covariance between each X score and the corresponding Y score. The latent variables are orthogonal and therefore independent of one another. In most cases, the first few latent variables can capture the most useful information between X and Y. The NIPALS algorithm [9] is an efficient way to compute the PLS latent variables sequentially.

1) Data Preprocessing

To apply PLS, it is useful to scale the data sets to simplify the calculations. The scaled form of a specific variable x is given by

$$\hat{x} = \frac{x - \bar{x}}{\mathrm{std}(x)} \tag{11}$$

where $\bar{x}$ is the mean value of the variable x and std (x) refers to the standard deviation.

All variables in the independent and dependent block are preprocessed in this way and all variables involved in the PLS procedure are scaled.

2) Calibrating a PLS Model

The main idea of PLS is to find an inner relationship between X and Y. This relationship can be approximated by their score matrices T and U with $U = B*T$, where B is the regression matrix. The NIPALS algorithm is applied to construct a PLS model via an iterative procedure.

3) Data Prediction

Prediction of the dependent matrix from the independent matrix is the most important part of PLS. This is performed by multiplying the independent matrix by the regression coefficient matrix B which is obtained in the calibration step, i.e., $Y = B * X$.

III. USE OF PLS IN MODEL REDUCTION

A. Use of PLS for Residualization for ODE Systems

1) Approximation of Residualization

The reduced system given by (5) is in fact a differential algebraic equation system where the retained states $\bar{x}_1$ are described by differential equations while the reduced states $\bar{x}_2$ can be represented as a vector of algebraic variables. A DAE solver is required to solve this type of model. However, solving a DAE system often requires a larger computational effort than solving an ODE system with the same number of states, which would defeat the purpose of nonlinear model reduction. Therefore, further reduction may be required for nonlinear balanced residualization.

The vector $\bar{x}_1$ from (5) contains states that are the most important ones for the input-output behavior of the system and $\bar{x}_2$ is only used as a correction term for the steady state behavior. As $\bar{x}_2$ only contributes to a lesser degree to the process' behavior it can be further reduced without significantly affecting the input-output behavior of the system.

Consider the implicit algebraic equation $0 = \bar{f}_2(\bar{x}_1, \bar{x}_2, u)$ in (5). $\bar{x}_2$ is dependent on $\bar{x}_1$ and on the input u in a static manner. Since $\bar{f}_2$ is nonlinear, it is hard to obtain an explicit expression for $\bar{x}_2$. However, a system identification technique can be applied to approximate $\bar{x}_2$ given $\bar{x}_1$ and u

$$\bar{x}_2 = \hat{g}(\bar{x}_1, u) \tag{12}$$

where $\hat{g}$ represents an identified expression, e.g. an artificial neural network (ANN) or a partial least squares (PLS) model. In this work, PLS is used to obtain this approximation, as it is often sufficient to represent these states of lesser importance by a linear approximation. The reduced system is given by

$$\begin{aligned} &\dot{\bar{x}}_1 = \bar{f}_1(\bar{x}_1, \bar{x}_2, u) \\ &\bar{x}_2 \text{ is predicted by PLS using } (\bar{x}_1, u) \text{ as inputs} \\ &y = h(\bar{x}_1, \bar{x}_2) \end{aligned} \tag{13}$$

If the number of states to be reduced is large, then it can happen that the large number of resulting algebraic equations can cause numerical problems. However, as many of the algebraic variables do not contribute to the control-relevant behavior, it is possible to use a combination of residualization and truncation to address this situation and compute a reduced-order model of the following form:

$$\begin{aligned} &\dot{\bar{x}}_1 = P_1 T f(T^{-1}\bar{x}, u) \\ &0 = P_2 T f(T^{-1}\bar{x}, u) \qquad \text{where } \bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \end{bmatrix} \\ &\bar{x}_3 = \bar{x}_{3,ss} \\ &y = h(T^{-1}x) \\ &P_1 = [I_{k\times k} \quad 0 \quad 0], \; P_2 = [0 \quad I_{m\times m} \quad 0] \end{aligned} \tag{14}$$

k: number of retained differential equations

m: number of retained algebraic variables

Following the same approximation procedure as in (12) and (13), the reduced system can be described by

$$\begin{aligned} &\dot{\bar{x}}_1 = \bar{f}_1(\bar{x}_1, \bar{x}_2, u) \\ &\bar{x}_2 \text{ is predicted by PLS using } (\bar{x}_1, u) \text{ as inputs} \\ &\bar{x}_3 = \bar{x}_{3,ss} \\ &y = h(\bar{x}_1, \bar{x}_2) \end{aligned} \tag{15}$$

Note that in the approximations (13) and (15), $\bar{x}_2$ also depends on the input u and, therefore, the independent matrix consists of $\bar{x}_1$ and u.

2) Implementation of PLS

To compute a PLS model which approximates $\bar{x}_2$ given values of $\bar{x}_1$ and u, the independent matrix is required to contain $\bar{x}_1$ and input u, and the dependent matrix consists of $\bar{x}_2$. The training data set can be collected by the following procedure:

Step 1: Simulate the original model

Uniformly distributed random perturbations of the inputs u are used to excite the original model starting from the steady state operating point. The values of the inputs and the states are recorded along each trajectory. For convenience of notation, the data matrix of states is represented by A and the input matrix is referred to as B in the following.

Step 2: Obtain the training data

Since $\bar{x}_1$ and $\bar{x}_2$ are states/variables of the transformed system, the data collected in step 1 also need to be transformed:

$$\bar{A} = TA \quad (T \text{ is the transformation matrix}) \tag{16}$$

If the number of retained states is chosen to be equal to k, then the first k rows of $\bar{A}$ correspond to the data for $\bar{x}_1$, denoted as $\bar{X}_1$ and the following (n-k) rows in (4) or m rows in (14) are data for $\bar{x}_2$, denoted by $\bar{X}_2$.

The independent matrix X and dependent matrix Y before scaling are constructed as in (17), respectively.

$$X = \begin{bmatrix} \bar{X}_1 \\ B \end{bmatrix} \qquad Y = [\bar{X}_2] \tag{17}$$

Step 3: Scale the training data

X and Y are scaled to have zero mean and unit variance as shown in (11). For convenience of notation, the scaled matrices are still denoted as X and Y in the following.

After obtaining the training data set, the NIPALS

algorithm is applied to construct the PLS model including the regression matrix, referred to as *R*. By use of the regression matrix, the reduced systems corresponding to (13) and (15) are given by (18) and (19), respectively.

$$\begin{aligned} \dot{\bar{x}}_1 &= \bar{f}_1(\bar{x}_1, \bar{x}_2, u) \\ \bar{x}_2 &= R * [\bar{x}_1, u] \\ y &= h(\bar{x}_1, \bar{x}_2) \end{aligned} \tag{18}$$

$$\begin{aligned} \dot{\bar{x}}_1 &= \bar{f}_1(\bar{x}_1, \bar{x}_2, u) \\ \bar{x}_2 &= R * [\bar{x}_1, u] \\ \bar{x}_3 &= \bar{x}_{3,ss} \\ y &= h(\bar{x}_1, \bar{x}_2) \end{aligned} \tag{19}$$

B. Use of PLS for DAE System Reduction

1) Further Reduction of DAE Systems

Although the system from (10) is smaller than the original system, it is possible to further reduce the system while retaining most of process information. This is especially important as most DAE systems contain significantly more algebraic than differential equations.

Consider the algebraic equation $0 = \hat{g}(\bar{x}_1, \bar{z}_1)$, where $\bar{z}_1$ is dependent on $\bar{x}_1$ in a static manner. This relationship can be approximated by system identification, e.g. PLS:

$$\begin{aligned} \dot{\bar{x}}_1 &= \hat{f}(\bar{x}_1, \bar{z}_1, u) \\ \bar{z}_1 &\text{ is predicted by PLS using } \bar{x}_1 \text{ as input} \\ y &= \hat{h}(\bar{x}_1) \end{aligned} \tag{20}$$

Note that $\bar{z}_1$ only depends on $\bar{x}_1$ and does not depend on the input *u* as the original system was a regular DAE system. Therefore, the input *u* does not need to be included in the independent matrix when computing the PLS model for DAE systems.

2) Implementation of PLS

The procedure to implement PLS for model reduction of the algebraic equations of a DAE system is similar to the one for ODE systems. Only minor modifications need to be performed for DAE systems for obtaining the training data set.

The state matrix *A* and input matrix *B* can be collected by simulating the original DAE system. Note that the matrix *A* not only consists of the matrix for the states but also includes the algebraic variables. The matrix *A* can be split into block matrices

$$A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \quad \begin{aligned} A_1&: \text{ data for states} \\ A_2&: \text{ data for algebraic variables} \end{aligned} \tag{21}$$

The transformed data matrices can be computed by:

$$\bar{A}_1 = T_1 A_1 \quad \text{and} \quad \bar{A}_2 = T_2 A_2 \tag{22}$$

If the number of retained states is *k*, then the first *k* rows of $\bar{A}_1$ are the data matrix of $\bar{x}_1$, denoted as $\bar{X}_1$. Similarly, if the number of retained algebraic variables is *m*, then the first *m* rows of $\bar{A}_2$ correspond to the data matrix of $\bar{z}_1$, denoted as $\bar{Z}_1$. $\bar{X}_1$ and $\bar{Z}_1$ are in fact the independent matrix *X* and dependent matrix *Y* before scaling is applied. The final reduced system is given by (23) where the regression matrix *R* is computed via PLS.

$$\begin{aligned} \dot{\bar{x}}_1 &= \hat{f}(\bar{x}_1, \bar{z}_1, u) \\ \bar{x}_2 &= \bar{x}_{2,ss} \\ \bar{z}_1 &= R\bar{x}_1 \\ \bar{z}_2 &= \bar{z}_{2,ss} \\ y &= \hat{h}(\bar{x}_1) \end{aligned} \qquad \text{where, } \bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix}, \ \bar{z} = \begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} \tag{23}$$

IV. CASE STUDIES

Two examples are presented in this section to illustrate the use of PLS for nonlinear model reduction. One example is a catalytic fixed-bed reactor, described by 120 ODEs while the other example is a distillation column, consisting of 32 differential equations and 32 implicit algebraic equations.

A. Example 1: Catalytic Fixed-Bed Reactor

The reactor system is a multi-tubular reactor with a highly exothermic reaction that synthesizes phthalic anhydride from *o*-xylene [10]. Two partial differential equations are used to describe the mass and energy balances along the length of the reactor. A finite difference method is applied to discretize the partial differential equations into a set of ODEs consisting of 120 states, 60 of which describe the concentrations and 60 represent the temperatures at the discretization points. The steady state concentration and temperature distribution along the reactor length is shown in Fig. 1. It can be observed that there is a "hot spot" in the reactor, which is commonly found in exothermic processes. As the temperature at this hot spot is of primary interest, a measurement is located at the position where the hot spot occurs. The inlet temperature serves as the input variable.

1) Data Set Collection

The reduced reactor system is described by (14) and (15). It has been determined from the magnitude of the singular values that 25 states are sufficient for a good approximation of the original systems, i.e., $k = 25$. Therefore, the reduced system consists of 25 differential equations. Ten additional states are approximated via PLS, i.e., $m = 10$, to obtain a good approximation at steady state. The other 85 states are truncated, i.e., kept as constants. Due to the high degree of nonlinearity for the reactor model, only perturbations of the input up to ±5% are acceptable for the stability of the operating point. The training data set is collected and preprocessed as described in Section III-A (2).

2) PLS Implementation

After the PLS model is obtained by use of the training data set, this model is evaluated on test data which is generated by exciting the system with a series of random

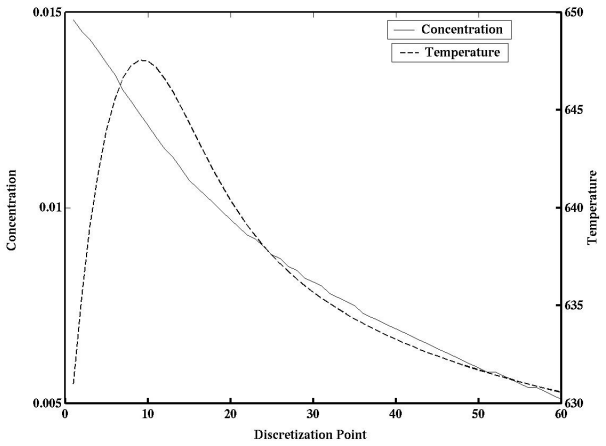

Fig. 1. Steady state of reactor

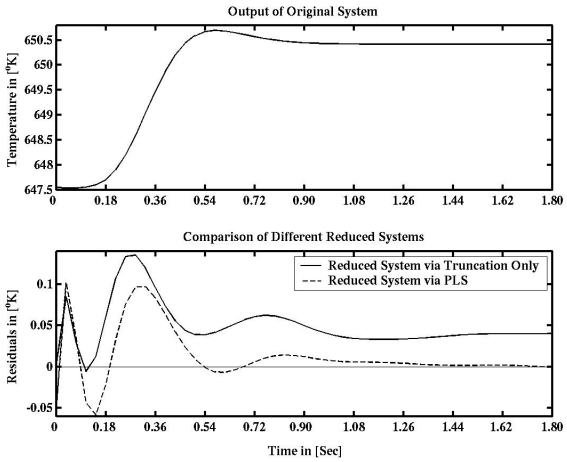

Fig. 3. Comparison of reduced reactor systems' step responses

input perturbations. The prediction error of the PLS model, i.e., the error between the original value of one dependent variable and the corresponding predicted value, is shown in Fig. 2. Note that this dependent variable corresponds to a variable of the transformed system. The upper graph in Fig. 2 is the absolute value and the lower graph shows the prediction error. Based on the small prediction error, it can be concluded that the PLS model provides a good fit and can be used for model reduction of the reactor system.

3) Performance of Reduced System

To show the performance of the reduced system via the presented technique, a comparison between the reduced system derived from truncation (model 1) and a reduced system resulting from residualization via PLS (model 2) is given in Fig. 3. In this figure, the step responses of these reduced systems and the original system with an input perturbation of 3% are compared. The graph in the upper subplot depicts the outputs of the original system and the lower graphs represent the residuals between this output and the outputs of the reduced systems. It can be seen that the performance of model 2 is much better than that of model 1, especially with regard to the steady state behavior. These results illustrate that while PLS is a linear system identification technique, it can return excellent results if it is included in a model reduction procedure that combines PLS and nonlinear balancing.

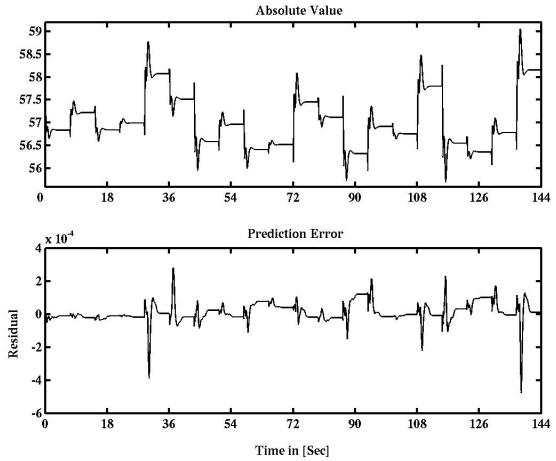

Fig. 2. Prediction performance of PLS model for reactor system

B. Example 2: Distillation Column

Consider a distillation column with 30 trays for the separation of a binary mixture of cyclohexane and heptane. This system contains 32 differential equations [11] which describes the dynamics of concentrations of cyclohexane. The Wilson equation is applied for computation of the vapor-liquid equilibrium, resulting in a model with 32 differential equations and 32 algebraic equations [5]. The reflux ratio serves as the input variable while the concentration of the distillate is the output of the system.

1) Data Set Collection

Based on the singular values it is determined that 3 states are sufficient for approximating the 32 differential equations and 3 algebraic variables are retained, i.e., $k = m = 3$. The data set is collected by exciting the distillation column with a series of input perturbations of up to ±15%.

2) PLS Implementation

Similar to the reactor model, the prediction error is shown in Fig. 4. This error is sufficiently small so that the PLS model can be used to predict the retained algebraic variables from the values of the states of the reduced-order model.

3) Performance of Reduced System

Given a -10% input perturbation, the step responses of three systems are compared: 1) a reduced system where only the differential equations are reduced via balanced truncation (model 1); this method is computationally efficient but will have steady state offset; 2) a reduced system where only the differential equations are reduced via balanced residualization (model 2); the procedure will result in no steady state offset but is computationally not the most efficient; 3) a reduced system where both differential and algebraic equations are reduced, and the retained algebraic variables are approximated by PLS (model 3). A comparison is shown in Fig. 5. The performance for model 3 is

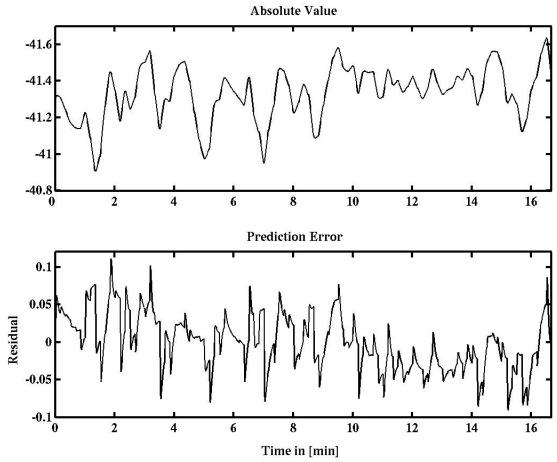

Fig. 4. Prediction performance of PLS model for distillation system

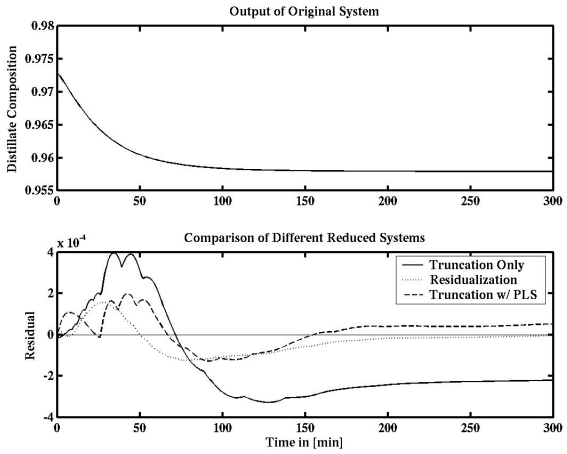

Fig. 5. Comparison of reduced distillation systems' step responses

significantly better than that of model 1 and comparable with model 2. Moreover, a comparison of the computation times is provided in Table 1, where the computational effort of all of the reduced systems is significantly smaller than for the original system. Additionally, model 3 is easier to solve than model 2 while it has a comparable degree of accuracy. It can be concluded that a combination of PLS and balancing performs very well for model reduction of DAE systems and represents a good trade-off between accuracy and the computational effort required for its solution.

V. CONCLUSIONS

This paper presents a new technique for nonlinear model reduction by using a combination of nonlinear balancing and partial least squares (PLS). The reduction procedure can be applied for model reduction of nonlinear ODE and DAE systems. The technique has been illustrated in two case studies, where the reduced-order models were significantly smaller and faster to solve than the original model, while they provide an excellent approximation to the input-output behavior of the original system.

ACKNOWLEDGMENT

Chuili Sun thanks Dr. Uwe Kruger for his helpful comments on PLS.

TABLE I
COMPARISON OF COMPUTATION TIME

Model	1	2	3	4
Time (s)	0.13	0.23	0.15	0.43

Model 1: Reduced system via balanced truncation
Model 2: Reduced system via balanced residualization
Model 3: Reduced system via balancing and PLS
Model 4: Original System

REFERENCES

[1] S. J. Qin, T. A. Badgwell, "A survey of industrial model predictive control technology", *Control Eng. Practice*, Vol. 11, No. 7, 2003, pp. 733-764.

[2] J. Hahn, T.F. Edgar, "An improved method for nonlinear model reduction using balancing of empirical gramians*", Computers and Chemical Engineering*, Vol. 26, 2002, pp. 1379-1397.

[3] S. Lall, J.E. Marsden, S. Glavaski, Empirical model reduction of controlled nonlinear systems*, in 14th IFAC World Congress*, Beijing, 1999.

[4] P.D. Christofides, P. Daoutidis, Finite-dimensional control of parabolic PDE systems using approximate inertial manifolds, *J. Math. Anal. Appl.*, Vol. 216, No. 2, 1997, pp. 398-420.

[5] C. Sun, J. Hahn, Reduction of differential-algebraic equation systems via projections and system identification, *Journal of Process Control*, 2005, in press.

[6] H. Wold, "Partial least squares", *in Encyclopedia of Statistical Sciences*, Vol. 6, pp. 581-591, New York: Wiley.

[7] G. López-Cueto a, M. Ostra b, C. Ubide b, "Linear and non-linear multivariate calibration methods for multicomponent kinetic determinations in cases of severe non-linearity", *Analytica Chimica Acta*, Vol. 405, 2000, pp. 285-295.

[8] J. Hahn, T.F. Edgar, W. Marquardt, "Controllability and observability covariance matrices for the analysis and order reduction of stable nonlinear systems", *Journal of Process Control*, Vol. 13, 2003, pp. 115-127.

[9] P. Geladi, B.R. Kowalski, "Partial Least Squares Regression: A Tutorial", *Analytica Chimica Acta*, Vol. 185, 1986, pp. 1-17.

[10] P. J. Van Welsenaere, G. F. Froment, "Parametric sensitivity and runaway in fixed bed catalytic reactors", *Chem. Eng. Sci*., Vol. 25, 1970, pp. 1503-1516.

[11] J. Hahn, T.F. Edgar, "An improved method for nonlinear model reduction using balancing of empirical gramians", *Computers and Chemical Engineering,* Vol. 26, 2002, pp. 1379-1397.