ELSEVIER

Chemical Engineering Science 59 (2004) 2801–2810

Chemical Engineering Science

www.elsevier.com/locate/ces

A new data-based methodology for nonlinear process modeling

Cheng Cheng, Min-Sen Chiu*

Department of Chemical and Biomolecular Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260, Singapore

Received 17 March 2003; received in revised form 24 June 2003; accepted 18 April 2004

Abstract

A new data-based method for nonlinear process modeling is developed in this paper. In the proposed method, both distance measure and angle measure are used to evaluate the similarity between data, which is not exploited in the previous work. In addition, parametric stability constraints are incorporated into the proposed method to address the stability of local models. Furthermore, a new procedure of selecting the relevant data set is proposed. Literature examples are presented to illustrate the modeling capability of the proposed method. The adaptive capability of the proposed method is also evaluated.

Keywords: Process modeling; Just-in-time learning; Distance measure; Angle measure; Stability

1. Introduction

Mathematical models are often required for purposes of process modeling, control, and fault detection and isolation. However, because most chemical processes are multivariate and nonlinear in nature, and their dynamics can be time-varying, first-principle models are often unavailable due to the lack of complete physicochemical knowledge of chemical processes. An alternative approach is to develop data-based methods to build model from process data measured in industrial processes.

Traditional treatments of the data-based modeling methods focus on global approaches, such as neural networks, fuzzy set, and other kinds of nonlinear parametric models (Nelles, 2001). However, when dealing with large sets of data, this approach becomes less attractive because of the difficulties in specifying model structure and the complexity of the associated optimization problem, which is usually highly non-convex. Another fundamental limitation of these methods is that it is difficult for them to be updated online when the process dynamics are moved away from the nominal operating space. On the other hand, the idea of local modeling is to approximate a nonlinear system with a set of relatively simple local models valid in a certain operating regimes. The T–S fuzzy model (Takagi and Sugeno, 1985) and neuro-fuzzy network (Jang and Sun, 1995; Nelles, 2001) are well-known examples of local modeling approach. However, most local modeling approaches suffer from the drawback of requiring a priori knowledge to determine the partition of operating space and when this information is lacking, complicated training strategy needs to be resorted to determine both optimal model structure and parameters of the local models.

To alleviate the aforementioned problems, just-in-time learning (JITL) (Cybenko, 1996) was recently developed as an attractive alternative for modeling the nonlinear systems. It is also known as instance-based learning (Aha et al., 1991), locally weighted model (Atkeson et al., 1997; Rhodes and Morari, 1997), lazy learning (Bontempi et al., 2001), or model-on-demand (Braun et al., 2001; Hur et al., 2003) in the literature. This approach is inspired by ideas from local modeling and database technology. JITL assumes that all available observations are stored in a database, and the models are built dynamically upon query. Compared with the traditional modeling methods, JITL exhibits three main characteristics. First, the model building is postponed until an output for a given query data is requested. Next, the predicted output for the query data is computed by exploiting the stored data in the database. Finally, the constructed answer and any intermediate results are discarded after the predicted output is obtained (Atkeson et al., 1997; Bontempi et al., 2001; Nelles, 2001). Fig. 1 illustrates the difference between the traditional methods and JITL. Standard methods like neural networks and neuro-fuzzy network are typically

* Corresponding author. Tel.: +65+6874-2223, fax: +65+6779-1936.
E-mail address: checms@nus.edu.sg (M.-S. Chiu).

0009-2509/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ces.2004.04.020

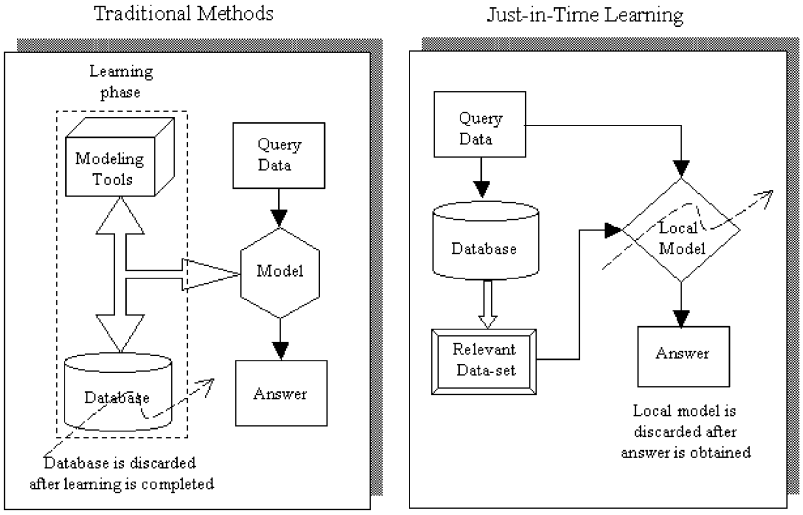

Fig. 1. Comparison between traditional modeling methods and just-in-time learning.

trained offline. Thus, all learning data is processed a priori in a batch-like manner. This can become computationally expensive or even impossible for huge amounts of data, and therefore data reduction techniques may have to be applied. In contrast, JITL has no standard learning phase. It merely gathers the data and stores them in the database and the computation is not performed until a query data arrives. It should be noted that JITL is only locally valid for the operating condition characterized by the current query data. In this sense, JITL constructs local approximation of the dynamic systems. Therefore, a simple model structure can be chosen, e.g. a low-order ARX model. Another advantage of JITL is its inherently adaptive nature, which is achieved by storing the current measured data into the database (Bontempi et al., 2001). In comparison, online adaptation of neural network and neuro-fuzzy models requires model update from scratch, namely both network structure (e.g. the number of hidden neurons in the former case and the number of the fuzzy rules in the latter) and model parameters may need to be changed simultaneously. Evidently, this procedure is not only time-consuming, but also it will interrupt the plant operation, if these models are used for other purposes like model based controller design.

In the previous work, distance measures are overwhelmingly used to evaluate the similarity between two data samples (Atkeson et al., 1997; Rhodes and Morari, 1997; Bontempi, et al., 2001; Braun et al., 2001). Complementary information available from angular relationship has not been exploited. In addition, the stability of local model is not addressed in the previous work, resulting in unstable local models even when the process is stable. In this paper, by incorporating the stability constraints, an enhanced JITL methodology based on both angle measure and distance measure is proposed. In addition, a new procedure of selecting the relevant data set is proposed. Two literature examples are used to illustrate the modeling capability of the proposed method in nonlinear process modeling. This paper is organized as follows: in the next section, the conventional JITL methodology is briefly described, followed by the development and discussion of the proposed method. Section 4 is devoted to the case studies of modeling the van de Vusse reactor with output multiplicity characteristic as well as a multivariable process. Finally, the conclusions will be drawn.

2. Just-in-time learning

There are three main steps in JITL to predict the model output corresponding to the query data: (1) the relevant data samples in the database are searched to match the query data by some nearest neighborhood criterion; (2) a local model is built based on the relevant data; (3) model output is calculated based on the local model and the current query data. The local model is then discarded right after the answer is obtained. When the next query data comes, a new local model will be built based on the aforementioned procedure.

To facilitate the ensuing developments, the JITL algorithm is described next. Suppose that a database consisting of N process data $(y_i, \mathbf{x}_i)_{i=1-N}$, $y_i \in R$, $\mathbf{x}_i \in R^n$, is collected. It is worth pointing out that the vector $\mathbf{x}_i$ is formed by the past values of both process input and process output in modeling a dynamic system, which will become evident in the following discussion. Given a specific query data $\mathbf{x}_q \in R^n$ whose elements are identical to those defined for $\mathbf{x}_i$, the objective of JITL is to predict the model output $\hat{y}_q = f(\mathbf{x}_q)$ according to the known database $(y_i, \mathbf{x}_i)_{i=1-N}$. In the literature, distance measure $d(\mathbf{x}_q, \mathbf{x}_i)$, e.g. Euclidean norm $d(\mathbf{x}_q, \mathbf{x}_i) = \|\mathbf{x}_q - \mathbf{x}_i\|_2$, is commonly used to select the relevant data set from the database by evaluating the relevance (or similarity) between the query data $\mathbf{x}_q$ and $\mathbf{x}_i$ in the entire database, i.e. smaller value of distance measure

indicates greater similarity between $\mathbf{x}_q$ and $\mathbf{x}_i$. In doing so, a weight w_i is assigned to each data $\mathbf{x}_i$ and it is calculated by the kernel function, $w_i = \sqrt{K(d(\mathbf{x}_q, \mathbf{x}_i)/h)}$, where h is the bandwidth of the kernel function K that normally uses a Gaussian function, $K(d) = \mathrm{e}^{-d^2}$. If a linear model is employed to calculate the model output $\hat{y}_q$, the query answer is (Atkeson et al., 1997)

$$\hat{y}_q = \mathbf{x}_q^{\mathrm{T}}(P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}\mathbf{v}, \tag{1}$$

where $P = W\Phi$, $\mathbf{v} = W\mathbf{y}$, $W \in R^{N\times N}$ is a weight matrix with diagonal elements w_i, $\Phi \in R^{N\times n}$ is the matrix with every row corresponding to $\mathbf{x}_i^{\mathrm{T}}$, and $\mathbf{y} = [y_1, y_2, \ldots, y_N]^{\mathrm{T}}$.

In JITL, PRESS statistic (Myers, 1990) is used to perform leave-one-out cross validation to assess the generalization capability of the model (Atkeson et al., 1997). For a current query data $\mathbf{x}_q$, the leave-one-out cross validation test determines the optimal values of h, h_{opt}, as follows: for a given h, Eq. (1) is used to compute the predicted outputs as required in the leave-one-out cross validation test and the corresponding validation error is calculated. This procedure repeats for a number of h and h_{opt} is chosen as the one resulting in the smallest validation error. With h_{opt} known, the optimal model prediction $\hat{y}_q$ is then computed using Eq. (1) for the current query data $\mathbf{x}_q$.

As mentioned above, each local model obtained by JITL is only locally valid around the query data, therefore simple model structure can be chosen as local model at each query point. For a dynamic system, ARX model can be chosen as the local model for JITL. The ARX model is given as follows:

$$\hat{y}(k) = \mathbf{z}^{\mathrm{T}}(k-1)\Psi, \tag{2}$$

where $\hat{y}(k)$ is the model output at the kth sampling instant, $\mathbf{z}(k-1)$ is the regression vector, and Ψ is the model parameter vector as given by

$$\mathbf{z}(k-1) = [y(k-1), \ldots, y(k-n_y), u(k-n_d-1), \ldots, u(k-n_d-n_u)]^{\mathrm{T}}, \tag{3}$$

$$\Psi = [\psi_1, \ldots, \psi_{n_y}, \psi_{n_y+1}, \ldots, \psi_{n_y+n_u}]^{\mathrm{T}}, \tag{4}$$

where n_y and n_u are integers related to the model's order, and n_d is the process time delay. By comparing Eqs. (1) and (2), it is evident that the local model parameters obtained by JITL method is computed as $(P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}\mathbf{v}$. Furthermore, the vector $\mathbf{x}_i$ in the database and query data $\mathbf{x}_q$ have the same input and output variables as those defined for $\mathbf{z}(k-1)$. For example, for a first-order model with $n_y = n_u = 1$ and $n_d = 0$, the database $(y_i, \mathbf{x}_i)_{i=1-N}$ is given by $[y(k), y(k-1), u(k-1)]_{k=1-N}$ where $y(k)$ and $u(k)$ are the process output and input data collected at the kth sampling instant in the identification test. Similarly, in the prediction phase, the query data $\mathbf{x}_q$ at the $(k-1)$th sampling instant is arranged in the form of $[y(k-1), u(k-1)]^{\mathrm{T}}$ as the input to the JITL algorithm, by which the predicted output at the next sampling instant $\hat{y}(k)$ can be computed. Finally, referring from Eq. (3), the dimensionality of $\mathbf{x}_i$ and $\mathbf{x}_q$ is equal to $n = n_y + n_u$.

3. Enhanced JITL methodology

In the preceding section, it is evident that the conventional JITL methods only use distance measure to evaluate the similarity between two data samples. However, considering data observations as points in space leads to two types of measures: distance and angle between two vectors. Some researchers have demonstrated the advantage of using additional angle measure in evaluating the similarity/dissimilarity between data in principal component analysis (Raich and Cinar, 1997; Yoon and MacGregor, 2001; Singhal and Seborg, 2002). In this paper, we aim to enhance the predictive capability of JITL by incorporating the angular relationship in the formulation of JITL. To this end, the following similarity number, s_i, is defined:

$$s_i = \gamma\sqrt{\mathrm{e}^{-d^2(\mathbf{x}_q,\mathbf{x}_i)}} + (1-\gamma)\cos(\theta_i), \quad \text{if } \cos(\theta_i) \geqslant 0, \tag{5}$$

where γ is a weight parameter and is constrained between 0 and 1, and θ_i is the angle between $\Delta\mathbf{x}_q$ and $\Delta\mathbf{x}_i$, where $\Delta\mathbf{x}_i = \mathbf{x}_i - \mathbf{x}_{i-1}$ and $\Delta\mathbf{x}_q = \mathbf{x}_q - \mathbf{x}_{q-1}$. The value of s_i is bounded between 0 and 1 and when s_i approaches to 1, $\mathbf{x}_i$ resembles closely to $\mathbf{x}_q$.

It is important to note that Eq. (5) will not be used to compute the similarity number s_i between $\mathbf{x}_q$ and $\mathbf{x}_i$ if $\cos\theta_i$ is negative. For simplicity, this point is illustrated in the two-dimensional space as shown in Fig. 2, where $\Delta\mathbf{x}_q^{\perp}$ denotes the vector perpendicular to $\Delta\mathbf{x}_q$. It is clear that a vector lies to the right of $\Delta\mathbf{x}_q^{\perp}$ (say $\Delta\mathbf{x}_1$) is more similar to $\Delta\mathbf{x}_q$ than a vector to the left of $\Delta\mathbf{x}_q^{\perp}$ (say $\Delta\mathbf{x}_2$). The use of cosine function to discriminate the directionality between $\Delta\mathbf{x}_q$ and $\Delta\mathbf{x}_i$ indicates that these two vectors are dissimilar if the cosine function is negative (e.g. $\cos(\pi - \theta_2)$), and hence the corresponding $\mathbf{x}_i$ in the database will be discarded and not involved in the subsequent JITL procedure. On the other hand, a positive cosine function, e.g. $\cos\theta_1$, requires the subsequent calculation of the proposed similarity number s_i in order to further discriminate the similarity between $\mathbf{x}_q$ and $\mathbf{x}_i$.

Another shortcoming of the conventional methods is that all the data $\mathbf{x}_i$ in the database are employed in the regression, as shown in Eq. (1). This may lead to a large sparse regression matrix $P \in R^{N\times n}$ that is prone to numerical problems. To circumvent this problem, we propose that only a pre-specified number of relevant data with greater resemblance to the query data $\mathbf{x}_q$, as determined by the similarity number defined in Eq. (5), are used in the regression. Specifically, two parameters $k_{\min}$ and $k_{\max}$ are chosen such that only the relevant data sets formed by the $k_{\min}$th relevant data to the $k_{\max}$th relevant data are used in the regression. Because $k_{\min}$ and $k_{\max}$ are much smaller than the number of data in the entire database, i.e. N, the computational burden

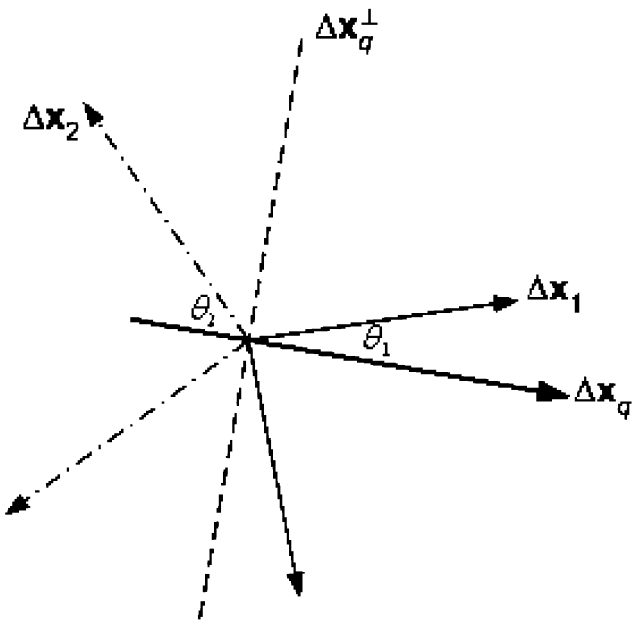

Fig. 2. Illustration of angle measure.

is significantly reduced compared to the conventional JITL methods.

Lastly, it is noted that the stability of local model is not taken into account in the conventional JITL methods. As a result, some local models generated by JITL may be unstable even when the database employed and the query data are obtained from the stable process. This feature is not desirable, especially when these models are to be employed in the controller design. Thus, for a stable system, the parameters of each ARX model obtained by JITL need to be verified whether the corresponding stability constraint is met or not. In case the parameters fail to satisfy the stability constraint, constrained optimization is employed to obtain a stable model. Similar procedure can be devised to obtain an unstable local model when the process of interest is unstable. The parametric stability constraints imposed on Ψ will be discussed in the ensuing development.

The detailed algorithm of the proposed JITL methodology is described as follows. Given a database $(y_i, \mathbf{x}_i)_{i=1-N}$, the parameters $k_{\min}$, $k_{\max}$, and weight parameter γ, and a query data $\mathbf{x}_q$:

Step 1: Compute the distance and angle between $\mathbf{x}_q$ and each data $(y_i, \mathbf{x}_i)$

$$d_i = \|\mathbf{x}_q - \mathbf{x}_i\|_2, \quad i = 1\text{–}N, \tag{6}$$

$$\cos(\theta_i) = \frac{\Delta\mathbf{x}_q^{\mathrm{T}}\Delta\mathbf{x}_i}{\|\Delta\mathbf{x}_q\|_2 \cdot \|\Delta\mathbf{x}_i\|_2}, \quad i = 1\text{–}N. \tag{7}$$

If $\cos(\theta_i) \geqslant 0$, compute the similarity number s_i

$$s_i = \gamma\sqrt{\mathrm{e}^{-d_i^2}} + (1-\gamma)\cos(\theta_i) \tag{8}$$

If $\cos(\theta_i) < 0$, the data $(y_i, \mathbf{x}_i)$ is discarded.

Step 2: Arrange all s_i in the descending order. For $l = k_{\min}$ to $k_{\max}$, the relevant data set $(\mathbf{y}_l, \Phi_l)$, where $\mathbf{y}_l \in R^{l\times 1}$ and $\Phi_l \in R^{l\times n}$, are constructed by selecting l most relevant data $(y_i, \mathbf{x}_i)$ corresponding to the largest s_i to the lth largest s_i. Denote $W_l \in R^{l\times l}$ a diagonal weight matrix with diagonal elements being the first l largest values of s_i, and calculate

$$P_l = W_l\Phi_l, \tag{9}$$

$$\mathbf{v}_l = W_l\mathbf{y}_l. \tag{10}$$

The local model parameters are then computed by

$$\Psi_l = (P_l^{\mathrm{T}}P_l)^{-1}P_l^{\mathrm{T}}\mathbf{v}_l, \tag{11}$$

where $(P_l^{\mathrm{T}}P_l)^{-1}$ is calculated by SVD method. Next, the leave-one-out cross validation test is conducted and the validation error is calculated by (Myers, 1990)

$$e_l = \frac{1}{\sum_{j=1}^{l} s_j^2}\sum_{j=1}^{l}\left(s_j\frac{y_j - \phi_j^{\mathrm{T}}(P_l^{\mathrm{T}}P_l)^{-1}P_l^{\mathrm{T}}\mathbf{v}_l}{1 - \mathbf{p}_j^{\mathrm{T}}(P_l^{\mathrm{T}}P_l)^{-1}\mathbf{p}_j}\right)^2, \tag{12}$$

where y_j is the jth element of $\mathbf{y}_l$, ϕ_j^{T} and $\mathbf{p}_j^{\mathrm{T}}$ are the jth row vector of Φ_l and P_l, respectively.

Step 3: According to the validation errors, the optimal l is determined by

$$l_{\mathrm{opt}} = \arg\min_{l}(e_l) \tag{13}$$

Step 4: Verify the stability of local model built by the optimal model parameters $\Psi_{l_{\mathrm{opt}}}$. Because JITL constructs the local approximation of the dynamic systems, only the stability constraints of first- and second-order models are given as follows:

First-order model:

$$-1 < \psi_1 < 1. \tag{14}$$

Second-order model:

$$\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix} < \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \tag{15}$$

$$-1 < \psi_2 < 1. \tag{16}$$

If $\Psi_{l_{\mathrm{opt}}}$ satisfies the stability constraint, the predicted output for query data is computed by

$$(\hat{y}_q)_{l_{\mathrm{opt}}} = \mathbf{x}_q^{\mathrm{T}}\Psi_{l_{\mathrm{opt}}}. \tag{17}$$

Otherwise, $\Psi_{l_{\mathrm{opt}}}$ is used as the initial value in the following optimization problem subject to the appropriate stability constraint, Eq. (14) or Eqs. (15) and (16).

$$\min_{\Psi}\|P_{l_{\mathrm{opt}}}\Psi - \mathbf{v}_{l_{\mathrm{opt}}}\|_2. \tag{18}$$

With the optimal solution $\Psi^*_{l_{\mathrm{opt}}}$ obtained from Eq. (18), the predicted output for query data is then calculated as $\mathbf{x}_q^{\mathrm{T}}\Psi^*_{l_{\mathrm{opt}}}$.

Step 5: When the next query data comes, go to step 1.

One remark about the proposed method is the determination of γ. Typically, the prediction accuracy of the proposed method improves initially when γ decreases from one to a smaller value of γ, after which the prediction accuracy degrades. Owing to the lack of the systematic guideline of determining the optimal value of γ, the following procedure is adopted in this paper. The proposed method is applied to the validation data for a number of γ and the corresponding validation error is calculated. The optimal γ is then chosen as the one resulting in the smallest validation error.

Although the aforementioned data-based modeling methodology is developed for single-input single-out systems, it carries straightforwardly over to the multivariable systems. This is because the modeling of a multivariable system with m outputs can be treated as m multiple-input single-output problems. A chemical reactor with two inputs and three outputs will be presented in the next section to demonstrate the application of the proposed method for nonlinear modeling of multivariable systems.

4. Examples

Example 1. Considering the van de Vusse reactor with the following reaction kinetic scheme: $A \rightarrow B \rightarrow C$, $A \rightarrow D$, which is carried out in an isothermal CSTR. The dynamics of the reactor are described by the following equations (Doyle et al., 1995):

$$\frac{dC_A}{dt} = -k_1 C_A - k_3 C_A^2 + \frac{F}{V}(C_{Af} - C_A), \tag{19}$$

$$\frac{dC_B}{dt} = k_1 C_A - k_2 C_B - \frac{F}{V} C_B, \tag{20}$$

where the parameters used are: $k_1 = 50\ \mathrm{h}^{-1}$, $k_2 = 100\ \mathrm{h}^{-1}$, $k_3 = 10$ l/(mol h), $C_{Af} = 10$ mol/l, and $V = 1$ l. The nominal operation condition is $C_A = 3.0$ mol/l, $C_B = 1.12$ mol/l, and $F = 34.3$ l/h. The concentration of component B, C_B, is the process output and the flow rate, F, is the process input.

A salient feature of the above reactor is that the sign of its steady-state gain may change as the operation condition changes (see Fig. 3). To apply the proposed method, a second-order ARX model is employed as the local model, i.e. the regression vector in Eq. (3) is chosen as $\mathbf{z}(k-1) = [C_B(k-1), C_B(k-2), F(k-1)]^{\mathrm{T}}$, and set $k_{\min} = 6$ and $k_{\max} = 60$. The database is generated by introducing uniformly random steps with distribution of $[10, 150]$ and the switching probability of 0.1 at every sampling time to the process input F. Two thousand input/output data as shown in Fig. 4 are collected to build the database $[C_B(k), \mathbf{z}(k-1)]_{k=1-2000}$. Because this system is stable in the operating space under consideration, the local model needs to satisfy the stability constraints given in Eqs. (15) and (16).

To determine the optimal value of the weight parameter γ, Table 1 lists the mean-squared-error (MSE) of the validation test for different values of γ. The input signal employed in the validation test is shown in Fig. 5. As can be seen from Table 1, the error decreases initially as γ decreases from 0.98 to 0.95, after which the error starts to increase. Therefore, the optimal γ is chosen to be 0.95. Based on the same database, JITL with distance measure alone is also considered for comparison purpose. The predictive performances of these two methods are compared in Table 1 and Fig. 5. It is evident that the proposed method complemented with angle measure and stability constraint outperforms the conventional JITL.

Fig. 6 demonstrates the prediction capability of the proposed method with $\gamma = 0.95$ when F is subject to step change of 15 and -20, respectively. The steady-state errors are 1.12×10^{-3} and 1.52×10^{-4}, respectively. This simulation condition is adopted from the work done by Doyle et al. (1995) who constructed a second-order Volterra model to predict this reactor's dynamics. In their paper, the steady-state prediction errors for positive and negative step changes are estimated to be 0.016 and 0.056, respectively.

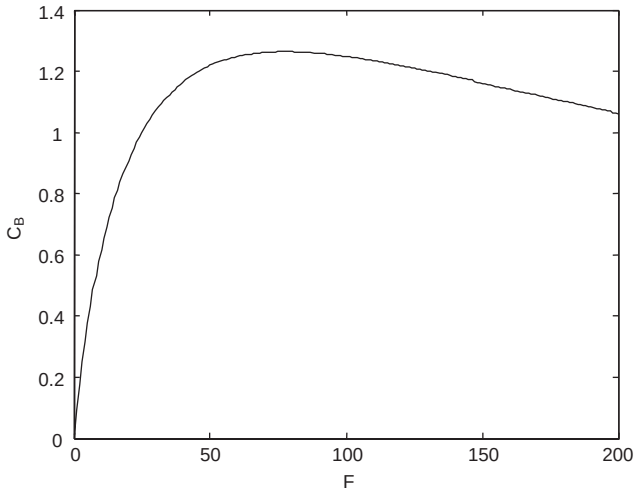

Fig. 3. Steady-state curve of van de Vusse reactor.

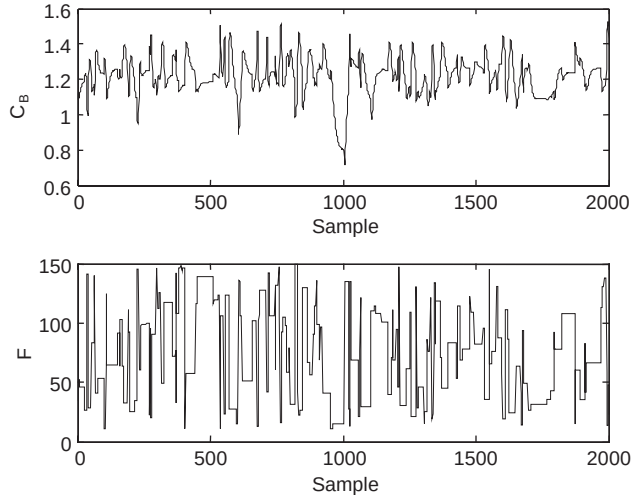

Fig. 4. Input–output data used for constructing the database (van de Vusse reactor).

Table 1
Validation error of the proposed method for various values of γ

Distance measure	$\gamma = 0.98$	$\gamma = 0.95$	$\gamma = 0.90$	$\gamma = 0.85$	$\gamma = 0.80$
7.72×10^{-4}	8.11×10^{-5}	7.80×10^{-5}	7.82×10^{-5}	8.03×10^{-5}	9.45×10^{-5}

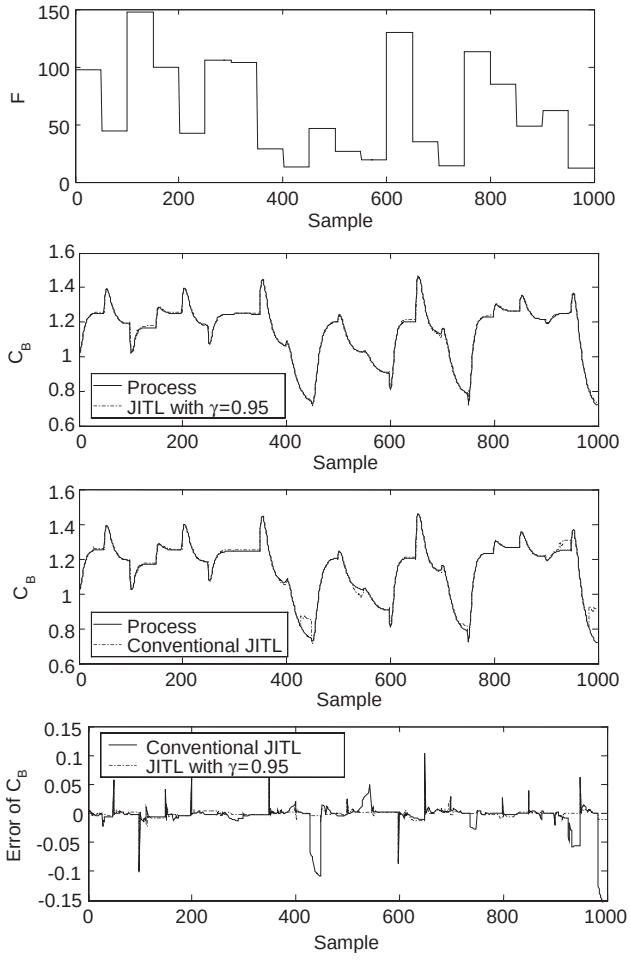

Fig. 5. Validation result of C_B.

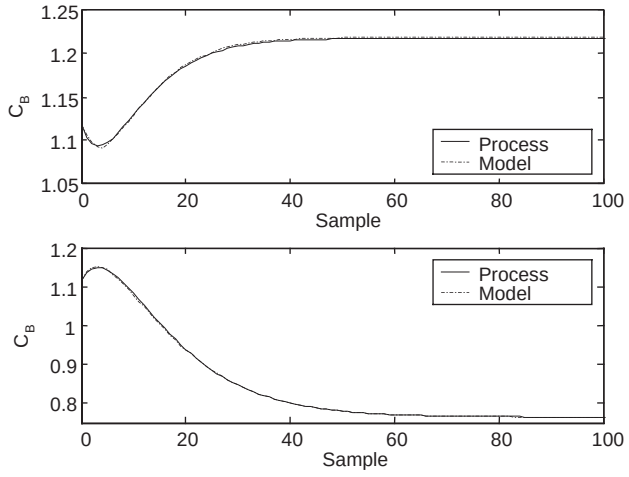

Fig. 6. Response for step change from 34.3 to 49.3 (top) and 14.3 (bottom) in F.

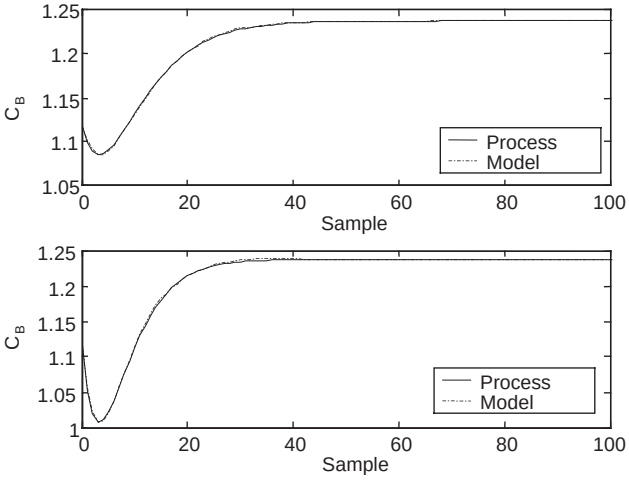

Fig. 7. Response for step change from 34.3 to 55 (top) and 109 (bottom) in F.

Clearly, the proposed method gives more accurate prediction than the Volterra model. To illustrate the capability of the proposed method to model reactor's dynamics when the value of input F is changed from one side of the extreme point to the opposite side (see Fig. 3), the step changes of F from 34.3 to 55 and 109 are conducted. Note that these two final values of F correspond to the identical steady-state value of C_B. As illustrated in Fig. 7, the proposed method can predict the actual process very closely, as also evidenced by very small steady-state errors of 6.66×10^{-5} (top curve) and 1.18×10^{-5} (bottom curve).

To test the robustness of the proposed method, both process output and input variables are corrupted by 2% Guassian white noise. Despite that both database and validation data contain the corrupted signals, the proposed method maintains good prediction accuracy in the presence of process noise, as illustrated in Fig. 8.

Example 2. The application of the proposed method in modeling multivariable systems is illustrated by considering a nonisothermal CSTR with first-order reaction, which can be described by the following equations (You and Nikolaou, 1993):

$$A\frac{dL}{dt} = F_i - K\sqrt{L}, \tag{21}$$

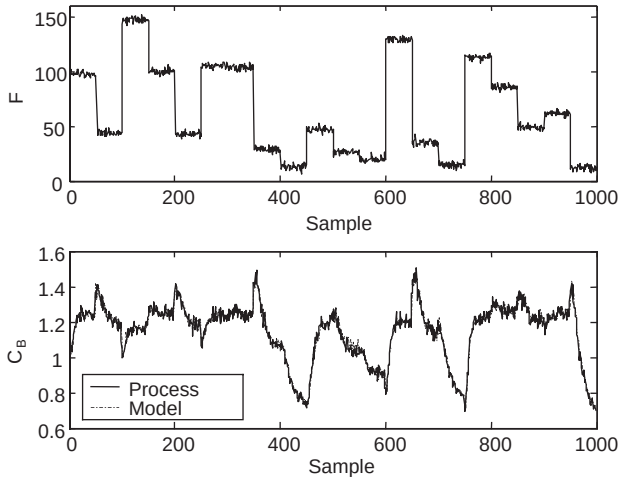

Fig. 8. Validation result (with noisy process data).

Table 2
Parameters and nominal values of CSTR example

Variable	Value
A	1.000 m^2
K	0.9715 $m^{5/2}/h$
k_0	7.08×10^7 1/h
E/R	8375 K
ΔH_R	−69 755 J/mol
c_p	3140 J/kg K
ρ	800.8 kg/m^3
Q	1.055×10^8 J/h
v	0
V	1.360 m^3
T_i	373.3 K
C_A	393.3 mol/m^3
T	547.5 K
L	1.360 m
F_i	1.133 m^3/h
C_{Ai}	8008 mol/m^3

$$\frac{dC_A}{dt} = \frac{F_i}{V}(C_{Ai} - C_A) - k_0 e^{-E/RT} C_A, \tag{22}$$

$$\frac{dT}{dt} = \frac{F_i}{V}(T_i - T) + \frac{-\Delta H_R}{c_p \rho} k_0 e^{-E/RT} C_A - \frac{Q}{c_P \rho V}(1 + v). \tag{23}$$

The parameters used in the simulation and the nominal operating condition are summarized in Table 2. In this example, the variables F_i and C_{Ai} are process inputs, wheras L, C_A, and T are process outputs.

To proceed with the proposed method, the following regression vectors are chosen:

$$L\text{: } \mathbf{z}_1(k-1) = [L(k-1),\ L(k-2),\ F_i(k-1)], \tag{24}$$

$$C_A\text{: } \mathbf{z}_2(k-1) = [C_A(k-1),\ C_A(k-2),\ F_i(k-1),\ C_{Ai}(k-1)] \tag{25}$$

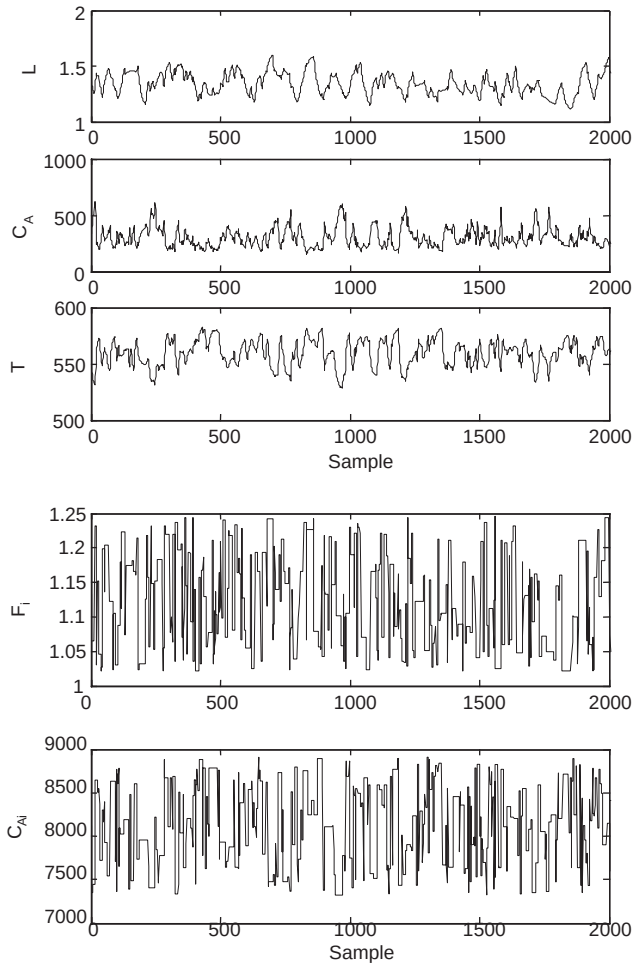

Fig. 9. Input–output data used for constructing the database (CSTR example).

$$T\text{: } \mathbf{z}_3(k-1) = [T(k-1),\ T(k-2),\ F_i(k-1),\ C_{Ai}(k-1)]. \tag{26}$$

To generate the database, random step signals with uniform distribution of [1.02 1.25] and [7207 8808] and the switching probability of 0.15 are added to F_i and C_{Ai}, respectively, as illustrated in Fig. 9. The input/output data given in Fig. 9 are then used to construct three databases: $[L(k), \mathbf{z}_1(k-1)]_{k=1-2000}$, $[C_A(k), \mathbf{z}_2(k-1)]_{k=1-2000}$, and $[(T(k), \mathbf{z}_3(k-1)]_{k=1-2000}$ for predicting L, C_A, and T, respectively.

As a result of the open-loop stable nature of this reactor, the parameters of three local models obtained for each query data need to satisfy the stability constraints given in Eqs. (15) and (16). In addition, $k_{\min} = 6$ and $k_{\max} = 60$ are chosen to predict the output L, whereas $k_{\min} = 8$ and $k_{\max} = 60$ are used for the other two predicted outputs. To determine the optimal value of γ, Table 3 summarizes the mean squared errors of the validation test, where the input

Table 3
Validation error of the proposed method for various values of γ

	Distance measure	$\gamma = 0.98$	$\gamma = 0.95$	$\gamma = 0.90$	$\gamma = 0.85$	$\gamma = 0.75$	$\gamma = 0.70$
L	3.90×10^{-3}	1.18×10^{-5}	1.11×10^{-5}	1.16×10^{-5}	1.17×10^{-5}	1.17×10^{-5}	1.18×10^{-5}
C_A	211.66	93.50	82.02	68.17	67.02	67.91	68.00
T	1.62	0.92	0.86	0.68	0.64	0.53	0.56

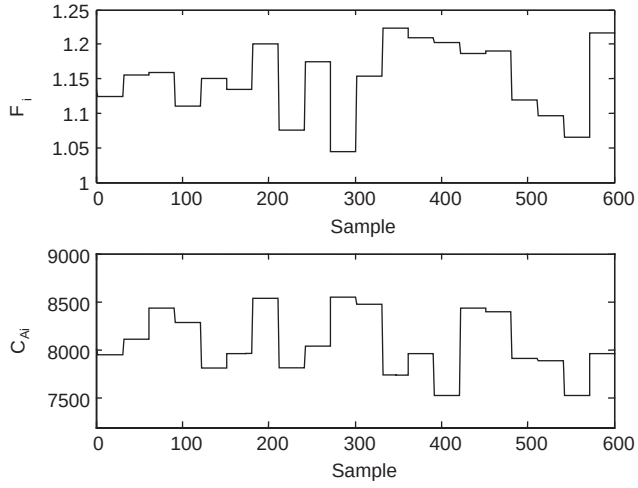

Fig. 10. Input data used in validation test of CSTR example.

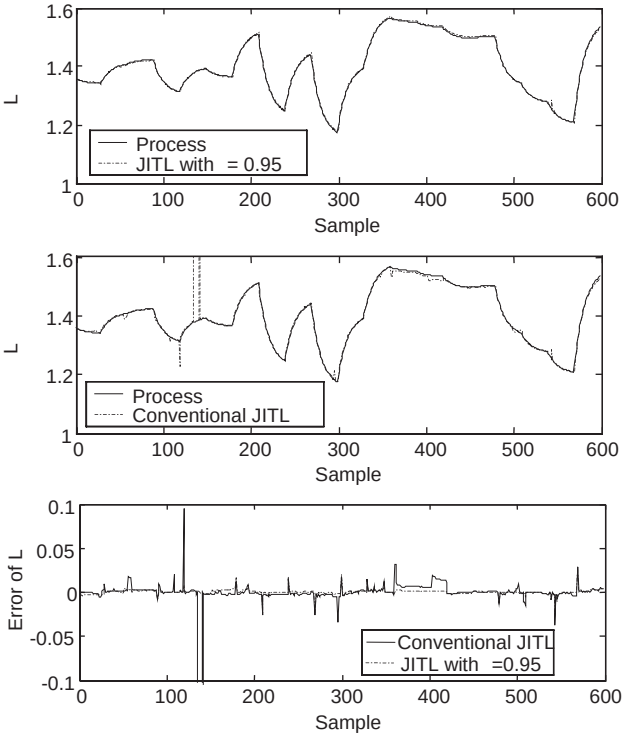

Fig. 11. Validation results of L.

variables employed are shown in Fig. 10, for three process outputs. It is evident that $\gamma = 0.95$, 0.85, and 0.75 are the respective optimal values to predict L, C_A, and T. For comparison purpose, JITL based on the distance measure alone

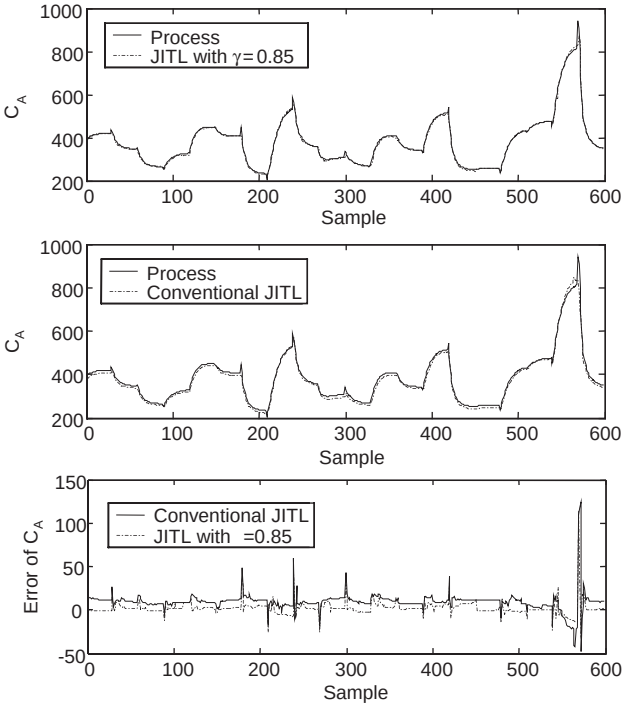

Fig. 12. Validation results of C_A.

and the identical database is again used to model this process. The comparison results given in Table 3 and Figs. 11–13 show that the proposed method has superior prediction accuracy than the conventional JITL. The robustness of the proposed method is evaluated by introducing 1% Guassian white noise to the measured process variables. As illustrated in Fig. 14, the proposed method is insensitive to process noise to some extent.

For traditional data-based modeling methods, it is not a trivial task to update its model online. For example, neural networks need to be retrained to adjust the network parameter according to the new operating condition. In the extreme cases, the network structure needs to be re-determined to achieve better prediction of the new process dynamics. Evidently, this procedure is not desirable from computational point of view. In contrast, JITL is inherently adaptive by simply adding the current on-line process data to the database. For illustration, assuming that the heat transfer Q in Eq. (23) is suddenly changed $\pm 25\%$ from its nominal value due to the effect of unmeasured disturbance, which is equivalent to the change of the parameter v from its nominal

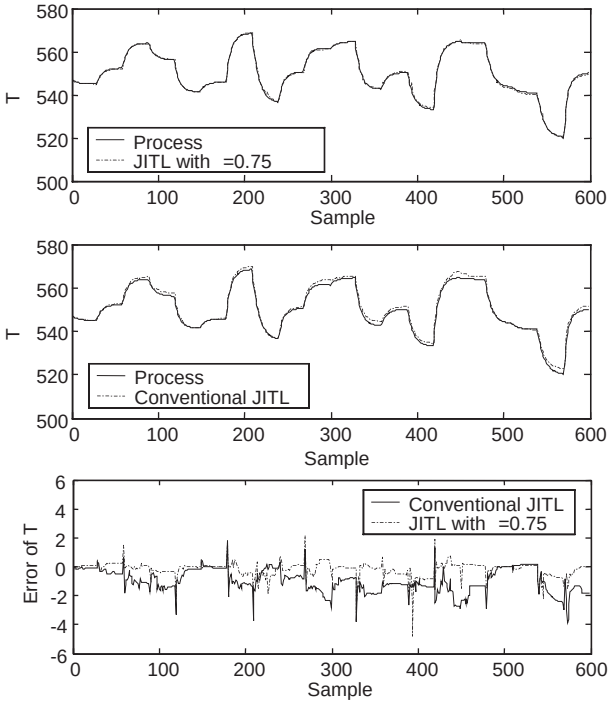

Fig. 13. Validation results of T.

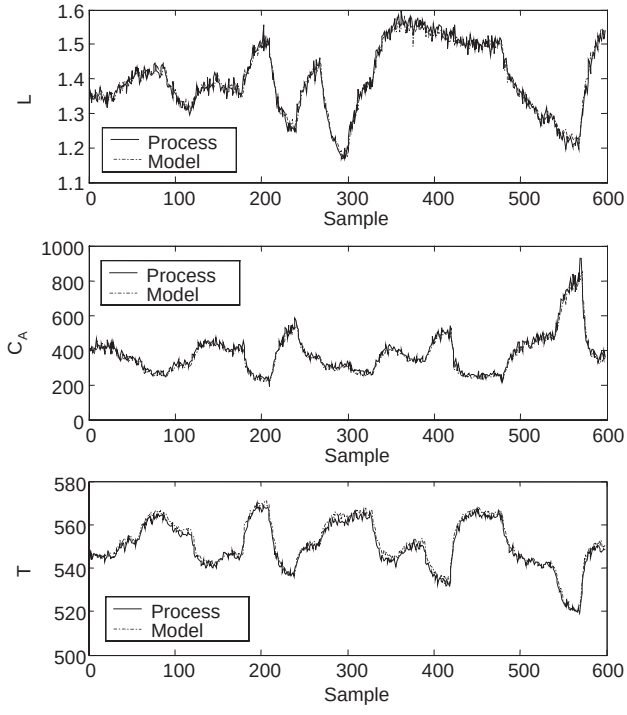

Fig. 14. Validation result (with noisy process data).

value of 0 to 0.25 and −0.25, respectively. Two scenarios are studied: non-adaptive version and adaptive version of the proposed method. In the former, the original databases mentioned previously remain unchanged, whereas in the latter the databases are constantly updated by adding the new available input–output data to the databases at each sampling time. Simulation results in Figs. 15 and 16 show that significantly smaller modeling error is achieved by the adaptive version of the proposed method.

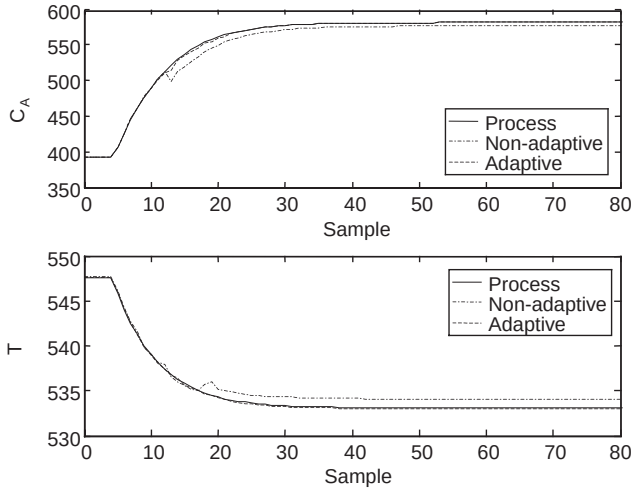

Fig. 15. Response when v varies from 0 to 0.25.

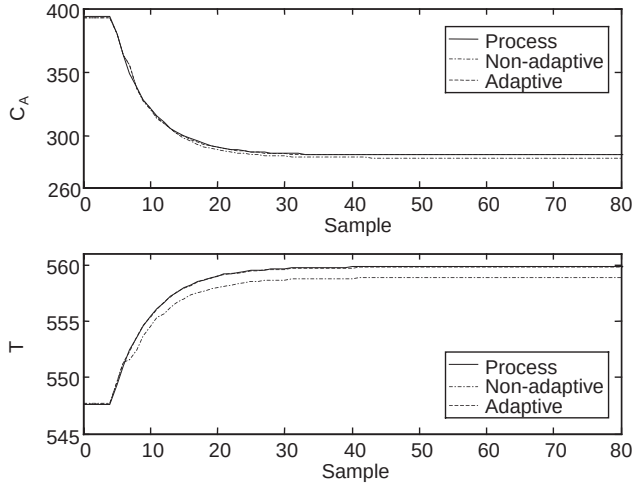

Fig. 16. Response when v varies from 0 to −0.25.

5. Conclusion

In this paper, a data-based methodology for nonlinear process modeling is proposed. The proposed method makes use of both distance measure and angular measure to evaluate the similarity between the query data and data in the database. In addition, a constrained optimization problem is formulated and incorporated into the proposed method to address the stability of local model. Simulation studies illustrate that the proposed method gives marked improvement over its conventional counterparts in nonlinear process

modeling. It is also demonstrated that the proposed method can be made adaptive online readily by simply adding the new process data to the database.

Acknowledgements

The authors are thankful to Professor Yoshihiro Hashimoto for his constructive comments on the revision of this paper.

References

Aha, D.W., Kibler, D., Albert, M.K., 1991. Instance-based learning algorithms. Machine Learning 6, 37–66.

Atkeson, C.G., Moore, A.W., Schaal, S., 1997. Locally weighted learning. Artificial Intelligence Review 11, 11–73.

Bontempi, G., Bersini, H., Birattari, M., 2001. The local paradigm for modeling and control: from neuro-fuzzy to lazy learning. Fuzzy Sets and Systems 121, 59–72.

Braun, M.W., Rivera, D.E., Stenman, A., 2001. A model-on-demand identification methodology for nonlinear process systems. International Journal of Control 74, 1708–1717.

Cybenko, G., 1996. Just-in-time learning and estimation. In: Bittanti, S., Picci, G. (Eds.), Identification, Adaptation, Learning: The Science of Learning Models from Data. Springer, Berlin, pp. 423–434.

Doyle, F.J., Ogunnaike, A., Pearson, R.K., 1995. Nonlinear model-based control using second-order Volterra models. Automatica 31, 697–714.

Hur, S.M., Park, M.J., Rhee, H.K., 2003. Design and application of model-on-demand predictive controller to a semibatch copolymerization reactor. Industrial Engineering and Chemical Research 42, 847–859.

Jang, J.S.R., Sun, C.T., 1995. Neuro-fuzzy modeling and control. Proceedings of the IEEE 83, 378–406.

Myers, R.H., 1990. Classical and Modern Regression with Applications. PWS-Kent Publ., Boston, MA.

Nelles, O., 2001. Nonlinear System Identification. Springer, Berlin.

Raich, A., Cinar, A., 1997. Diagnosis of process disturbances by statistical distance and angel measures. Computers and Chemical Engineering 21, 661–673.

Rhodes, C., Morari, M., 1997. Data-based control trajectory planning for nonlinear systems. Physical Review E 56, 2398–2406.

Singhal, A., Seborg, D.E., 2002. Pattern matching in multivariate time series database using a moving-window approach. Industrial Engineering and Chemical Research 41, 3822–3838.

Takagi, T., Sugeno, M., 1985. Fuzzy identification of systems and its application to modeling and control. IEEE Transactions on Systems, Man, and Cybernetics 15, 116–132.

Yoon, S., MacGregor, F., 2001. Fault diagnosis with multivariate statistical models part I: using steady state fault signatures. Journal of Process Control 11, 387–400.

You, Y., Nikolaou, M., 1993. Dynamic process modeling with recurrent neural networks. A.I.Ch.E. Journal 39, 1654–1667.