Journal of the Taiwan Institute of Chemical Engineers 41 (2010) 446–452

Contents lists available at ScienceDirect

Journal of the Taiwan Institute of Chemical Engineers

journal homepage: www.elsevier.com/locate/jtice

Adaptive IMC controller design using linear multiple models

Yasuki Kansha [a], Li Jia [b], Min-Sen Chiu [a,*]

[a] Department of Chemical and Biomolecular Engineering, National University of Singapore, 4 Engineering Drive 4, Singapore 117576, Singapore
[b] Shanghai Key Laboratory of Power Station Automation Technology, College of Mechatronics Engineering and Automation, Shanghai University, Shanghai 200072, PR China

ARTICLE INFO

Article history:
Received 4 February 2010
Received in revised form 1 April 2010
Accepted 9 April 2010

Keywords:
Just-in-time learning
Adaptive control
Internal model control
Lyapunov method

ABSTRACT

In this paper, an adaptive IMC controller using just-in-time learning (JITL) technique for nonlinear process control is proposed. Based on a set of linear models obtained on-line by the JITL, not only the parameters of IMC controller are updated, but also IMC filter parameter is adjusted on-line by an updating algorithm derived based on the Lyapunov method to guarantee the convergence of JITL's predicted tracking error. Simulation results are presented to demonstrate the advantage of the proposed adaptive IMC design over its conventional counterparts.

1. Introduction

Internal model control (IMC) is a powerful controller design strategy for the open-loop stable dynamic systems (Morari and Zafiriou, 1989). IMC design is expected to perform satisfactorily as long as the process is operated in the vicinity of the point where the linear process model is obtained. However, the performance of IMC controller will degrade or even become unstable when it is applied to nonlinear processes with a range of operating conditions. To extend the IMC design to nonlinear processes, various nonlinear IMC schemes have been proposed in the literature. For instance, Economou *et al.* (1986) provided a nonlinear extension of IMC by employing contraction mapping principle and Newton method. However, this numerical approach to nonlinear IMC design is computationally demanding. Calvet and Arkun (1988) implemented a state-space linearization approach within IMC framework to improve disturbance performance for nonlinear systems. A disadvantage of their approach is that an artificial controlled output is introduced in the controller design procedure and cannot be specified a priori. Another drawback of this method is that the nonlinear controller requires state feedback. Henson and Seborg (1991) proposed a state-space approach and used nonlinear filter to account for the plant/model mismatch. However, these IMC control strategies relied on the availability of a first-principle model, which is often unavailable or too time-consuming and costly to obtain due to the lack of complete physicochemical knowledge of chemical processes. An alternative approach is to develop black-box models from process data collected from the industrial processes. To this end, the ability of artificial neural networks to model almost any nonlinear function without a priori knowledge has led to the investigation of nonlinear IMC schemes using neural networks (NN). In the methods proposed by Bhat and Mcavoy (1990) and Hunt and Sbarbaro (1991), one NN was trained to represent the nonlinear dynamics of process, which was then used as the IMC model, while another NN was trained to learn the inverse dynamics of the process and was employed as the nonlinear IMC controller. Because IMC model and controller were built by the separate neural networks, the controller might not invert the steady-state gain of the model and thus steady-state offset might not be eliminated (Nahas *et al.*, 1992). Moreover, these control schemes do not provide a tuning parameter that can be adjusted to account for the plant/model mismatch. To alleviate those drawbacks, Nahas *et al.* (1992) developed a nonlinear IMC strategy consisting of a model-inverse controller obtained from a neural network and a filter with a single tuning parameter.

However, the above nonlinear IMC designs sacrifice the simplicity associated with linear IMC in order to achieve improved performance. This is mainly due to the use of computationally demanding analytical or numerical methods and neural networks to obtain the inverse of process dynamics. To overcome these difficulties, Doyle *et al.* (1995) proposed a partitioned model-based IMC design based on the Volterra model that retains the original spirit and characteristics of conventional IMC while extending its capabilities to nonlinear systems. However, Volterra model derived using local expansion results such as Carleman linearization is accurate for capturing local nonlinearities around an

* Corresponding author. Tel.: +65 6516 2223; fax: +65 6779 1936.
E-mail address: checms@nus.edu.sg (M.-S. Chiu).

1876-1070/$ – see front matter
doi:10.1016/j.jtice.2010.04.002

Nomenclature

$C_I, C_{I_{in}}$	initiator concentration
$C_m, C_{m_{in}}$	monomer concentration
D_0, D_1	dead polymers in polymerization reaction model
e	error between set-point and output
e_r	error between set-point and predicted output
F, F_I	flow rate
f	low-pass filter
f^*	parameter of polymerization reaction
G	process
$\tilde{G}$	process model
$k_{\min}$, $k_{\max}$	number of minimum and maximum relevant data set
$k_I, k_{f_m}, k_P, k_{T_c}, k_{T_d}$	kinetic parameters
l	number of nearest neighbors
l_{dec}, l_{inc}	factors for adjusting learning rate
M_m	molecular weight of monomer
P_{l^*}	matrix of relevant data
Q	IMC controller
r	set-point
s_i	similarity number
u	process input
V	reactor volume
$V(\cdot)$	Lyapunov function
W_{l^*}	weight matrix of JITL
$\mathbf{x}_i$, $\mathbf{x}_q$	database and query vector of JITL
y_i	process output
$\tilde{y}_i$	predicted process output
y_l, y_{l^*}	relevant process output of JITL

Greek symbols

$\alpha_1^k, \alpha_2^k, \beta_1^k$	coefficients of ARX model
γ	positive constant of Lyapunov function
Δt	sampling time
ζ	update parameter
η	learning rate
κ	weight parameter of JITL
λ	IMC filter time constant
ν	state value
θ_i	angle measure

operating point, but may be erroneous in describing global nonlinear behavior (Maner *et al.*, 1996). Harris and Palazoglu (1998) proposed an alternative partitioned model-based IMC scheme based on the functional expansion models instead of Volterra model. However, functional expansion models are limited to fading memory systems and consequently, the resulting controller gives satisfactory performance only for a limited range of operation. Shaw *et al.* (1997) used recurrent neural network within the partitioned model-based IMC scheme as an alternative for NN-based control application. Maksumov *et al.* (2002) investigated a similar control scheme consisting of a linear ARX model and a NN model. However, one fundamental limitation of these types of global approaches for modeling is that it is difficult for them to be updated on-line when the process dynamics are moved away from the nominal operating space. In this situation, on-line adaptation of these models requires model update from scratch, namely both network structure (e.g. the number of hidden neurons) and network parameters may need to be changed simultaneously. Evidently, this process is not only time-consuming but also it will interrupt the plant operation, if these models are incorporated into model-based controller design.

To circumvent the aforementioned drawback, an adaptive IMC design based on the just-in-time learning (JITL) technique was proposed by Cheng and Chiu (2007). The JITL is considered not only because its prediction capability for nonlinear processes but also the low-order model employed in the JITL, which enables the construction of model-inverse in a straightforward manner. However, the adaptation algorithm developed for the IMC filter did not address the convergence of the tracking error. This motivates our current work to develop an updating algorithm for the IMC filter parameter based on the Lyapunov method to guarantee the convergence of predicted tracking error. Simulation results are presented to illustrate the proposed control strategy and a comparison with its conventional counterparts is made.

2. JITL-based adaptive IMC design

Aha *et al.* (1991) developed instance-based learning algorithms for modeling nonlinear systems. This approach is inspired by the ideas from local modeling and machine learning techniques. Different variants of instance-based learning are also developed in the literature, e.g. locally weighted learning (Atkeson *et al.*, 1997) and just-in-time learning (JITL) techniques (Bontempi *et al.*, 2001). JITL has no standard learning phase because it merely stores the data in the database and the computation is not performed until a query data arrival. Furthermore, JITL constructs local approximation of the dynamic systems characterized by the current query data, and thus low-order model is usually employed in the JITL technique. To achieve better predictive performance of the JITL algorithm, Cheng and Chiu (2004) developed an enhanced JITL algorithm using a similarity measure by combining the commonly used distance metric with the complementary angular metric, which will be used in this work and is briefly introduced in Appendix A.

The block diagram of the IMC structure is shown in Fig. 1, where G and $\tilde{G}$ denote the open-loop stable process and process model, respectively. The IMC controller, Q, can be designed by the following equation (Morari and Zafiriou, 1989):

$$Q = \tilde{G}_-^{-1} f \tag{1}$$

where $\tilde{G}_-$ is the minimum-phase part of $\tilde{G}$ and f is a low-pass filter, which is designed to make the IMC controller Q realizable and to meet the design trade-off between the performance and robustness requirements. The IMC framework allows the use of a variety of process models, such as first-principle models as well as neural network models. However, the difficulty in the use of these models in the IMC framework arises in the design of IMC controller, which is based on the inverse of minimum-phase part of the model $\tilde{G}$ that necessitates a reliable and efficient method to obtain this inversion (Maksumov *et al.*, 2002). To address this problem, the JITL model is employed in the IMC framework so that the model-inverse can be obtained readily. The proposed adaptive IMC scheme is depicted in

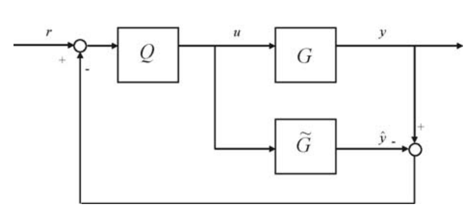

Fig. 1. Block diagram of IMC structure.

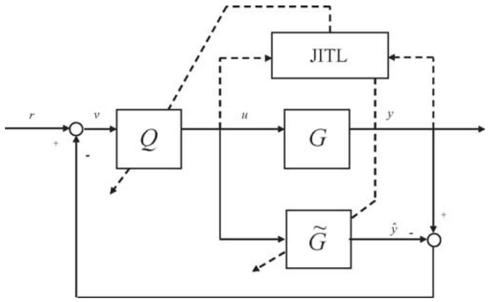

Fig. 2. JITL-based adaptive IMC scheme.

Fig. 2, where the process model $\tilde{G}$ is updated by the JITL algorithm on-line and controller Q is designed based on the inverse of the minimum-phase dynamics of process model $\tilde{G}$ augmented with a low-pass filter. In the proposed method, filter parameter is not fixed, instead it is adjusted on-line by an updating algorithm to be developed in the sequel. As such, the JITL is employed to update the parameters of both IMC model and controller.

As discussed previously, low-order ARX models are usually employed by the JITL. Therefore, without the loss of generality, the following second-order ARX model is considered in the proposed controller design:

$$\hat{y}(k) = \alpha_1^k y(k-1) + \alpha_2^k y(k-2) + \beta_1^k u(k-1) \tag{2}$$

where $\hat{y}(k)$ is the predicted output by the JITL at the kth sampling time, $y(k-1)$ and $u(k-1)$ are the output and manipulated variables at the $(k-1)$th sampling time, α_1^k, α_2^k and β_1^k are the model coefficients at the kth sampling time.

The transfer function model corresponding to Eq. (2) is given by

$$\tilde{G}^k(z^{-1}) = \frac{\beta_1^k z^{-1}}{1-\alpha_1^k z^{-1} - \alpha_2^k z^{-2}} \tag{3}$$

Using a first-order filter, IMC controller is designed as following:

$$\tilde{Q}^k(z^{-1}) = \frac{1-\alpha_1^k z^{-1} - \alpha_2^k z^{-2}}{\beta_1^k} \frac{1-\lambda(k)}{1-\lambda(k)z^{-1}} \tag{4}$$

where $\lambda(k)$ is the IMC filter parameter obtained at the kth sampling instant.

The control law resulting from Eq. (4) is then given by

$$u(k) = \lambda(k)u(k-1) + \frac{1-\lambda(k)}{\beta_1^k}(v(k) - \alpha_1^k v(k-1) - \alpha_2^k v(k-2)) \tag{5}$$

where $v(k) \triangleq r(k) + \hat{y}(k) - y(k)$.

Because IMC filter parameter $\lambda(k)$ is constrained between 0 and 1, the following sigmoid function is employed to map the set [0 1] to $\Re$, which denotes the set of real number:

$$\lambda(k) = \frac{1}{1+e^{-\zeta(k)}} \tag{6}$$

where $\zeta(k) \in \Re$. In the sequel, an updating algorithm will be developed to adjust $\zeta(k)$ on-line, and subsequently the filter parameter $\lambda(k)$ can be easily calculated by Eq. (6).

In order to update the parameter $\zeta(k)$ so that the convergence of the JITL's predicted output to the desired set-point trajectory is guaranteed, Lyapunov function is chosen as follows:

$$V(k) = \gamma e_r^2(k) \tag{7}$$

where $e_r(k)$ is the predicted tracking error defined as $e_r(k) = r(k) - \hat{y}(k)$ and γ is a positive constant.

Define

$$e_r(k+1) = e_r(k) + \Delta e_r(k+1) \tag{8}$$

The increment of the Lyapunov function, $\Delta V(k)$, is obtained by

$$\begin{aligned}\Delta V(k) &= V(k+1) - V(k) \\ &= \gamma e_r^2(k+1) - \gamma e_r^2(k) \\ &= 2\gamma e_r(k)\Delta e_r(k+1) + \gamma \Delta e_r^2(k+1)\end{aligned} \tag{9}$$

In Eq. (9), $\Delta e_r(k+1)$ can be further expressed as

$$\begin{aligned}\Delta e_r(k+1) &= \frac{\partial e_r(k+1)}{\partial k} \\ &= \frac{\partial[r(k+1) - \hat{y}(k+1)]}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda(k)} \frac{\partial \lambda(k)}{\partial \zeta(k)} \frac{\partial \zeta(k)}{\partial k} \\ &= -\frac{\partial \hat{y}(k+1)}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda(k)} \frac{\partial \lambda(k)}{\partial \zeta(k)} \Delta\zeta(k)\end{aligned} \tag{10}$$

where

$$\begin{aligned}\frac{\partial \hat{y}(k+1)}{\partial u(k)} &= \beta_1^{k+1} \\ \frac{\partial u(k)}{\partial \lambda(k)} &= u(k-1) - \frac{1}{\beta_1^k}(v(k) - \alpha_1^k v(k-1) - \alpha_2^k v(k-2)) \\ \frac{\partial \lambda(k)}{\partial \zeta(k)} &= \lambda(k)(1-\lambda(k))\end{aligned} \tag{11}$$

Based on the on-going analysis, the following theorem provides the theoretical basis for the convergence property of the proposed updating algorithm for $\zeta(k)$.

Theorem 1. *Considering the open-loop dynamic systems that can be represented by Eq. (2), if Eq. (7) is a Lyapunov function and the IMC filter parameter in Eq. (4) is updated by the following equation:*

$$\zeta(k+1) = \zeta(k) + \frac{\eta(k)}{\beta_1^{k+1}} \times \frac{e_r(k)}{\lambda(k)(1-\lambda(k))} \left[\frac{\partial u(k)}{\partial \lambda(k)}\right]^{-1}, \quad 0 < \eta(k) < 2 \tag{12}$$

then $\Delta V(k) < 0$ holds and therefore the tracking error $e_r(k)$ is guaranteed to converge to zero asymptotically.

Proof. Based on Eqs. (10) and (12), Eq. (9) is further derived as

$$\begin{aligned}\Delta V(k) &= -2\gamma e_r(k) \frac{\partial \hat{y}(k+1)}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda(k)} \frac{\partial \lambda(k)}{\partial \zeta(k)} \Delta\zeta(k) + \gamma\left(-\frac{\partial \hat{y}(k+1)}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda(k)} \frac{\partial \lambda(k)}{\partial \zeta(k)} \Delta\zeta(k)\right)^2 \\ &= -2\gamma e_r(k)\beta_1^{k+1} \frac{\partial u(k)}{\partial \lambda(k)} \lambda(k)(1-\lambda(k)) \frac{\eta(k)}{\beta_1^{k+1}} \frac{e_r(k)}{\lambda(k)(1-\lambda(k))} \left[\frac{\partial u(k)}{\partial \lambda(k)}\right]^{-1} + \gamma\left(-\beta_1^{k+1} \frac{\partial u(k)}{\partial \lambda(k)} \lambda(k)(1-\lambda(k)) \frac{\eta(k)}{\beta_1^{k+1}} \frac{e_r(k)}{\lambda(k)(1-\lambda(k))} \left[\frac{\partial u(k)}{\partial \lambda(k)}\right]^{-1}\right)^2 \\ &= -2\gamma\eta(k)e_r^2(k) + \gamma\eta^2(k)e_r^2(k) \\ &= -\eta(k)(2-\eta(k))\gamma e_r^2(k)\end{aligned} \tag{13}$$

It is evident from Eq. (13) that $\Delta V(k)$ is always negative if $0 < \eta(k) < 2$ holds, meaning that the predicted tracking error $e_r(k)$ is guaranteed to converge to zero by using the updating algorithm, Eq. (12), to design $\zeta(k+1)$. This completes the proof.

One remark about Theorem 1 is the determination of $\eta(k)$. Generally, a larger $\eta(k)$ in the specified range [0 2] would lead to faster convergence but might result in overshoot and oscillatory response, while a smaller $\eta(k)$ has the opposite effect. Owing to the lacking of systematic guidelines for the determination of $\eta(k)$, it is usually chosen experimentally for each problem. In the proposed design, the following rules are used to update the learning rate: (i) if the increment of $|e_r(k)|$ is more than the threshold, the filter parameter remains unchanged and the learning rate is decreased by a factor l_{dec}, i.e. $\eta(k+1) = l_{\mathrm{dec}}\eta(k)$, (ii) if the absolute value of the change of $|e_r(k)|$ is within the threshold, only the filter parameter is updated; otherwise and (iii) the filter parameter is updated and the learning rate is increased by a factor l_{inc}, i.e. $\eta(k+1) = l_{\mathrm{inc}}\eta(k)$.

The implementation of the proposed adaptive IMC controller design is summarized as follows:

(1) Given the initial database for the JITL, initialize the IMC filter parameter and its learning rate.
(2) Given the current process output $y(k)$, compute the manipulate variable $u(k)$ from Eq. (5).
(3) The JITL's database is updated by the current process data if the absolute value of prediction error between the JITL's output and the current process output is larger than the specified threshold.
(4) Obtain ARX model for the next sampling instant by using the current process data and JITL algorithm, followed by updating $\eta(k)$ and $\zeta(k)$. Consequently, IMC filter parameter at the next sampling instant, $\lambda(k+1)$, is calculated by using Eq. (6).
(5) Set $k = k + 1$ and go to step 2.

3. Example

Consider a continuous polymerization reaction that takes place in a jacketed CSTR (Doyle *et al.*, 1995), where an isothermal free-radical polymerization of methyl methacrylate (MMA) is carried out using azo-bis-isobutyronitrile (AIBN) as initiator and toluene as solvent. The control objective is to regulate the product number average molecular weight (y = NAMW) by manipulating the flow rate of the initiator ($u = F_I$). This process can be described by the following balance equations:

$$
\begin{aligned}
\frac{dC_m}{dt} &= -(k_P + k_{f_m})C_m P_0 + \frac{F(C_{m_{in}} - C_m)}{V} \\
\frac{dC_I}{dt} &= -k_I C_I + \frac{F_I C_{I_{in}} - F C_I}{V} \\
\frac{dD_0}{dt} &= (0.5k_{T_c} + k_{T_d})P_0^2 + k_{f_m} C_m P_0 - \frac{FD_0}{V} \\
\frac{dD_1}{dt} &= M_m(k_P + k_{f_m})C_m P_0 - \frac{FD_1}{V}
\end{aligned}
\tag{14}
$$

where $P_0 = [2f^* k_I C_I/(k_{T_d} + k_{T_c})]^{0.5}$ and $y = (D_1/D_0)$. The model parameters and steady-state operation condition are given in Tables 1 and 2.

Table 1
Model parameters for polymerization reactor.

$k_{T_c} = 1.3281 \times 10^{10}\ \mathrm{m^3/(kmol\,h)}$	$F = 1.00\ \mathrm{m^3/h}$
$k_{T_d} = 1.0930 \times 10^{11}\ \mathrm{m^3/(kmol\,h)}$	$V = 0.1\ \mathrm{m^3}$
$k_I = 1.0225 \times 10^{-1}\ \mathrm{L/h}$	$C_{I_{in}} = 8.0\ \mathrm{kmol/m^3}$
$k_P = 2.4952 \times 10^6\ \mathrm{m^3/(kmol\,h)}$	$M_m = 100.12\ \mathrm{kg/kmol}$
$k_{f_m} = 2.4522 \times 10^3\ \mathrm{m^3/(kmol\,h)}$	$C_{m_{in}} = 6.0\ \mathrm{kmol/m^3}$
$f^* = 0.58$	

Table 2
Steady-state operating condition of polymerization reactor.

$C_m = 5.506774\ \mathrm{kmol/m^3}$	$D_1 = 49.38182\ \mathrm{kmol/m^3}$
$C_I = 0.132906\ \mathrm{kmol/m^3}$	$u = 0.016783\ \mathrm{m^3/h}$
$D_0 = 0.0019752\ \mathrm{kmol/m^3}$	$y = 25000.5\ \mathrm{kg/kmol}$

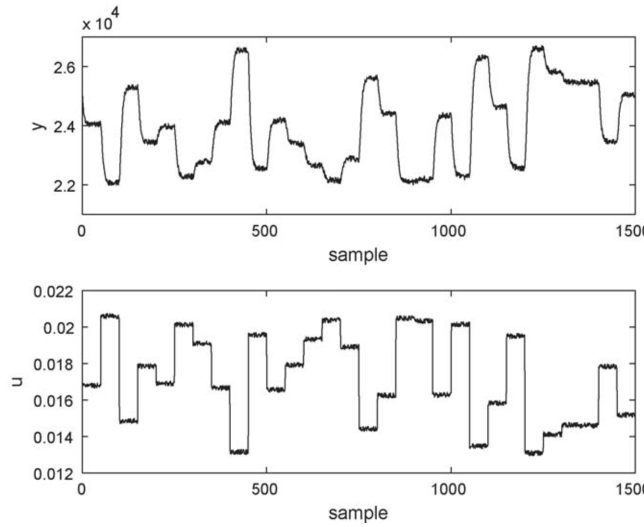

Fig. 3. Input and output data used to construct the JITL's database.

To proceed with the JITL method, input and output data are generated by introducing uniformly random steps with distribution of [0.012 0.021] to the process input F_I. With sampling time of 0.03 h, input and output data thus obtained (see Fig. 3) are used to

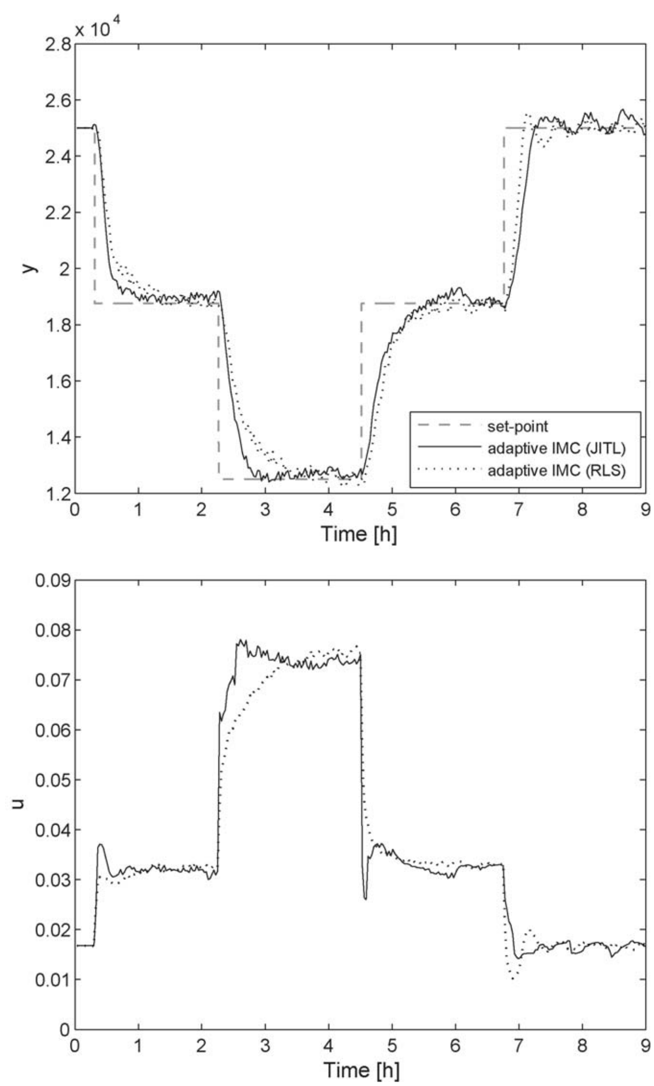

Fig. 4. Servo responses of two IMC designs.

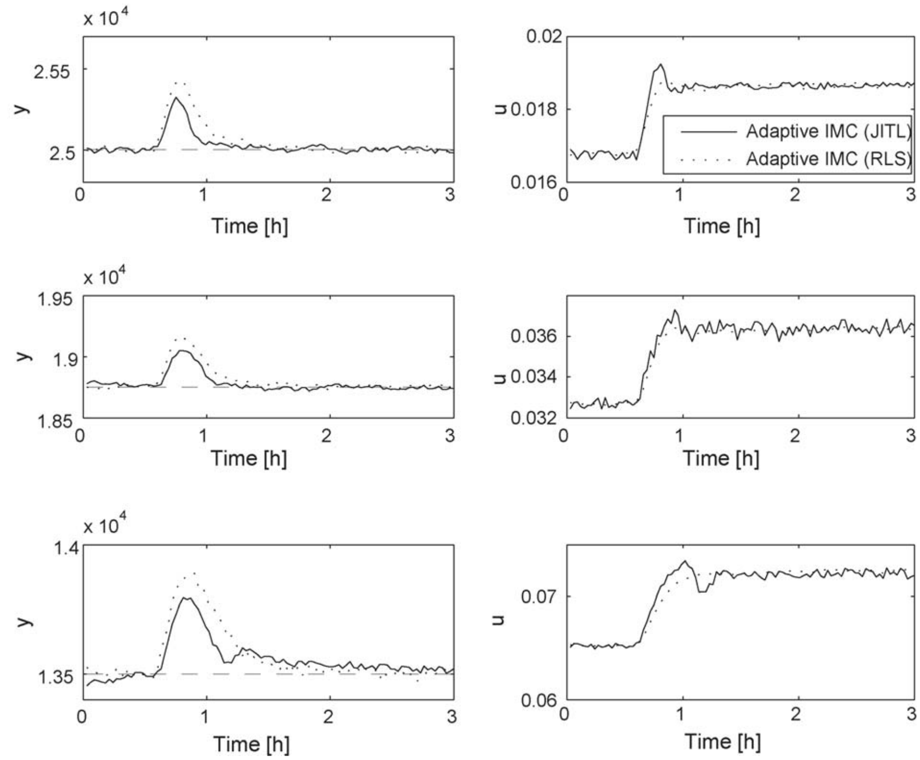

Fig. 5. Closed-loop responses for −10% step change in $C_{I_{in}}$.

build the database. Both process input and output are corrupted by 3% Gaussian white noise. A second-order ARX model is used as the local model and the parameters chosen for JITL algorithm are as follows: $\kappa = 0.95$, $k_{\min} = 6$ and $k_{\max} = 60$. The initial IMC filter is $\lambda = 0.81$ with the initial learning rate $\eta = 1 \times 10^{-7}$. For the purpose of comparison, an adaptive IMC controller is designed based on a second-order ARX model with parameter adaptation by the recursive least-square (RLS) identification procedure (Shahrokhi and Baghmisheh, 2005) and IMC filter parameter $\lambda = 0.85$.

To evaluate the servo performances of two controllers, successive set-point changes between 25000.5 and 12500 kg/kmol as illustrated in Fig. 4 are considered. It is obvious that the JITL-

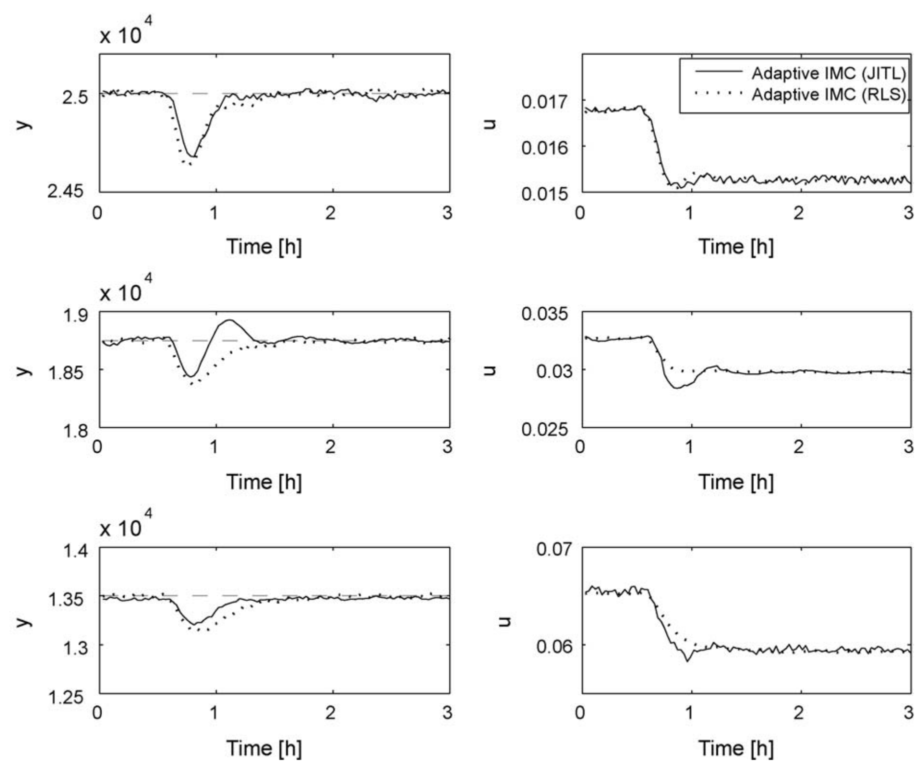

Fig. 6. Closed-loop responses for +10% step change in $C_{I_{in}}$.

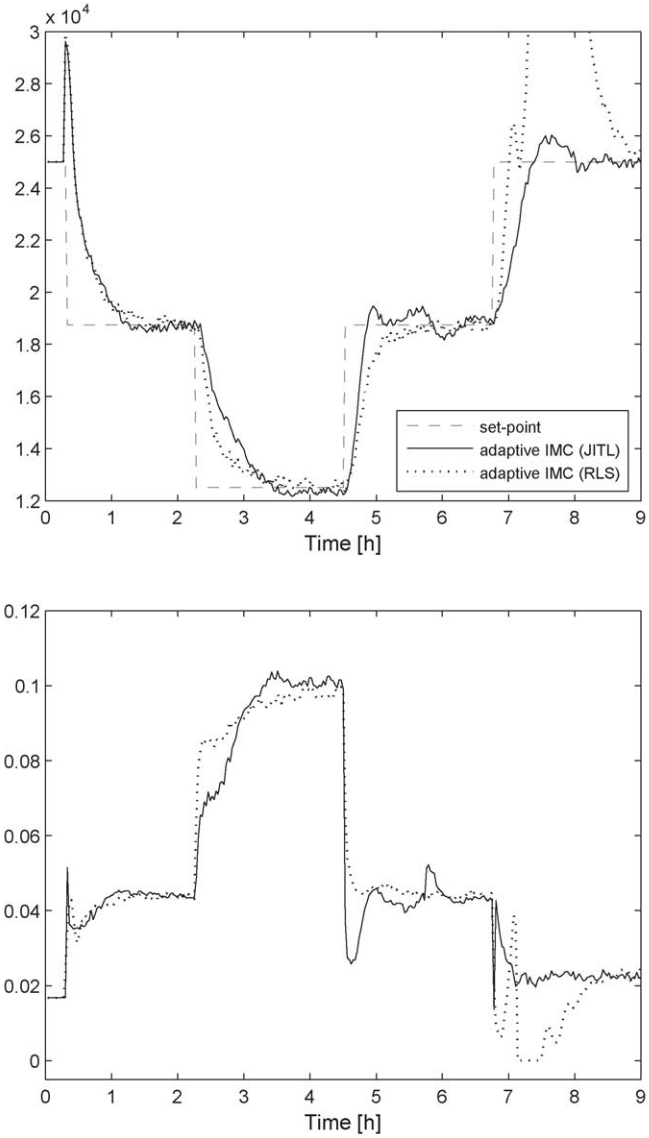

Fig. 7. Servo responses in the presence of modeling error.

based IMC design produces better performance than that achieved by the RLS-based IMC design, as evident by the corresponding reduction of the mean absolute error (MAE) by 14.9%. Next, to compare the disturbance rejection capability of two IMC controllers, unmeasured ±10% step disturbances with 5% disturbance noise in the inlet initiator concentration $C_{I_{in}}$ are considered. The resulting closed-loop responses of two controllers are compared in Figs. 5 and 6. As can be seen, the proposed IMC controller yields consistent better performance than the RLS-based IMC controller at three different operating conditions. Lastly, to evaluate the robustness of the two controllers, 10% modeling error in the kinetic parameter k_I and 20% error in the coefficients of D_1 and M_m are assumed. It is evident from Fig. 7 that the proposed controller gives better control performance than the RLS-based IMC controller because the latter fails to give satisfactory response for the last set-point change.

4. Conclusion

By incorporating the JITL into IMC framework, an adaptive IMC design methodology is developed for nonlinear process control. The IMC controller parameters are updated not only based on a set of linear models identified on-line by the JITL, but also its filter parameter is adjusted on-line by an updating algorithm derived from the Lyapunov method. Compared with the previous nonlinear IMC controller design methods, it is straightforward for the proposed method to obtain the model-inverse, and consequently the design of IMC controller can be carried out readily. Simulation results are presented to demonstrate the advantage of the proposed adaptive IMC design over its conventional counterpart.

Appendix A. JITL algorithm

Suppose that a database consisting of N process data $(y_i, \mathbf{x}_i)_{i=1\sim N}$, $y_i \in R$, $\mathbf{x}_i \in R^n$, is collected. Given a specific query data $\mathbf{x}_q \in R^n$, the objective of JITL is to predict the model output $\hat{y}_q = f(\mathbf{x}_q)$ according to the known database $(y_i, \mathbf{x}_i)_{i=1\sim N}$. To do so, the relevant data are selected from the database first by using the following similarity measure, s_i, which was recently proposed to improve the predictive accuracy of JITL technique (Cheng and Chiu, 2004):

$$s_i = \kappa\sqrt{e^{-\|\mathbf{x}_q-\mathbf{x}_i\|^2}} + (1-\kappa)\cos(\theta_i), \quad \text{if } \cos(\theta_i) \geq 0 \tag{15}$$

where κ 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 $\Delta\mathbf{x}_q = \mathbf{x}_q - \Delta\mathbf{x}_{q-1}$ and $\Delta\mathbf{x}_i = \mathbf{x}_i - \Delta\mathbf{x}_{i-1}$. The value of s_i is bounded between 0 and 1. When s_i approaches to 1, it indicates that $\mathbf{x}_i$ resembles closely to $\mathbf{x}_q$.

To apply JITL in the modeling of dynamic systems, all s_i are computed by Eq. (15) first and for each $l \in [k_{\min}\, k_{\max}]$, where $k_{\min}$ and $k_{\max}$ are the pre-specified minimum and maximum numbers of relevant data, the relevant data set $(\mathbf{y}_l, \Phi_l)$, where $\mathbf{y}_l \in R^{l\times 1}$ and $\Phi_l \in R^{l\times n}$, is constructed by selecting the l most relevant data $(y_i, \mathbf{x}_i)$ corresponding to the largest s_i to the lth largest s_i. Next, the leave-one-out cross validation test is conducted and the validation error is calculated. Upon the completion of the above procedure, the optimal l, l^*, is determined by that giving the smallest validation error. Subsequently, the predicted output for query data is calculated as $\mathbf{x}_q^T(\mathbf{P}_{l^*}^T\mathbf{P}_{l^*})^{-1}\mathbf{P}_{l^*}^T\mathbf{W}_{l^*}\mathbf{y}_{l^*}$, where $\mathbf{P}_{l^*}^T = \mathbf{W}_{l^*}\Phi_{l^*}$ and $\mathbf{W}_{l^*}$ is a diagonal matrix with entries being the first l^* largest s_i, provided this optimal model satisfies the stability constraint; otherwise, an optimization procedure is carried out to re-compute the optimal model.

References

Aha, D. W., D. Kibler, and M. K. Albert, "Instance-Based Learning Algorithms," *Mach. Learn.*, **6**, 37 (1991).

Atkeson, C. G., A. W. Moore, and S. Schaal, "Locally Weighted Learning," *Artif. Intell. Rev.*, **11**, 11 (1997).

Bhat, N. and T. J. Mcavoy, "Use of Neural Nets for Dynamic Modeling and Control of Chemical Process Systems," *Comput. Chem. Eng.*, **14**, 573 (1990).

Bontempi, G., H. Bersini, and M. Birattari, "The Local Paradigm for Modeling and Control: from Neuro-Fuzzy to Lazy Learning," *Fuzzy Sets Syst.*, **121**, 59 (2001).

Calvet, J. P. and Y. Arkun, "Feedforward and Feedback Linearization of Non-linear Systems and its Implementation Using Internal Model Control (IMC)," *Ind. Eng. Chem. Res.*, **27**, 1822 (1988).

Cheng, C. and M. S. Chiu, "A New Data-Based Methodology for Nonlinear Process Modeling," *Chem. Eng. Sci.*, **59**, 2801 (2004).

Cheng, C. and M. S. Chiu, "Adaptive IMC Controller Design for Nonlinear Process Control," *Chem. Eng. Res. Des.*, **85**, 234 (2007).

Doyle F. J. III, B. A. Ogunnaike, and R. K. Pearson, "Nonlinear Model-Based Control Using Second-Order Volterra Models," *Automatica*, **31**, 697 (1995).

Economou, C. G., M. Morari, and B. O. Palsson, "Internal Model Control. 5. Extension to Nonlinear-Systems," *Ind. Eng. Chem. Proc. Des. Dev.*, **25**, 403 (1986).

Harris, K. R. and A. Palazoglu, "Studies on the Analysis of Nonlinear Processes via Functional Expansions-III: Controller Design," *Chem. Eng. Sci.*, **53**, 4005 (1998).

Henson, M. A. and D. E. Seborg, "An Internal Model Control Strategy for Nonlinear Systems," *AIChE J.*, **37**, 1065 (1991).

Hunt, K. J. and D. Sbarbaro, "Neural Networks for Nonlinear Internal Model Control," *IEE Proc. D Control Theory Appl.*, **138**, 431 (1991).

Maksumov, A., D. J. Mulder, K. R. Harris, and A. Palazoglu, "Experimental Application of Partitioned Model-Based Control to pH Neutralization," *Ind. Eng. Chem. Res.*, **41**, 744 (2002).

Maner, B. R., F. J. III Doyle, B. A. Ogunnaike, and R. K. Pearson, "Nonlinear Model Predictive Control of a Simulated Multivariable Polymerization Reactor using Second-Order Volterra Models," *Automatica*, **32**, 1285 (1996).

Morari, M. and E. Zafiriou, *Robust Process Control,* , Prentice-Hall, Englewood Cliffs (1989).

Nahas, E. P., M. A. Henson, and D. E. Seborg, "Nonlinear Internal Model Control Strategy for Neural Network Models," *Comput. Chem. Eng.*, **16**, 1039 (1992).

Shahrokhi, M. and G. R. Baghmisheh, "Modeling, Simulation and Control of a Methanol Synthesis Fixed-Bed Reactor," *Chem. Eng. Sci.*, **60**, 4275 (2005).

Shaw, A. M., F. J. III Doyle, and J. S. Schwaber, "A Dynamic Neural Network Approach to Nonlinear Process Modeling," *Comput. Chem. Eng.*, **21**, 371 (1997).