ADAPTIVE IMC CONTROLLER DESIGN FOR NONLINEAR PROCESS CONTROL

C. Cheng and M.-S. Chiu*

Department of Chemical and Biomolecular Engineering, National University of Singapore, Singapore

Abstract: In this paper, a new adaptive internal model control (IMC) controller is proposed based on just-in-time learning (JITL) technique for nonlinear process control. The parameters of the proposed IMC controller are updated not only based on the information provided by the JITL, but also its filter parameter is adjusted online by an adaptive learning algorithm. Compared with the previous nonlinear IMC controller design methods, it is straightforward for the proposed method to obtain the model inversion based on the JITL modelling technique. Simulation results are presented to demonstrate the advantage of the proposed adaptive IMC controller design over its conventional counterpart.

Keywords: just-in-time learning; internal model control; adaptive control.

**Correspondence to:*
Professor M.-S. Chiu,
Department of Chemical and
Biomolecular Engineering,
National University of Singapore,
4 Engineering Drive 4, Singapore
117576.
E-mail: checms@nus.edu.sg

DOI: 10.1205/cherd06071

0263–8762/07/
$30.00 + 0.00

Chemical Engineering Research and Design

Trans IChemE,
Part A, February 2007

INTRODUCTION

Internal model control (IMC) is a powerful controller design strategy for the open-loop stable dynamic systems (Morari and Zafiriou, 1989). This is mainly due to two reasons. First, integral action is included implicitly in the controller because of the IMC structure. Moreover, plant/model mismatch can be addressed via the design of the robustness filter. 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) used an IMC scheme to implement their state-space linearization approach for nonlinear systems with disturbance. A disadvantage of the state-space linearization 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, 1991a). Henson and Seborg (1991b) proposed a state-space approach and used nonlinear filter to account for plant/model mismatch. However, their method relied on the availability of a nonlinear state-space model, which may be time-consuming and costly to obtain. Doyle *et al*. (1995) proposed a partitioned model inverse controller to obtain the nonlinear model inversion. This controller synthesis scheme based on Volterra model retrains the original spirit and characteristics of conventional IMC while extending its capabilities to nonlinear systems. When implemented as part of the control law, the nonlinear controller consists of a standard linear IMC controller augmented by an auxiliary loop of nonlinear 'correction'. However, Volterra model derived using local expansion results such as Carleman linearization is accurate for capturing local nonlinearities around an operating point, but may be erroneous in describing global nonlinear behaviour (Maner *et al*., 1996). Another drawback of this method is that parameters of second-order Volterra model are not parsimonious to describe the process nonlinearities. Harris and Palazoglu (1998) proposed another nonlinear 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.

The ability of artificial neural networks to model almost any nonlinear function without *a priori* knowledge has lead to the investigation of nonlinear IMC schemes using neural networks (NN). In the earlier methods given in Bhat and McAvoy (1990) and Hunt and Sbarbaro (1991), two NNs were used in the IMC framework, where one NN was trained to represent the nonlinear dynamics of process, which was used as the IMC model, while another NN was trained to learn the inverse dynamics of the process and was employed as the nonlinear 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 plant/model mismatch. Nahas *et al*. (1992) developed another NN based nonlinear IMC strategy, which consists 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, Shaw *et al*. (1997) used recurrent neural network (RNN) within the partitioned model inverse controller synthesis scheme in IMC framework and showed that this strategy provided an attractive alternative for NN-based control application. Maksumov *et al*. (2002) investigated partitioned model structure consisting of a linear ARX model and a NN model in the IMC framework. 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 model 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 used in model based controller design.

Just-in-time learning (JITL) technique (Cybenko, 1996) was recently developed as an attractive alternative for modelling the nonlinear systems. It is also known as instance-based learning (Aha *et al*., 1991), local weighted model (Atkeson *et al*., 1997), lazy learning (Aha, 1997; Bontempi *et al*., 2001), or model-on-demand (Braun *et al*., 2001; Hur *et al*., 2003) in the literature. 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. Another advantage of JITL is its inherent adaptive nature, which is achieved by storing the current measured data into the database (Nelles, 2001). Recently, to achieve better predictive performance of the JITL algorithm, Cheng and Chiu (2004) proposed an enhanced JITL algorithm by introducing a new similarity measure by combining the commonly used distance measure with the complementary angular relationship between two data samples.

In this paper, we propose to incorporate the JITL into the IMC framework to develop an adaptive IMC design methodology that takes advantage of the low-order model employed in JITL, by which the model inverse can be readily obtained for IMC design at each sampling instant. The proposed design strategy exploits the information provided by the JITL and adjusts the controller parameters on-line by an adaptive learning method. In the proposed IMC design scheme, the JITL is employed as the process model and the IMC controller is designed based on the JITL model augmented with a filter. Literature examples are presented to illustrate the proposed control strategy and a comparison with its conventional counterparts is made.

JITL-BASED ADAPTIVE IMC DESIGN

The block diagram of the IMC structure is shown in Figure 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 difficult in the use of these models in the IMC framework arises in the design of the controller. Because the IMC controller is based on the inverse of the minimum phase part of the model $\tilde{G}$, a reliable and efficient method is required to achieve this inversion (Maksumov *et al*., 2002). To this end, the JITL model is embedded in the IMC framework so that the model inverse can be obtained readily. The proposed adaptive IMC scheme is depicted in Figure 2, where the process model $\tilde{G}$

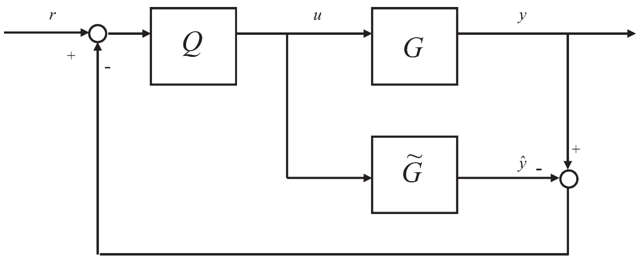

Figure 1. Block diagram of IMC structure.

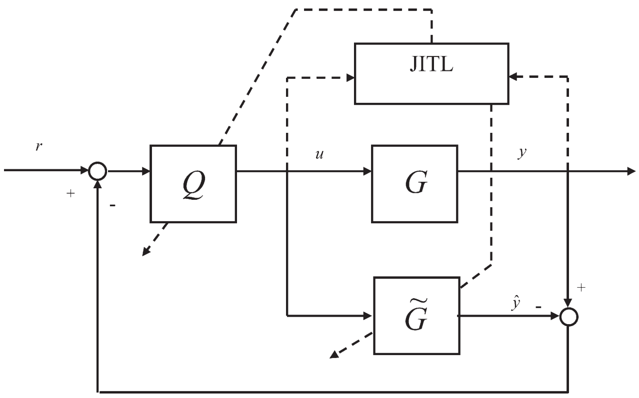

Figure 2. JITL-based adaptive IMC scheme.

Table 1. Model parameters for polymerization reactor.

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

Table 2. Steady-state operating condition of polymerization reactor.

$C_m = 5.506774\,kmol\,m^{-3}$	$D_1 = 49.38182\,kmol\,m^{-3}$
$C_1 = 0.132906\,kmol\,m^{-3}$	$u = F_I = 0.016783\,m^3\,h^{-1}$
$D_0 = 0.0019752\,kmol\,m^{-3}$	$y = 25\,000.5\,kg\,kmol^{-1}$

is updated by the JITL algorithm on-line. According to the IMC design, controller Q is designed based on the inversion of the minimum phase of process model $\tilde{G}$ augmented with a low-pass filter. In the proposed method, filter parameter of Q is not fixed, instead it is adjusted on-line by the gradient decent algorithm to be developed in the sequel. As such, the JITL is employed not only to update the model parameters but also to adjust the IMC controller as well.

The JITL technique is able to identify the current process dynamics at each sampling instant, therefore a simple model structure, e.g., low-order ARX models, can be used for each local model at every sampling instant. Therefore, the following second-order ARX model is considered in the proposed controller design,

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

where $\hat{y}(k+1)$ is the predicted output by JITL model, $y(k)$ and $u(k)$ denote the process output and input at the kth sample instant, and the model parameters α_1^k, α_2^k, and β^k are identified by the JITL at the kth sampling instant. A brief description of the JITL algorithm is given in the Appendix.

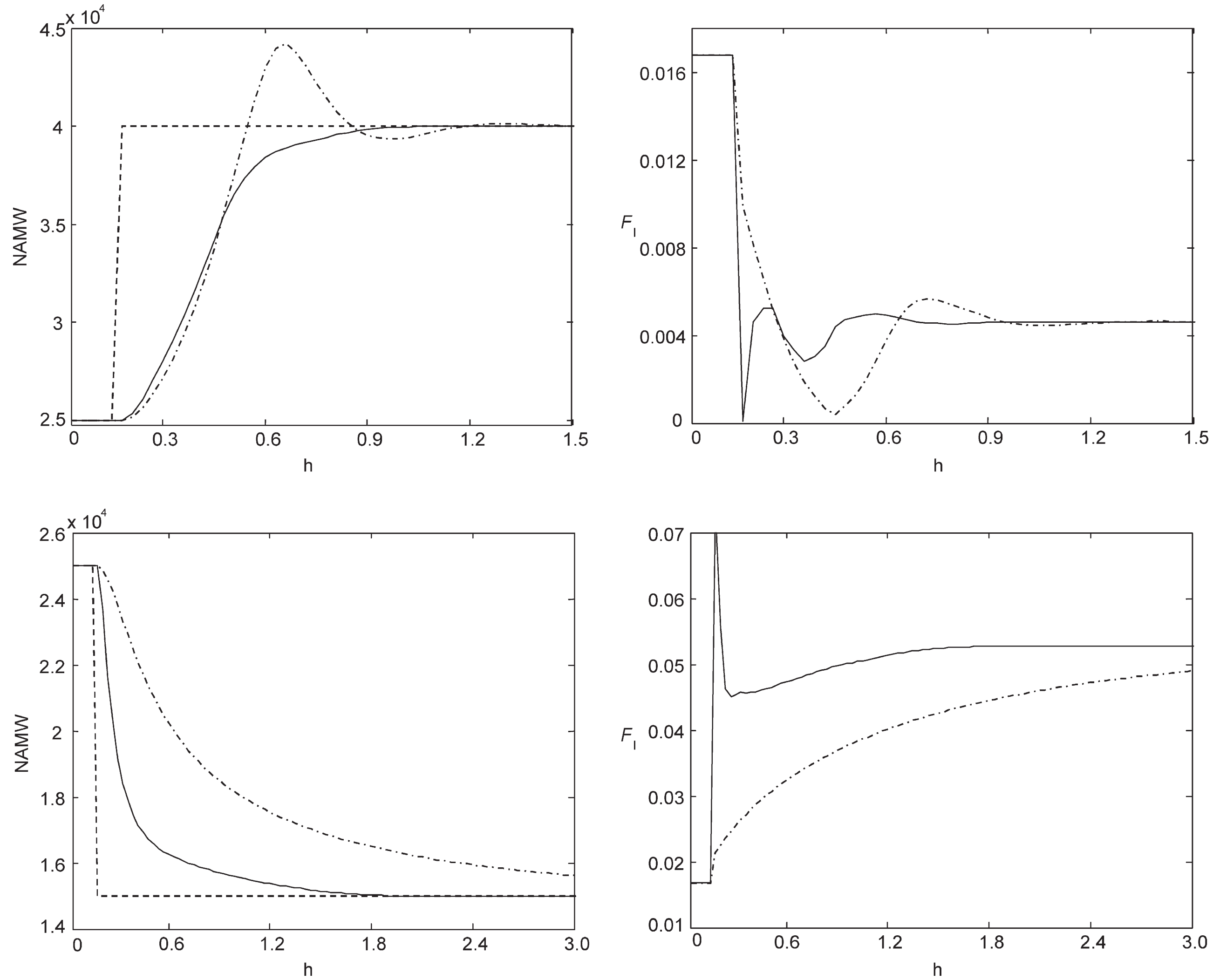

Figure 3. Closed-loop responses for set-point changes to 40 000 kg kmol^{-1} (top) and 15 000 kg kmol^{-1} (bottom). Dashed: set-point; solid: adaptive IMC; dashed-dot: IMC.

The transfer function model corresponding to equation (2) is given by:

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

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

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

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

The controller law resulting from equation (4) is then given by

$$\begin{aligned} u(k) &= \lambda(k) u(k-1) \\ &\quad + \frac{1 - \lambda(k)}{\beta^k} [\nu(k) - \alpha_1^k \nu(k-1) - \alpha_2^k \nu(k-2)] \end{aligned} \tag{5}$$

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

To update the IMC filter parameter on-line, the following objective function is considered:

$$\begin{aligned} \operatorname{Min} J(k+1) &= [r(k+1) - \hat{y}(k+1)]^2 \\ &\quad + \kappa [u(k) - u(k-1)]^2 \end{aligned} \tag{6}$$

where $r(k+1)$ is the set-point, $\hat{y}(k+1)$ is one-step-ahead prediction from the JITL, and κ is the weight parameter.

Because $\lambda(k)$ is constrained between 0 and 1, the following sigmoid function is employed to map the space [0 1] to the entire real number space:

$$\lambda(k) = \frac{1}{1 + e^{-\varsigma(k)}} \tag{7}$$

where $\varsigma(k)$ is a real number. In the sequel, an adaptive learning algorithm will be developed to update $\varsigma(k)$ on-line, and the filter parameter $\lambda(k)$ can be easily calculated by equation (7). The following summarizes the updating algorithm for $\varsigma(k)$:

$$\begin{aligned} \varsigma(k+1) &= \varsigma(k) - \eta(k) \frac{\partial J}{\partial \varsigma(k)} \\ &= \varsigma(k) - \eta(k) \frac{\partial J}{\partial \lambda(k)} \frac{\partial \lambda(k)}{\partial \varsigma(k)} \end{aligned} \tag{8}$$

$$\frac{\partial J}{\partial \lambda(k)} = \frac{\partial J}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda(k)} \tag{9}$$

$$\begin{aligned} \frac{\partial J}{\partial u(k)} &= -2(r(k+1) - \hat{y}(k+1)) \frac{\partial \hat{y}(k+1)}{\partial u(k)} \\ &\quad + 2\kappa (u(k) - u(k-1)) \end{aligned} \tag{10}$$

$$\begin{aligned} \frac{\partial u(k)}{\partial \lambda(k)} &= u(k-1) \\ &\quad - \frac{\nu(k) - \alpha_1^k \nu(k-1) - \alpha_1^k \nu(k-2)}{\beta^k} \end{aligned} \tag{11}$$

$$\frac{\partial \lambda(k)}{\partial \varsigma(k)} = \lambda(k)(1 - \lambda(k)) \tag{12}$$

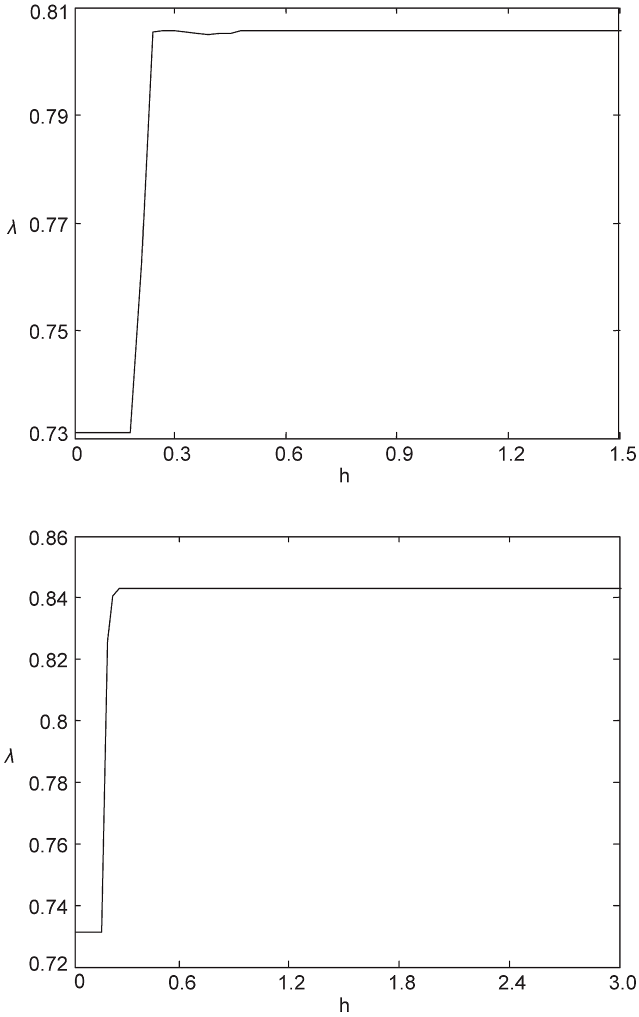

Figure 4. Updating of filter parameters for set-point changes to 40 000 kg kmol^{-1} (top) and 15 000 kg kmol^{-1} (bottom).

where $[\partial \hat{y}(k+1)/\partial u(k)]$ is obtained from the most updated ARX model identified by the JITL.

In equation (8), the controller parameter $\varsigma(k)$ and adaptive learning rate $\eta(k)$ are determined by the following rules: (1) if the increment of J is more than the threshold, the controller parameter remains unchanged and the learning rate is decreased by a factor l_{dec}, i.e., $\eta(k) = l_{dec}\eta(k-1)$; (2) if the increment of J is smaller than the threshold, only the controller parameter is updated; (3) if the increment of J is negative, the controller parameter is updated and the learning rate is increased by a factor l_{inc}, i.e., $\eta(k) = l_{inc}\eta(k-1)$. The parameters $l_{dec} = 0.7$ and $l_{inc} = 1.05$ are employed in the simulation studies presented in the next section.

Table 3. MAEs of two controllers for various set-point changes.

Set-point	IMC	Adaptive IMC	Improvement
40 000	2.80×10^3	2.18×10^3	22.2%
35 000	1.11×10^3	8.90×10^2	19.8%
30 000	5.98×10^2	4.25×10^2	28.9%
20 000	9.96×10^2	1.97×10^2	80.2%
15 000	2.75×10^3	7.29×10^2	73.5%

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

(1) Given the weight parameter κ, initialize the filter parameter and learning rate η.
(2) Given $r(k)$, $y(k)$ and $\hat{y}(k)$ at the kth sampling instant, compute $u(k)$ according to equation (5).
(3) Update the linear model by applying the JITL algorithm to the most current process data and subsequently adjust $\varsigma(k)$ and $\eta(k)$ according to the aforementioned adaptive rules and equation (8).
(4) Calculate IMC filter parameter for the next sampling instant by equation (7).
(5) Set $k = k + 1$ and go to step 2.

SIMULATION RESULTS

Example 1

Considering 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 (NAMW) by manipulating the flow rate of the initiator (F_I), i.e., NAMW is the process output y and F_I is as process input u. The dynamics of the reactor can be described by the following model equations:

$$
\begin{aligned}
\frac{\mathrm{d}C_m}{\mathrm{d}t} &= -(k_p + k_{fm})C_m P_0 + \frac{F(C_{m_{in}} - C_m)}{V} \\
\frac{\mathrm{d}C_I}{\mathrm{d}t} &= -k_I C_I + \frac{F_I C_{I_{in}} - FC_I}{V} \\
\frac{dD_0}{dt} &= (0.5k_{T_c} + k_{T_d})P_0^2 + k_{f_m} C_m P_0 - \frac{FD_0}{V} \\
\frac{dD_1}{dt} &= M_m(k_p + k_{f_m})C_m P_0 - \frac{FD_1}{V}
\end{aligned}
\tag{13}
$$

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

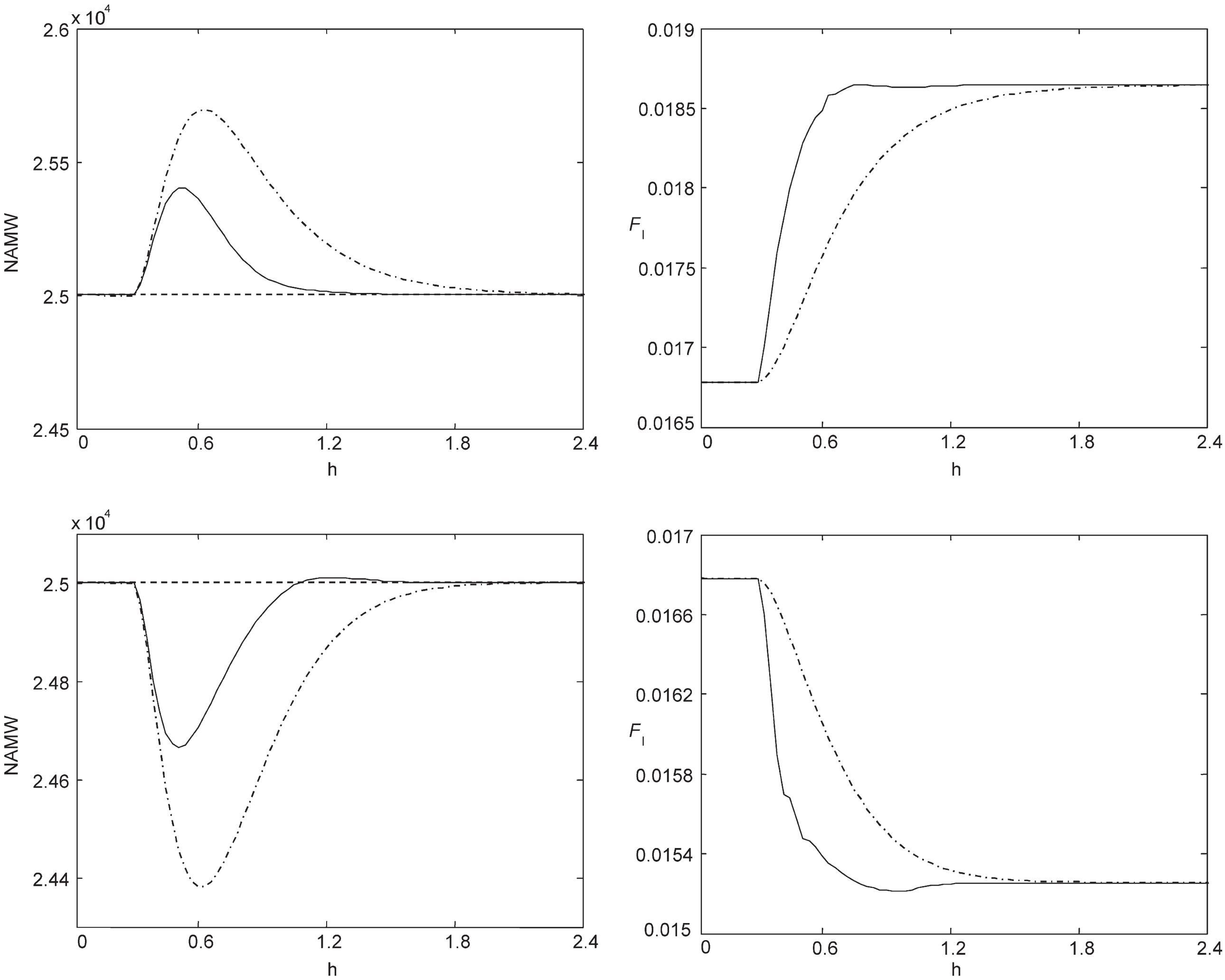

Figure 5. Closed-loop responses for -10% (top) and 10% (bottom) step changes in $C_{I_{in}}$. Dashed: set-point; solid: adaptive IMC; dashed-dot: IMC.

To apply the JITL method for process modeling, input/output data are generated by introducing uniformly random steps with distribution of [0.004, 0.080] and switching probability of 0.25 at every sampling time to the process input F_I. With sampling time of 0.03 h, input/output data thus obtained are used to build the database. A second-order ARX model is used as the local model and the parameters chosen for JITL algorithm are as follows: $\gamma = 0.9$, $k_{min} = 6$ and $k_{max} = 60$. The initial parameters used for the proposed adaptive IMC design are $\lambda = 0.731$, $\eta = 0.3$ and $\kappa = 0.1$. For the purpose of comparison, the following IMC controller, which is employed as the benchmark design in the work of Doyle *et al*. (1995), is designed based on the linear model obtained around the nominal operating condition and a second-order filter with filter time constant equal to 0.2:

$$\frac{-0.6560s^4 - 26.90s^3 - 413.3s^2 - 2822s - 72206}{s^4 + 31.79s^3 + 360.8s^2 + 1723.9s + 2947.8} \quad (14)$$

To evaluate the servo performances of two controllers, set-point changes from 25 000.5 to 40 000 kg kmol^{-1} and 15 000 kg kmol^{-1} are considered, as illustrated in Figure 3. It is obvious that adaptive IMC design has better performance than that achieved by the IMC controller. For set-point change to 40 000 kg kmol^{-1}, IMC controller gives a large overshoot and oscillatory response, while the proposed controller arrives at set-point more quickly without overshoot. As a result, the proposed controller reduces the mean squared error (MAE) by 22.2% compared with the IMC controller. For set-point change to 15 000 kg kmol^{-1}, the proposed controller reaches set-point much faster than the IMC controller, resulting in significant reduction of MAE, relative to the IMC controller, by approximately 73.5%. Figure 4 shows the updating of the filter parameter in the aforementioned closed-loop responses.

It is clear from the above simulation results that IMC controller cannot provide good control for this nonlinear process by noting that it yields an oscillatory response at the new set-point of 40 000, while a sluggish response is observed at the new set-point of 15 000. As a result, if one adopts a more aggressive IMC design (as compared with the present IMC design) to avoid sluggish response for set-point change to 15 000 kg kmol^{-1}, this will inevitably

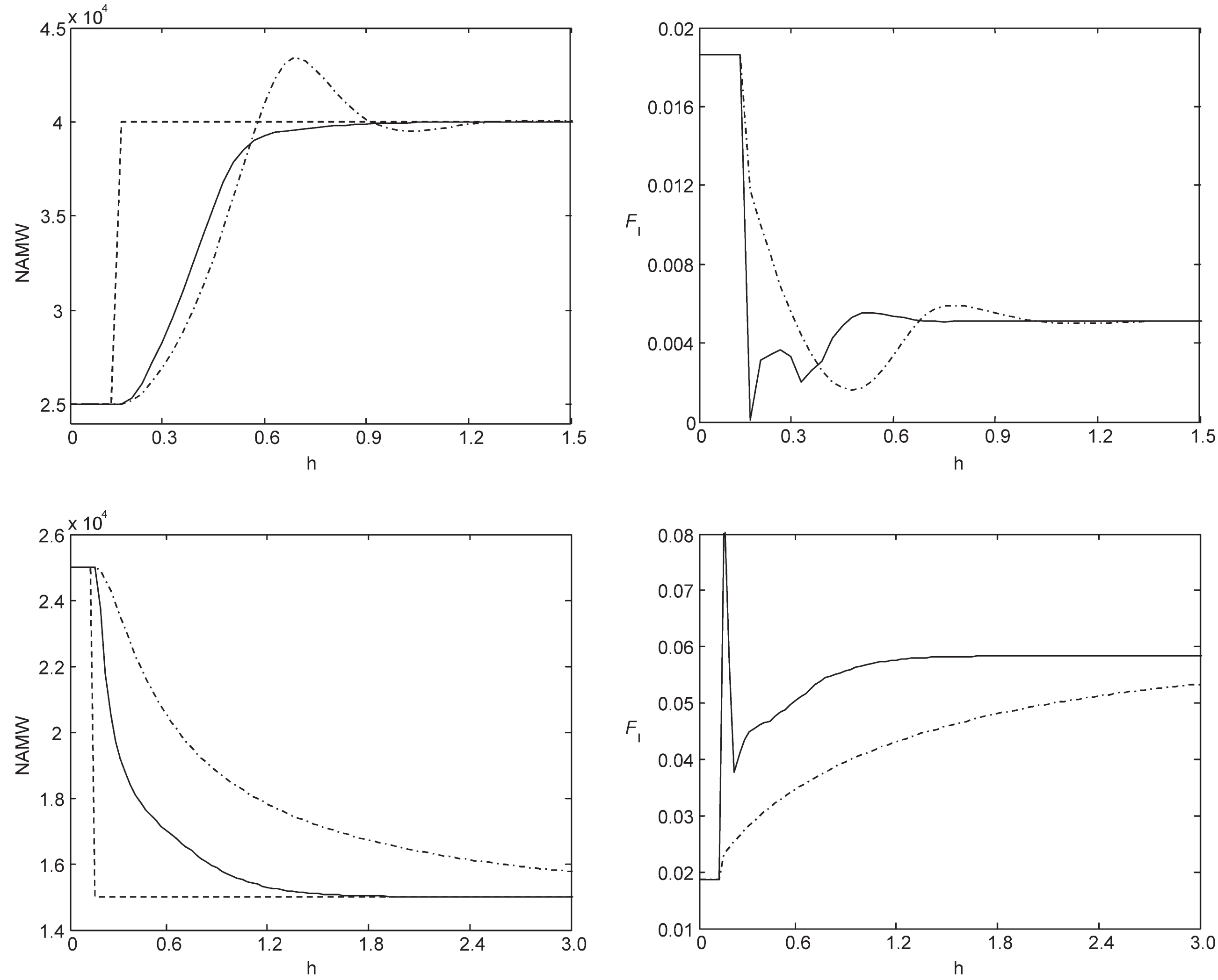

Figure 6. Close-loop responses for set-point changes to 40 000 kg mol^{-1} (top) and 15 000 kg kmol^{-1} (bottom) under -10% modelling error in k_I. Dashed: set-point; solid: adaptive IMC; dashed-dot: IMC.

make the servo response for the set-point change to 40 000 kg $kmol^{-1}$ highly oscillatory. Likewise, a more conservative IMC design can eliminate the oscillatory response in the latter case, but at the expense of even more sluggish response in the former case. Table 3 summarizes the tracking errors of these two controllers for various set-point changes. It is evident that the proposed controller gives a better performance over the operating space compared with the IMC controller.

Figure 5 compares the disturbance rejection capabilities of two controllers when $\pm 10\%$ step disturbances in $c_{I_{in}}$ are introduced into the process. It is apparent that the proposed controller has superior control performance over the IMC controller by reducing MAE by 65.9% and 64.7%, respectively. To further evaluate the robustness of the proposed controller, it is assumed that there exists -10% modelling error in the kinetic parameter k_I. As can be seen from Figure 6, the proposed controller outperforms the IMC controller, as also evidenced by the reduction of MAE by 25.6% for set-point change to 40 000 kg $kmol^{-1}$ and 70.2% for set-point change to 15 000 kg $kmol^{-1}$. Lastly, to eavaluate the effect of process noise on the proposed design, both process input and output are corrupted by 1% Gaussian white noise. As shown in Figure 7, the proposed IMC design can yield reasonably good control performance in the presence of process noise.

Example 2

Considering the van de Vusse reaction 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):

$$\frac{dC_A}{dt} = -k_1 C_A - k_3 C_A^2 + \frac{F}{V}(C_{Af} - C_A)$$
$$\frac{dC_B}{dt} = k_1 C_A - k_2 C_B - \frac{F}{V} C_B \quad (15)$$

where the parameters used are: $k_1 = 50\,h^{-1}$, $k_2 = 100\,h^{-1}$, $k_3 = 10$ L/(mol h), $C_{Af} = 10$ mol L^{-1} and $V = 1$ L. The nominal operation condition is $C_A = 3.0$ mol L^{-1}, $C_B = 1.12$ mol L^{-1}

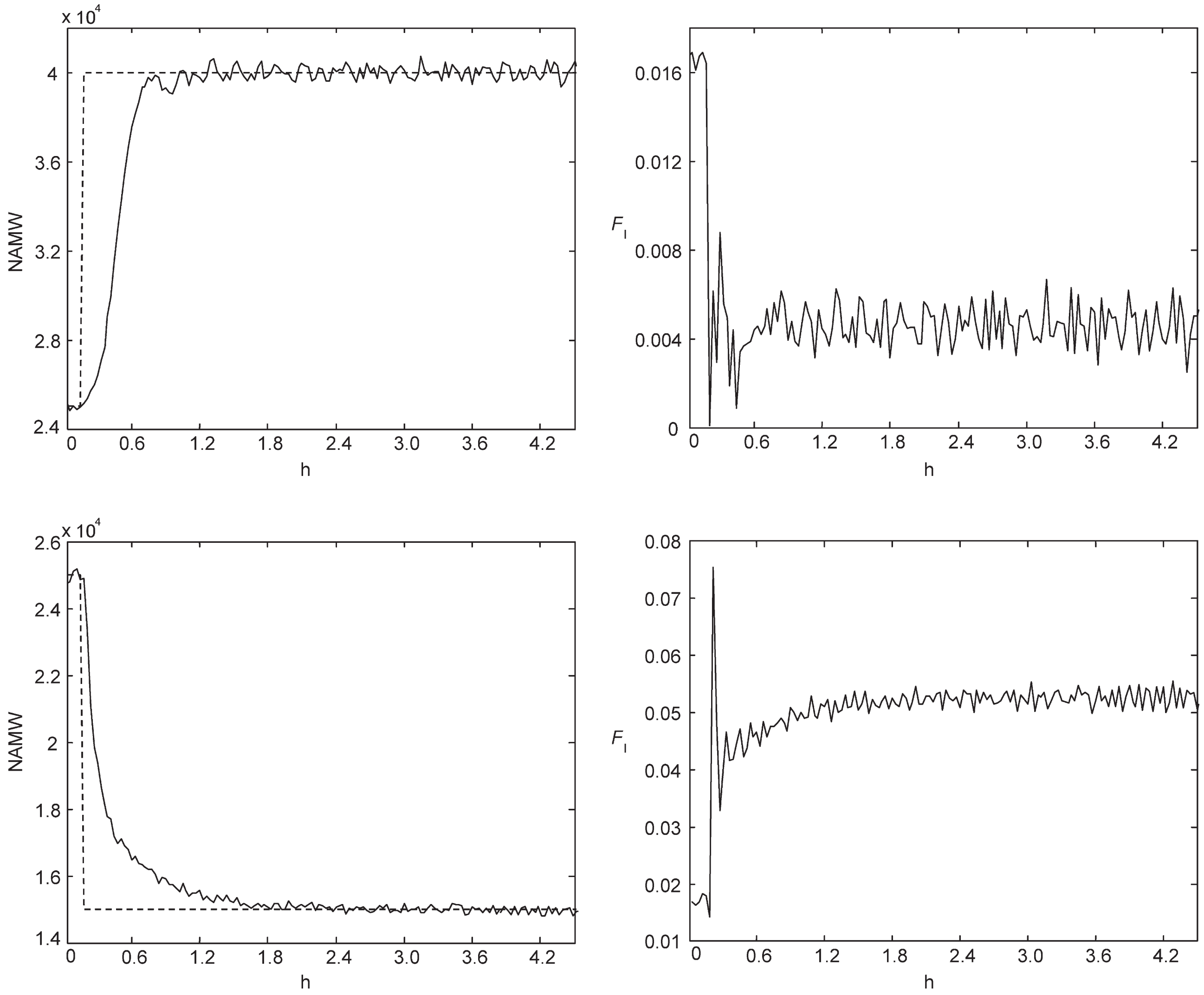

Figure 7. Servo responses in the presence of process noise.

and $F = 34.3\,\mathrm{L\,h^{-1}}$. The control objective is to regulate the concentration of component B (C_B) by manipulating the inlet flow rate F.

To apply the proposed controller strategy, a second-order ARX model is employed as the local model for JITL algorithm. A database is generated by introducing uniformly random steps with distribution of [4, 65] and switching probability of 0.1 at every sampling time to the process input F. The parameters used for JITL algorithm are: $\gamma = 0.95$, $k_{min} = 6$ and $k_{max} = 60$. In addition, the initial parameters $\lambda = 0.957$, $\eta = 0.2$, and $\kappa = 0.5$ are chosen for adaptive IMC controller. For the purpose of comparison, the benchmark IMC controller is designed based on the linear model obtained at the nominal operating condition. With a first-order filter and filter time constant equal to 0.01, the resulting IMC controller is given by (Doyle *et al.*, 1995):

$$\frac{89.29(s + 134.3)(s + 144.3)}{(s + 168.2)(s + 100)} \tag{16}$$

Figure 8 shows the servo responses of adaptive IMC and IMC controllers for 10% and -50% set-point changes, respectively. For the former, the setting time of the proposed controller is approximately 59% of that obtained by the IMC controller, resulting in 30.9% reduction of MAE. For -50% set-point change, IMC controller displays oscillatory response, while adaptive IMC controller gives smooth response and reaches the set-point faster than the IMC controller, leading to 14.8% reduction of MAE.

To evaluate the disturbance rejection performance, $\pm 10\%$ step disturbances are assumed to occur in the inlet concentration of component A. The resulting performances of two controllers are compared in the Figure 9. Again, the performance of the proposed controller shows marked improvement over that obtained by the IMC controller and consequently the resulting MAEs are reduced by 48.8% and 16.2%, respectively. The robustness of the proposed controller is also evaluated by assuming -10% modelling error in the kinetic parameter k_3. Figure 10 shows the performance of the two controller for 10% and -50% set-point changes. It is clear that the proposed controller still achieves better control

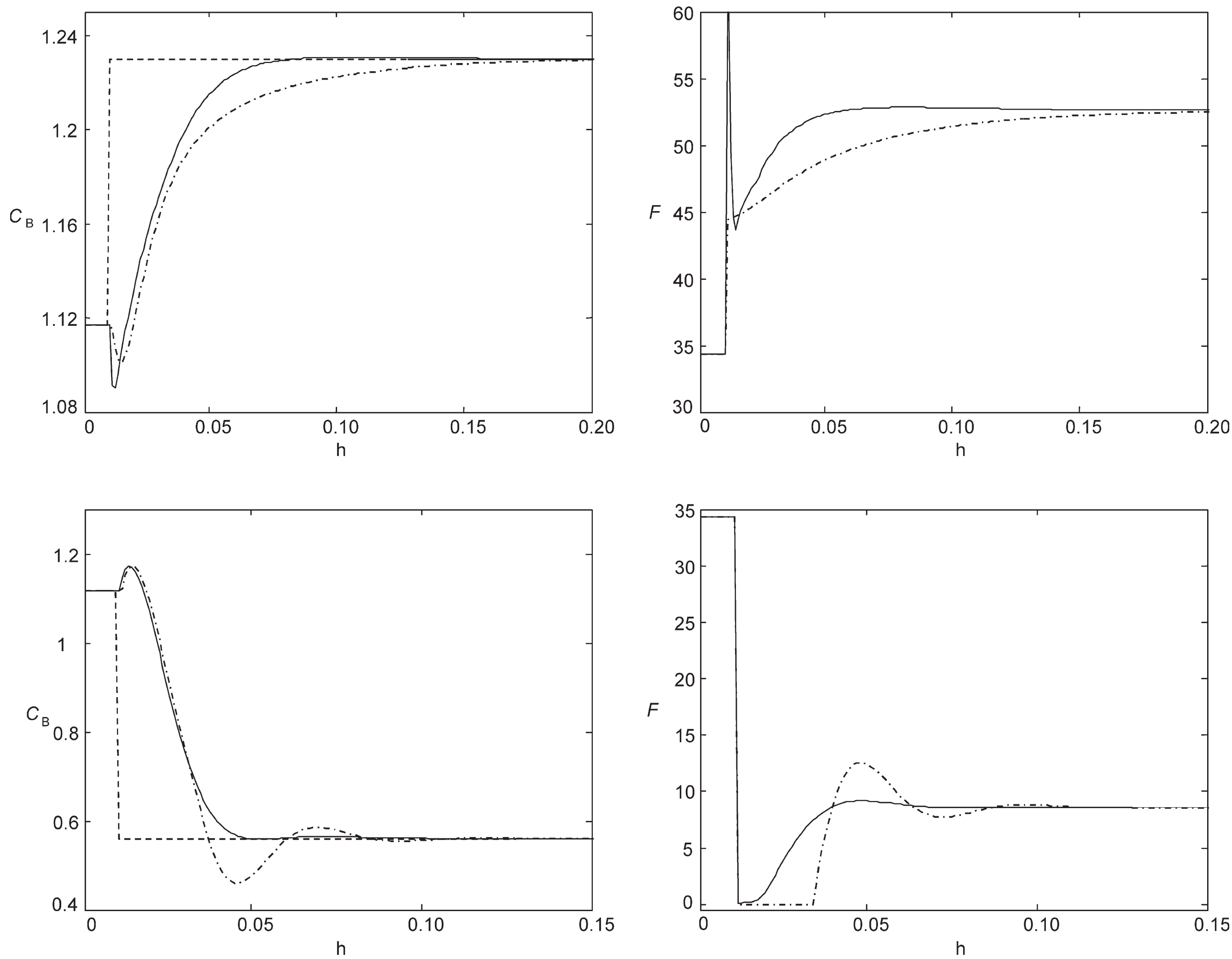

Figure 8. Closed-loop responses for 10% (top) and -50% (bottom) set-point changes. Dashed: set-point; solid: adaptive IMC; dashed-dot: IMC.

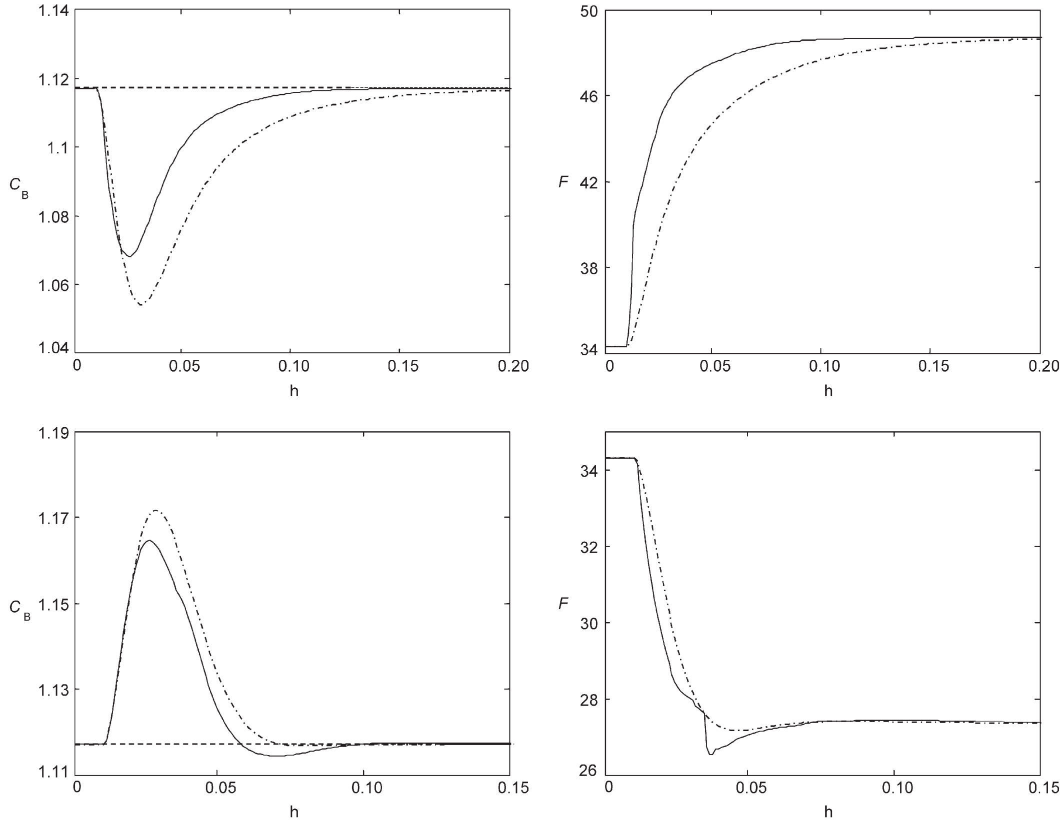

Figure 9. Closed-loop responses for -10% (top) and 10% (bottom) step changes in C_{Af}. Dashed: set-point; solid: adaptive IMC; dashed-dot: IMC.

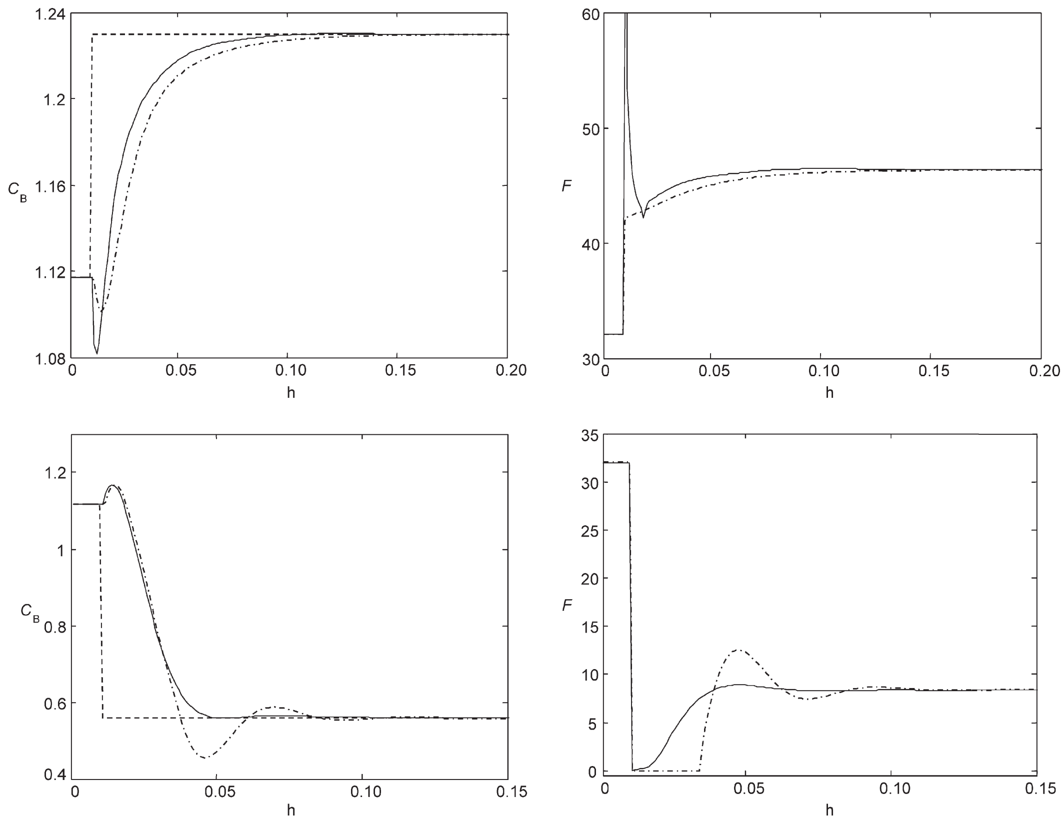

Figure 10. Close-loop responses for 10% (top) and -50% (bottom) set-point changes under -10% modelling error in k_3. Dashed: set-point; solid: adaptive IMC; dashed-dot: IMC.

performance by reducing MAEs by 23.4% and 13.2%, respectively.

CONCLUSION

By incorporating the JITL into IMC framework, an adaptive IMC design methodology is developed for nonlinear process control in this paper. The IMC controller parameters are updated not only based on the information provided by the JITL, but also its filter parameter is adjusted online by an adaptive learning algorithm. Compared with the conventional nonlinear IMC controller design method, it is straightforward for the proposed method to obtain the model inversion based on the JITL modelling technique. Simulation results are presented to demonstrate the advantage of the proposed adaptive IMC controller design over its conventional counterpart.

NOMENCLATURE

C_A, C_{Af}, C_B	reactor concentration in van de Vusse reaction
C_I, $C_{zi_{in}}$	initiator concentration
C_m, $C_{m_{in}}$	monomer concentration
D_0, D_1	dicyclopentadience concentration
$e(k)$	error between set-point and process output
F, F_1	flow rate
f	low-pass filter
f^*	parameter of polymerization reaction
J	objective function
G	process
$\tilde{G}$	process model
$\tilde{G}_-^{-1}$	minimum phase part of $\tilde{G}$
k_1, k_2	parameter of van de Vusse reaction
k_3	parameter of van de Vusse reaction
k_I	parameter of polymerization reaction
k_{fm}, k_p, k_{T_c}, k_{T_d}	parameter of polymerization reaction
k_{min}, k_{max}	number of minimum and maximum relevant data set
M_m	molecular weight of monomer
P_{l^*}	weighted relevant matrix
Q	IMC controller
$r(k)$	set-point
s_i	similarity number
$u(k)$	process input
V	reactor volume of reaction
W_{l^*}	weight matrix
$\mathbf{x}_i$, $\mathbf{x}_q$	information and query vector
$y(k)$	process output
$\hat{y}(k)$	model prediction
$\mathbf{y}_l$, $\mathbf{y}_{l^*}$	relevant process output of JITL
α_1^k, α_2^k, β^k	ARX model parameters
γ	parameter of JITL algorithm
$\eta(k)$	adaptive learning rate
θ_i	angle between $\Delta\mathbf{x}_i$ and $\Delta\mathbf{x}_q$
κ	weight of objective functions
$\lambda(k)$	IMC filter parameter
Φ_l, Φ_{l^*}	matrix of relevant data

REFERENCES

Aha, D.W., 1997, Lazy learning, *Artificial Intelligence Review*, 11: 7–10.

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

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

Bhat, N. and McAvoy, T.J., 1990, Use neural nets for dynamic modeling and control of chemical process systems, *Computer and Chemical Engineer*, 14: 573–583.

Bontempi, G., Bersini, H. and 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. and Stenman, A., 2001, A model-on-demand identification methodology for nonlinear process systems, *International Journal of Control*, 74: 1708–1717.

Calvet, J.P. and Arkun, Y., 1988, Feedforward and feedback linearization of nonlinear systems and its implementation using internal model control (IMC), *Ind Eng Chem Res*, 27: 1822–1831.

Cheng, C. and Chiu, M.S., 2004, New data-based methodology for nonlinear process modeling, *Chem Eng Sci*, 59: 2801–2810.

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

Doyle, F.J., Ogunnaike, B.A. and Pearson, R.K., 1995, Non-linear model-based control using second-order vollerra models, *Automatica*, 31: 697–714.

Economou, C.G., Morari, M. and Palsson, B.O., 1986, Internal model control. 5. Extention to nonlinear systems, *Ind Eng Chem Process Des Dev*, 25: 403–411.

Harris, K.R. and Palazoglu, A., 1998, Studies on the analysis of nonlinear process via functional expansions-III: Controller design, *Chem Eng Sci*, 53: 4005–4022.

Henson, M.A. and Seborg, D.E., 1991a, Critique of exact linearization strategies for process control, *Journal of Process Control*, 1: 122–139.

Henson M.A. and Seborg, D.E., 1991b, An internal model control strategy for nonlinear systems, *AIChE J*, 37: 1065–1078.

Hunt, K.J. and Sbarbaro, D., 1991, Neural networks for nonlinear internal model control, *IEE Proceedings-D*, 138: 431–438.

Hur, S.M., Park, M.J. and Rhee, H.K., 2003, Design and application of model-on-demand predictive controller to a semibatch copolymerization reactor, *Ind Eng Chem Res*, 42: 847–859.

Maksumov, A., Mulder, D.J., Harris, K.R. and Palazoglu, A., 2002, Experimental application of partitioned model-based control to pH neutralization, *Ind Eng Chem Res*, 41: 744–750.

Maner, B.R., Dolye, F.J., Ogunnaike, B.A. and Pearson, P.K., 1996, Nonlinear model predictive control of a simulated multivariable polymerization reactor using second-order Volterra models, *Automatica*, 32: 1285–1301.

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

Nahas, E.P., Henson, M.A. and Seborg, D.E., 1992, Nonlinear internal model control strategy for neural network models, *Computers and Chemical Engineering*, 16: 1039–1057.

Nelles, O., 2001, *Nonlinear System Identification* (Springer-Verlag, Berlin, Germany).

Shaw, A.M., Dolye, F.J. and Schwaber, J.S., 1997, Dynamic neural network approach to nonlinear process modeling, *Computers and Chemical Engineering*, 21: 371–385.

The manuscript was received 3 July 2006 and accepted for publication after revision 10 October 2006.

APPENDIX: 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).

$$s_i = \gamma \cdot \sqrt{e^{-d^2(\mathbf{x}_q,\mathbf{x}_i)}} + (1-\gamma)\cdot\cos(\theta_i), \quad \text{if } \cos(\theta_i) \geq 0 \tag{17}$$

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 $\Delta\mathbf{x}_q = \mathbf{x}_q - \mathbf{x}_{q-1}$ and $\Delta\mathbf{x}_i = \mathbf{x}_i - \mathbf{x}_{i-1}$. 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 modelling of dynamic systems, all s_i are computed by equation (17) first and for each $l \in [k_{min}\, k_{max}]$, where k_{min} and k_{max} are the pre-specified minimum and maximum number of relevant data, the relevant data set ($\mathbf{y}_l$, Φ_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. 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 $\mathbf{x}_q^T(P_{l^*}^T P_{l^*})^{-1} P_{l^*}^T W_{l^*} \mathbf{y}_{l^*}$, where $P_{l^*} = W_{l^*}\Phi_{l^*}$ and W_{l^*} is a diagonal matrix with entries being the first $W_l \in R^{l\times l}$ largest s_i, provided this optimal model satisfies the stability constraint; otherwise, an optimization procedure is carried out to recompute the optimal model.