IDENTIFICATION AND CONTROL OF GENERALIZED HAMMERSTEIN PROCESSES

YE MYINT HLAING
(B.Sc., B.E.)

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF ENGINEERING

DEPARTMENT OF CHEMICAL AND BIOMOLECULAR ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2005

ACKNOWLEDGEMENT

I wish to express my thanks to the following institution and persons, without whose assistance and guidance this thesis would not have been possible.

- To my supervisor, Associate Professor Min-Sen Chiu, for his constant guidance, kindness, forgiveness, care, concern shown throughout the project and time taken to read the manuscript.

- To the National University of Singapore, for the postgraduate research scholarship, without which I would not be able to continue my higher degree studies.

- Special thanks and appreciation are due to Cheng Cheng, Dr. Jia Li, Yasuki Kansha, Ankush Kalmukale for the simulating discussions that we have had and the help that they have rendered to me. My association with them was an enriching experience.

- To all the technical and clerical staff in the Chemical & Biomolecular Engineering Department for their patience and help.

- To my parents and family members and my best friend Miss Moe Thida Aung for their continuous love and encouragement throughout the study.

- Last but not least, my thanks to all who have contributed in one-way or another to make this thesis possible.

TABLE OF CONTENTS

SUMMARY

In this study, iterative identification procedures for generalized single-input single-output (SISO) and multi-input multi-output (MIMO) Hammerstein models are developed. By incorporating generalized Hammerstein model into controller design, adaptive IMC design method and adaptive PID control strategy are developed. The main contributions of this thesis are as follows.

(1) A generalized Hammerstein model consisting of a static nonlinear part in series with time-varying linear model is proposed. The generalized Hammerstein model is identified by updating the parameters of linear model and nonlinear part in an iterative manner. This method is applied to the identification of both SISO and MIMO generalized Hammerstein models. Simulation results demonstrate that generalized Hammerstein model has better predictive performance than the conventional Hammerstein model.

(2) Adaptive controller design methods for nonlinear processes using generalized Hammerstein model are proposed. For SISO processes, adaptive IMC design and adaptive PID controller are developed, while an adaptive decentralized PID controller is devised for MIMO processes. The proposed methods employ the reciprocal of static nonlinear part in order to remove the nonlinearity of the processes so that the resulting controller design is amenable to linear control design techniques. Parameter updating equations are developed by the gradient descent method and are used to adjust the controller parameters online. Simulation results show that the proposed adaptive controllers give better performance than their conventional counterparts.

NOMENCLATURE

A_1 = cross-sectional area of tank 1

A_2 = cross-sectional area of tank 2

A_w = heat exchange area

C_B = concentration of component B

$C_{I,in}$ = inlet concentration of initiator

$C_{m,in}$ = inlet concentration of monomer

C_p = average heat capacity

$C_{p,w}$ = coolant heat capacity

C_{v1} = constant valve coefficient

d_i = distance between $\mathbf{x}_i$ and $\mathbf{x}_q$

F = inlet flow rate of monomer

F_I = inlet initiator flow rate

f = low-pass filter

G = process

$\tilde{G}$ = model of the process

$\tilde{G}_-$ = minimum phase of $\tilde{G}$

ΔH = heat of reaction

h_1 = level of tank 1

h_2 = level of tank 2

k_w = coolant conductivity

M_m = molecular weight of monomer

M_p = number average molecular weight

m_w = coolant mass

N = number of input and output data

Q = IMC controller

Q_w = external heat exchanger duty

q_1 = base stream

q_2 = buffer stream

q_3 = acid stream

r = set-point

s_i = similarity number

T = reactor temperature

T_w = coolant temperature

T_0 = inlet temperature

u = process input

V = Reactor volume

W_{ai} = charge related quantity

W_{bi} = concentration of the CO_3^{2-} ion

w_1^k, w_2^k, w_3^k = parameters of adaptive PID controller

$\mathbf{x}_i, \mathbf{x}_q$ = past values of both process input and process output

y = process output

Greek Letters

α, β, γ = parameters of Hammerstein model

$\boldsymbol{\varepsilon}$ = model approximation error

τ = closed-loop time constant

Ω = weight parameter

ϑ_i = angle between $\Delta\mathbf{x}_i$ and $\Delta\mathbf{x}_q$

ρ = average density

λ = IMC filter time constant

η = user-specified learning rate

Abbreviations

JITL = just-in-time learning

IMC = internal model control

MAE = mean absolute error

MIMO = multi-input multi-output

PID = proportional-integral-derivative

SISO = single-input single-output

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Motivation

A chemical plant is a complex of many sub-unit processes and each sub-unit process may possess severe nonlinearity due to inherent features such as reaction kinetics and transport phenomena. Due to this complexity and nonlinearity, conventional linear controllers commonly used in industrial chemical plants show very different control performances depending on operating conditions. Many advanced control schemes have been developed to efficiently control nonlinear chemical process based on their mathematical models. However, it is very costly and time consuming procedure to rigorously develop and validate nonlinear models of chemical processes. To overcome these difficulties, the construction of models directly from the observed behavior of processes has attracted much attention in the recent past.

Nonlinear system identification from input-output data can be performed using general types of nonlinear models such as neuro-fuzzy networks, neural networks, Volterra series or other various orthogonal series to describe nonlinear dynamics. 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. To simplify the aforementioned problems of identifying a nonlinear model from input-output data, the

other alternative is to use block-oriented nonlinear models consisting of static nonlinear function and linear dynamics subsystem such as Hammerstein model, Wiener model and feedback block-oriented model (Pearson and Pottmann, 2000). When the nonlinear function precedes the linear dynamic subsystem, it is called the Hammerstein model, whereas if it follows the linear dynamic subsystem, it is called the Wiener model. A less common class of feedback block-oriented model structures is static nonlinearities in the feedback path around a linear model.

It has been shown that Hammerstein models can effectively model a number of chemical processes, e.g. pH neutralization processes (Lakshminarayanan et al., 1995; Fruzzetti et al., 1997) and polymerization reactor (Su and McAvoy, 1993; Ling and Rivera, 1998). The Hammerstein structure is useful in situations where the process gain changes with the operating conditions while the dynamics remain fairly constant. However, when both process gain and dynamics change over the region of process operation, the modeling accuracy of Hammerstein model may deteriorate significantly (Lakshminarayanan et al., 1997). Thus control system designs based on Hammerstein model may not deliver acceptable performance in this situation. The problem caused by the restriction of Hammerstein model consequently motivates the proposed research to investigate a new model called generalized Hammerstein model and its associated identification and controller design problems.

1.2 Contributions

In this thesis, iterative identification procedures for generalized single-input single-output (SISO) and multi-input multi-output (MIMO) Hammerstein models are

developed. By incorporating generalized Hammerstein model into controller design, adaptive IMC design method and adaptive PID control strategy are developed. The main contributions of this thesis are as follows.

Firstly, a generalized Hammerstein model consisting of a static nonlinear part in series with time-varying linear model is proposed. The generalized Hammerstein model is identified by updating the parameters of linear model and nonlinear part in an iterative manner. This method is applied to the identification of both SISO and MIMO generalized Hammerstein models. Simulation results demonstrate that generalized Hammerstein model has better predictive performance than the conventional Hammerstein model.

Secondly, adaptive controller design methods for nonlinear processes using generalized Hammerstein model are proposed. For SISO processes, adaptive IMC design and adaptive PID controller are developed, while an adaptive decentralized PID controller is devised for MIMO processes. The proposed methods employ the reciprocal of static nonlinear part in order to remove the nonlinearity of the processes so that the resulting controller design is amenable to linear control design techniques. Parameter updating equations are developed by the gradient descent method and are used to on-line adjust the controller parameters. Simulation results show that the proposed adaptive controllers give better performance than their conventional counterparts.

1.3 Thesis Organization

The thesis is organized as follows. Chapter 2 will review the concept of Just-in-Time learning algorithm and Narendra-Gallman method for iterative identification of Hammerstein model. The proposed identification methods for SISO and MIMO

generalized Hammerstein are developed in Chapter 3. Adaptive IMC design and adaptive PID controller for SISO generalized Hammerstein processes are developed in Chapter 4, while adaptive decentralized PID controller for MIMO generalized Hammerstein processes are presented in Chapter 5. The general conclusion and suggestions for future work are given in Chapter 6.

CHAPTER 2

Literature Survey

This chapter will give a brief overview of the Hammerstein model and previous results on the identification of Hammerstein model. Also the concept of Just-in-Time learning (JITL) algorithm which is employed in the proposed modeling and controller design methods is briefly reviewed. Some relevant background will also be presented for further development of this thesis.

2.1 Hammerstein Model

Many chemical processes have been modeled with Hammerstein model, for example pH neutralization processes (Lakshminarayanan et al., 1995; Fruzzetti et al., 1997), distillation columns (Eskinat et al., 1991; Pearson and Pottmann, 2000), heat exchangers (Eskinat et al., 1991; Lakshminarayanan et al., 1995) and polymerization reactor (Su and McAvoy, 1993; Ling and Rivera, 1998). Various system identification methods have been proposed to identify the Hammerstein model as depicted in Figure 2.1, which consists of a static nonlinear part (NL) and a linear dynamics $G(z)$, where the former is modeled in different manners such as using polynomials or a multilayer feedforward neural network (MFNN). Narendra and Gallman (1966) developed an iterative procedure to identify the nonlinear and linear parts, which is referred as

Narendra-Gallman method in this thesis. A number of papers extended linear identification method to identify Hammerstein model by treating such model as a multi-input single-output (MISO) linear model. For example, Chang and Luus (1971) used a simple least squares technique to estimate the system parameters. A comparison of the simple least squares estimation with the Narendra-Gallman method is given by Gallman (1976). Several approaches have been proposed to identify complex static nonlinear functions without iterative optimization. For example, Pottman et al. (1993) used Kolmogorov-Gabor polynomials to describe highly nonlinear dynamics. An optimal two-stage identification algorithm was proposed to extract the model parameters using singular value decomposition after estimating an adjustable parameter vector. Identification of discrete Hammerstein systems using kernel regression estimate was considered by Greblicki and Pawlak (1986). A nonparametric polynomial identification algorithm for the Hammerstein system was proposed by Lang (1997). Identification of Hammerstein models using multivariate statistical tools was proposed by Lakshminarayanan et al. (1995). Al-Duwaish and Karim (1997) used a hybrid model which consists of a MFNN to identify the static nonlinear part in series with autoregressive moving average (ARMA) model for identification of single-input single-output (SISO) and multi-input multi-output (MIMO) Hammerstein model with separate or combined nonlinearities.

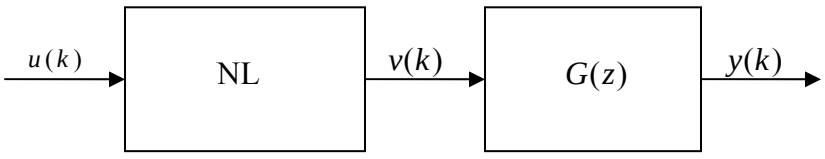


Figure 2.1 Hammerstein model

Because the modeling method to be developed in this thesis is based on the iterative identification procedure employed in the Narendra-Gallman method, a review of this method is given in what follows. In Narendra-Gallman method, the static nonlinear function is assumed to be approximated by a finite polynomial and therefore the Hammerstein model can be described by the following equation:

$$y(k) = \alpha_1 y(k-1) + \ldots + \alpha_{n_y} y(k-n_y) + \beta_1 v(k-1-n_d) + \ldots + \beta_{n_v} v(k-n_v-n_d) \qquad (2.1)$$

$$v(k) = \gamma_1 u(k) + \gamma_2 u^2(k) + \ldots + \gamma_m u^m(k) \qquad (2.2)$$

where $y(k)$ and $u(k)$ denote the process output and input at the k-th sampling instant, respectively, $v(k)$ is unmeasurable internal variable, α_i $(i = 1 \sim n_y)$ and β_i $(i = 1 \sim n_v)$ are the parameters of linear dynamics, γ_i $(i = 1 \sim m)$ are the parameters of static nonlinear part, n_y and n_v are integers related to the model order, and n_d is process time-delay.

Although the intermediate variable $v(k)$ cannot be measured, it can be eliminated from the output equation readily as given by:

$$\begin{aligned} y(k) = {} & \alpha_1 y(k-1) + \ldots + \alpha_{n_y} y(k-n_y) + \beta_1 \gamma_1 u(k-1-n_d) + \ldots + \beta_1 \gamma_m u^m(k-1-n_d) \\ & + \ldots + \beta_{n_v} \gamma_1 u(k-n_v-n_d) + \ldots + \beta_{n_v} \gamma_m u^m(k-n_v-n_d) \end{aligned} \qquad (2.3)$$

For brevity, Eq. (2.3) can be conveniently expressed by:

$$y(k) = \frac{B(q^{-1})}{A(q^{-1})} \sum_{j=1}^{m} \gamma_j u^j(k) \qquad (2.4)$$

where the polynomials $A(q^{-1})$ and $B(q^{-1})$ are given by:

$$\begin{aligned} A(q^{-1}) &= 1 - \alpha_1 q^{-1} - \ldots - \alpha_{n_y} q^{-n_y} \\ B(q^{-1}) &= \beta_1 q^{-1-n_d} + \beta_2 q^{-2-n_d} + \ldots + \beta_{n_v} q^{-n_v-n_d} \end{aligned} \qquad (2.5)$$

The identification procedure proposed by Narendra and Gallman (1966) essentially obtains the parameters of the Hammerstein model by separating the estimation problem of the linear dynamics from that of static nonlinear part. When the parameters γ_i $(i = 1 \sim m)$ are known, the intermediate variable $v(k)$ can be obtained from Eq. (2.2). Therefore, the process output can be predicted as:

$$\mathbf{y} = \mathbf{V}\psi + \boldsymbol{\varepsilon} \tag{2.6}$$

where $\boldsymbol{\varepsilon}$ is the approximation error and

$$\begin{aligned}
&\mathbf{y} = \left[y(1), y(2), \ldots, y(N)\right]^T \\
&\psi = \left[\hat{\alpha}_1, \hat{\alpha}_2, \ldots, \hat{\alpha}_{n_y}, \hat{\beta}_1, \hat{\beta}_2 \ldots, \hat{\beta}_{n_v}\right]^T \\
&\chi(k) = \left[y(k-1), \ldots, y(k-n_y), v(k-1-n_d), \ldots, v(k-n_v-n_d)\right]^T \\
&\mathbf{V} = \left[\chi(1), \chi(2), \ldots, \chi(N)\right]^T \\
&\boldsymbol{\varepsilon} = [\varepsilon(1), \varepsilon(2), \ldots, \varepsilon(N)]^T
\end{aligned} \tag{2.7}$$

where $\hat{\alpha}_i$ $(i = 1 \sim n_y)$ and $\hat{\beta}_i$ $(i = 1 \sim n_v)$ are the linear model parameters to be estimated, and N is the number of input and output data.

Subsequently, the parameters of the linear dynamics $G(z)$ of the Hammerstein model can be computed from

$$\psi = (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T\mathbf{y} \tag{2.8}$$

On the other hand, when the parameters of the linear dynamics are available, the parameters of nonlinear part can be obtained by solving the following objective function:

$$\underset{\theta}{\text{Min}}\, E(\theta) = \frac{1}{N}\sum_{k=1}^{N}(y(k) - \hat{y}(k;\theta))^2 \tag{2.9}$$

where $\hat{y}(k;\theta)$ is the output of Hammerstein model:

$$\hat{y}(k;\theta) = \frac{\hat{B}(q^{-1})}{\hat{A}(q^{-1})}\sum_{j=1}^{m}\hat{\gamma}_j u^j(k) \tag{2.10}$$

$$\hat{A}(q^{-1}) = 1 - \hat{\alpha}_1 q^{-1} - \ldots - \hat{\alpha}_{n_y} q^{-n_y}$$
$$\hat{B}(q^{-1}) = \hat{\beta}_1 q^{-1-n_d} + \hat{\beta}_2 q^{-2-n_d} + \ldots + \hat{\beta}_{n_v} q^{-n_v-n_d} \tag{2.11}$$

$$\theta = [\hat{\gamma}_1, \hat{\gamma}_2, \ldots, \hat{\gamma}_m]^T \tag{2.12}$$

and $\hat{\gamma}_i$ $(i = 1 \sim m)$ are the parameters of static nonlinear part to be identified.

By differentiating the objective function $E(\theta)$ given in Eq. (2.9) obtains (Eskinat et al., 1991):

$$\frac{\partial E}{\partial \theta} = \frac{2}{N} \sum_{k=1}^{N} \frac{\hat{B}(q^{-1})}{\hat{A}(q^{-1})} \mathbf{u}(k) \left(y(k) - \frac{\hat{B}(q^{-1})}{\hat{A}(q^{-1})} \mathbf{u}^T \theta \right) \tag{2.13}$$

where

$$\mathbf{u}(k) = [u(k), u^2(k), \ldots, u^m(k)]^T \tag{2.14}$$

$$\frac{\partial E}{\partial \theta} = \left[\frac{\partial E}{\partial \hat{\gamma}_1}, \frac{\partial E}{\partial \hat{\gamma}_2}, \ldots, \frac{\partial E}{\partial \hat{\gamma}_m} \right]^T \tag{2.15}$$

By setting Eq. (2.13) to zero, the solution of θ can be solved by:

$$\theta = \left[\sum_{k=1}^{N} \frac{\hat{B}(q^{-1})}{\hat{A}(q^{-1})} \mathbf{u}(k) \frac{\hat{B}(q^{-1})}{\hat{A}(q^{-1})} \mathbf{u}^T(k) \right]^{-1} \times \left[\sum_{k=1}^{N} \frac{\hat{B}(q^{-1})}{\hat{A}(q^{-1})} \mathbf{u}(k) y(k) \right] \tag{2.16}$$

To conclude this section, the identification procedure of Narendra-Gallman method can be summarized as follows:

1. Given the process data $\{y(k), u(k)\}_{k=1\sim N}$ and the parameters of static nonlinear part are initialized as $\hat{\gamma}_1 = 1$ and $\hat{\gamma}_i = 0$ $(i \neq 1)$;
2. Compute $v(k)$ from Eq. (2.2) and calculate the parameters of linear dynamics by Eq. (2.8);

3. Solve the static nonlinear part based on the result obtained in step 2 and Eq. (2.16) ;
4. When the convergence criterion is met, stop; otherwise, go to step 2 by using the updated parameters $\hat{\gamma}_i$ obtained in step 3.

2.2 Just-in-Time Learning Methodology

Aha et al. (1991) developed Instant-based learning algorithms for modeling the nonlinear systems. This approach is inspired by ideas from local modeling and machine learning techniques. Subsequent to Aha's work, different variants of instance-base learning are developed, e.g. locally weight learning (Atkeson et al., 1997) and just-in-time learning (JITL) (Bontempi et al., 1999). Standard methods like neural networks and neuro-fuzzy are typically trained offline. Thus, all learning data is processed a priori in a batch-like manner. This can become computationally expensive for huge amounts of data. In contrast, JITL has no standard learning phase. It merely gathers the data and stores 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.

Recently, a refined JITL algorithm by using both distance measure and angle measure as similarity criterion was developed by Cheng and Chiu (2004). This algorithm will be employed in this research and therefore it is described in the remaining of this section.

Step 1: Given the database $\{(y_i, \mathbf{x}_i)\}_{i=1\sim N}$ where the vector $\mathbf{x}_i$ is formed by the past values of both process input and process output, the parameters $k_{\min}$, $k_{\max}$, and weight parameter Ω.

Step 2: Given a query data $\mathbf{x}_q$, compute the distance and angle measures as follows:

$$d_i = \| \mathbf{x}_q - \mathbf{x}_i \|_2 \tag{2.17}$$

$$\cos(\vartheta_i) = \frac{\Delta \mathbf{x}_q^T \Delta \mathbf{x}_i}{\| \Delta \mathbf{x}_q \|_2 \ \| \Delta \mathbf{x}_i \|_2} \tag{2.18}$$

where $\Delta \mathbf{x}_i = \mathbf{x}_i - \mathbf{x}_{i-1}$ and $\Delta \mathbf{x}_q = \mathbf{x}_q - \mathbf{x}_{q-1}$.

If $\cos(\vartheta_i) \geq 0$, compute the similarity number s_i :

$$s_i = \Omega \cdot \sqrt{e^{-d_i^2}} + (1-\Omega) \cdot \cos(\vartheta_i) \tag{2.19}$$

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

Step 3: 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 $\Phi_l \in R^{1\times n}$, are constructed by selecting l most relevant data $\{(y_i, \mathbf{x}_i)\}$ corresponding to the largest s_i to the l-th largest s_i. Denote $\mathbf{W}_l \in R^{l\times l}$ a diagonal matrix with diagonal elements being the first l largest s_i, and calculate:

$$\mathbf{P}_l = \mathbf{W}_l \Phi_l \tag{2.20}$$

$$\mathbf{v}_l = \mathbf{W}_l \mathbf{y}_l \tag{2.21}$$

The local model parameters are then computed by:

$$\boldsymbol{\mu}_l = (\mathbf{P}_l^T \mathbf{P}_l)^{-1} \mathbf{P}_l^T \mathbf{v}_l \tag{2.22}$$

Next, the leave-one-out cross validation test is conducted and the validation error is calculated by (Myers, 1990):

$$e_l = \frac{1}{\sum_{j=1}^{l} s_j^2} \sum_{j=1}^{l} \left(s_j \frac{y_j - \phi_j^T (\mathbf{P}_l^T \mathbf{P}_l)^{-1} \mathbf{P}_l^T \mathbf{v}_l}{1 - \mathbf{p}_j^T (\mathbf{P}_l^T \mathbf{P}_l)^{-1} \mathbf{p}_j} \right)^2 \tag{2.23}$$

where y_j is the j-th element of $\mathbf{y}_l$, ϕ_j^T and $\mathbf{p}_j^T$ are the j-th row vector of Φ_l and $\mathbf{P}_l$, respectively.

Step 4: According to validation errors, the optimal l is determined by:

$$l_{opt} = \arg\operatorname{Min}_{l} (e_l) \tag{2.24}$$

Step 5: Verify the stability of local model built by the optimal model parameters $\boldsymbol{\mu}_{l_{opt}}$. Because both first-order and second-order models are adequate to describe process dynamics by using JITL algorithm, their respective stability constrains are given as follows:

First-order model:

$$-1 < \hat{\mu}_1 < 1 \tag{2.25}$$

Second-order model:

$$\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \hat{\mu}_1 \\ \hat{\mu}_2 \end{bmatrix} < \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{2.26}$$

$$-1 < \hat{\mu}_2 < 1 \tag{2.27}$$

If $\boldsymbol{\mu}_{l_{opt}}$ satisfies the stability constraint, the predicted output for query data is computed as:

$$(\hat{y}_q)_{l_{opt}} = \mathbf{x}_q^T \boldsymbol{\mu}_{l_{opt}} \tag{2.28}$$

Otherwise, $\boldsymbol{\mu}_{l_{opt}}$ is used as the initial value in the following optimization problem subject to appropriate stability constraint.

$$\underset{\mu}{\text{Min}} \| \mathbf{P}_{l_{opt}} \boldsymbol{\mu} - \mathbf{v}_{l_{opt}} \|_2 \tag{2.29}$$

With the optimal solution $\boldsymbol{\mu}^*_{l_{opt}}$ obtained from Eq. (2.29), the predicted output for query data is then calculated as $\mathbf{x}_q^T \boldsymbol{\mu}^*_{l_{opt}}$.

Step 6: When the next query data comes, go to step 2.

2.3 Adaptive Control

Adaptive control as depicted in Figure 2.2 covers a set of techniques for automatic adjustment of controller parameters in real time in order to achieve or to maintain a desired level for the performance of control systems when the dynamic parameters of the process are unknown or vary in time. Three schemes for adaptive control are gain scheduling, model reference control, and self-tuning regulators. The key problem is to find a convenient way of changing the regulator parameters in response to change in process and disturbance dynamics. The schemes differ only in the way the parameters of the regulator are adjusted. Gain scheduling has been successfully applied to problems in chemical process control (Astrom and Wittenmark, 1989). It is one of most widely and successfully applied techniques for the design of nonlinear controller. One drawback of gain scheduling is that it is open-loop compensation. There is no feedback which compensates for an incorrect schedule. Another drawback of gain scheduling is that the design is time consuming. A further major difficulty in the gain scheduling approach is the selection of appropriate scheduling variables. Model reference control is another way

to adjust the parameters of the regulator. The specifications are given in terms of a reference model which tells how the process output ideally should respond to the command signal. A third method for adjusting the regulator parameters is to use the self-tuning regulator (Astrom, 1983). Model identification adaptive controllers are sometimes also called self-optimizing controllers or self-tuning controllers. They perform three basic tasks: information gathering of the present process behavior; control performance criterion optimization; and adjustment of the controller. Information gathering of the process implies the continuous determination of the actual condition of the process to be controlled based on measurable process input and output. Suitable ways are identification and parameter estimation of process model. Various types of model identification adaptive controller can be distinguished, depending on the information gathered and the method of estimation. Performance criterion optimization implies the calculation of the

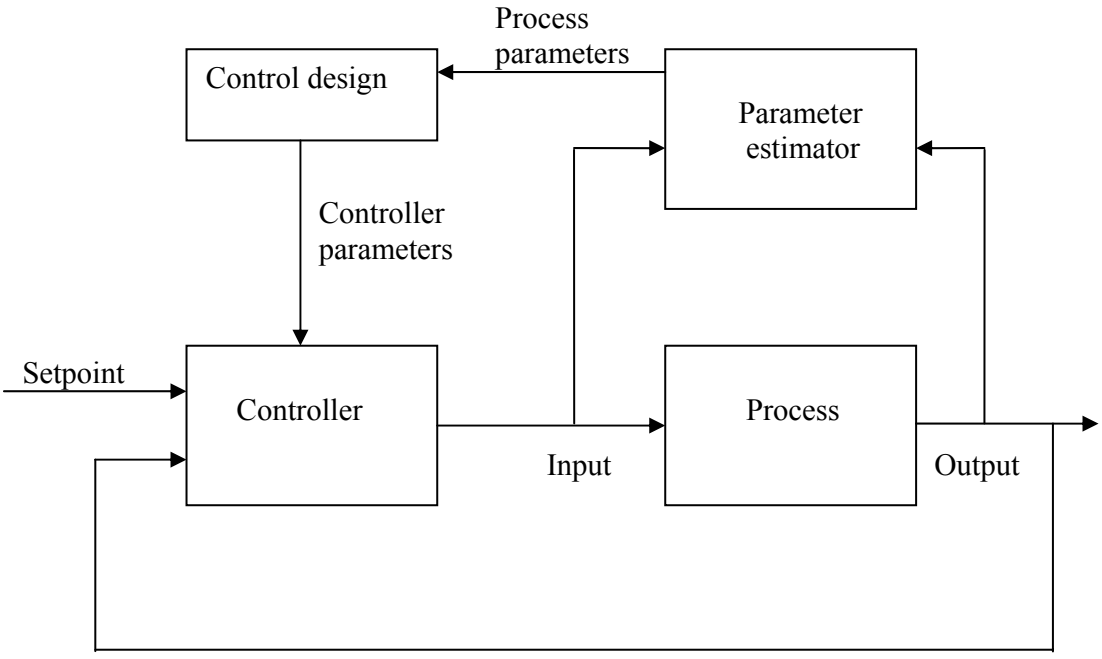


Figure 2.2 Adaptive control

control loop performance and the decision as to how the controller will be adjusted or adapted. Adjustment of the controller implies the calculation of the new controller parameter set and replacement of the old parameters in the control loop.

2.4 Internal Model Control

The Internal Model Control (IMC) design procedure (Morari and Zafiriou, 1989) utilizes the structure shown in Figure 2.3, in which G represents the process, $\widetilde{G}$ represents a model of the process, and Q represents the IMC controller. The effect of the parallel path with the model is to subtract the effect of the manipulated variables from the process output. If the model is perfect representation of the process, then feedback is equal to the influence of disturbances and is not affected by the action of the manipulated variables. Thus, the system is effectively open-loop and the usual stability problems associated with feedback have disappeared. The overall system is stable simply if and only if both the process and IMC controller are stable.

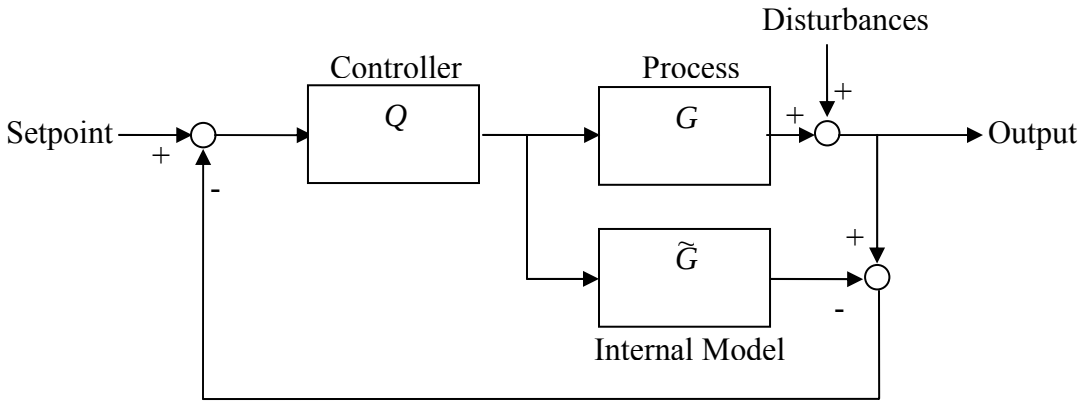


Figure 2.3 Internal model control

The IMC controller Q can be designed by the following equation:

$$Q = \tilde{G}_{-}^{-1} f \tag{2.30}$$

where $\tilde{G}_{-}$ is the minimum phase part of $\tilde{G}$ and f is a low-pass filter:

$$f = \frac{1}{(\tau s + 1)^r} \tag{2.31}$$

where τ is the desired closed-loop time constant and the parameter r is a positive integer that is selected so that Q is either a strictly proper or proper transfer function.

2.5 Decentralized Control

The decentralized control structure as shown in Figure 2.4 have been commonly used in the chemical process industries. The advantage is that fewer controller parameters need to be chosen than those for a centralized controller. This is particularly relevant in process control where often thousands of variables have to be controlled, which could lead to an enormously complex controller. It is also important that stability as well as performance is preserved to some degree when individual sensors or actuators fail. This failure tolerance is generally easier to achieve with decentralized control systems, where parts can be turned off without significantly affecting the rest of the system (Morari and Zafiriou, 1989).

It is evident that the aforementioned adaptation procedure can be applied to the decentralized control scheme as well. An adaptive decentralized control system based on the on-line adaptation of PID parameters will be developed in this research.

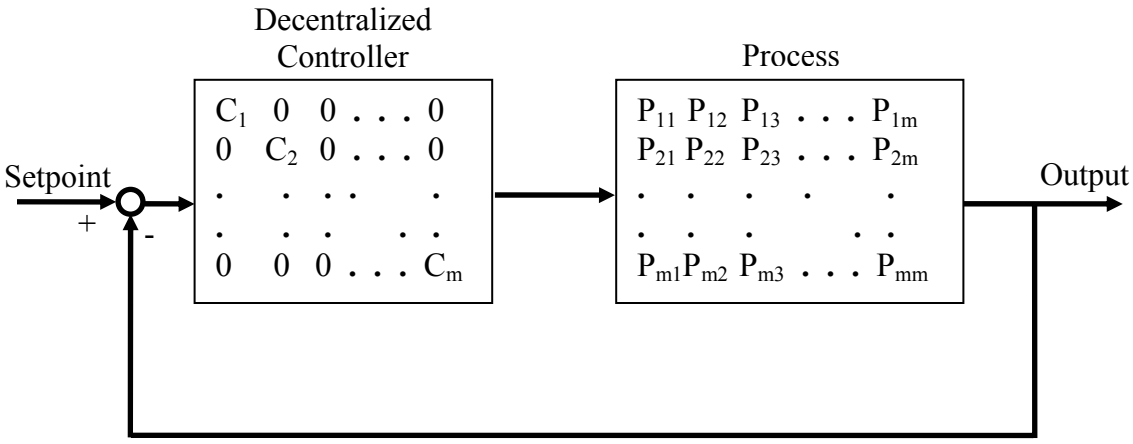


Figure 2.4 Decentralized control system

CHAPTER 3

Identification of Generalized Hammerstein Model

3.1 Introduction

Hammerstein model structure can effectively represent and approximate many industrial processes. For example, the nonlinear dynamics of chemical processes, such as pH neutralization processes (Lakshminarayanan et al., 1995; Fruzzetti et al., 1997), distillation columns (Eskinat et al., 1991; Pearson and Pottmann, 2000), heat exchangers (Eskinat et al., 1991; Lakshminarayanan et al., 1995) and polymerization reactor (Su and McAvoy, 1993; Ling and Rivera, 1998), have been modeled with Hammerstein model. However, Hammerstein model is restricted to the situations where the process gain changes with the operating conditions while the linear systems remain fairly constant over the operating space under consideration. As a result, the conventional Hammerstein model is not adequate for modeling the process when both the process gain and linear dynamics change over the region of plant operation (Lakshminarayanan et al., 1997). To overcome this drawback, a generalized Hammerstein model that consists of varying linear dynamics preceded by a static nonlinear part is proposed and its associated identification problem is considered in this chapter.

Obviously, the aforementioned generalized Hammerstein model is an extension of the conventional Hammerstein model by replacing the fixed linear model by the time-varying linear models. Thus, the generalized Hammerstein model is expressed by:

$$y(k) = \alpha_1^k y(k-1) + \ldots + \alpha_{n_y}^k y(k-n_y) + \beta_1^k v(k-1-n_d) + \ldots + \beta_{n_v}^k v(k-n_v-n_d) \quad (3.1)$$

$$v(k) = \gamma_1 u(k) + \gamma_2 u^2(k) + \ldots + \gamma_m u^m(k) \quad (3.2)$$

where $y(k)$ and $u(k)$ denote the process output and input at the k-th sampling instant respectively, $v(k)$ is unmeasurable internal variable, α_i^k $(i = 1 \sim n_y)$ and β_i^k $(i = 1 \sim n_v)$ are the parameters of linear dynamics at the k-th sampling instant, γ_i $(i = 1 \sim m)$ are the parameters of static nonlinear part, n_y and n_v are integers related to the model order, and n_d denotes the process time-delay.

Motivated by the Narendra-Gallman method (1966), an iterative procedure by incorporating JITL algorithm is developed to identify SISO and MIMO generalized Hammerstein models in the next two sections.

3.2 Identification of SISO Generalized Hammerstein Model

During the off-line identification phase, a dataset consisting of N process data $\{y(k), u(k)\}_{k=1\sim N}$ is collected. Because JITL is employed to identify the time-varying models in the proposed method, a low-order model ($n_y \le 2$ and $n_v \le 2$) is adequate to describe the linear dynamics of generalized Hammerstein model. Thus the generalized Hammerstein model to be identified by the proposed identification procedure has the following form:

$$y(k) = \alpha_1^k y(k-1) + \alpha_2^k y(k-2) + \beta_1^k v(k-1-n_d) + \beta_2^k v(k-2-n_d) \quad (3.3)$$

$$v(k) = \gamma_1 u(k) + \gamma_2 u^2(k) + \ldots + \gamma_m u^m(k) \tag{3.4}$$

The proposed iterative identification procedure obtains the parameters of the generalized Hammerstein model by separating the estimation problem of the static nonlinear part from that of the linear dynamics. When the parameters of the linear dynamics are available, the parameters of static nonlinear part are obtained by solving the following objective function:

$$\underset{\hat{\gamma}_1,\hat{\gamma}_2,\ldots,\hat{\gamma}_m}{\text{Min}} E(\hat{\gamma}_1,\hat{\gamma}_2,\ldots,\hat{\gamma}_m) = \frac{1}{N}\sum_{k=1}^{N}(y(k) - \hat{y}(k;\hat{\gamma}_1,\hat{\gamma}_2,\ldots,\hat{\gamma}_m))^2 \tag{3.5}$$

where $\hat{y}(k)$ is the predicted output of generalized Hammerstein model:

$$\hat{y}(k;\hat{\gamma}_1,\hat{\gamma}_2,\ldots,\hat{\gamma}_m) = \frac{\hat{B}^k(q^{-1})}{\hat{A}^k(q^{-1})}\sum_{j=1}^{m}\hat{\gamma}_j u^j(k) \tag{3.6}$$

$$\hat{A}^k(q^{-1}) = 1 - \hat{\alpha}_1^k q^{-1} - \hat{\alpha}_2^k q^{-2}, \quad \hat{B}^k(q^{-1}) = \hat{\beta}_1^k q^{-1-n_d} + \hat{\beta}_2^k q^{-2-n_d} \tag{3.7}$$

where $\hat{\alpha}_1^k, \hat{\alpha}_2^k, \hat{\beta}_1^k, \hat{\beta}_2^k$ are the known linear model parameters and $\hat{\gamma}_i$ $(i = 1 \sim m)$ are the nonlinear parameters to be determined.

By differentiating the objective function E with respect to $\hat{\gamma}_i$ obtains:

$$\frac{\partial E}{\partial \hat{\gamma}_1} = -\frac{2}{N}\sum_{k=1}^{N}\left(y(k) - \hat{y}(k;\hat{\gamma}_1,\hat{\gamma}_2,\ldots,\hat{\gamma}_m)\right)\times\ b_1^k \tag{3.8}$$

$$\frac{\partial E}{\partial \hat{\gamma}_2} = -\frac{2}{N}\sum_{k=1}^{N}\left(y(k) - \hat{y}(k;\hat{\gamma}_1,\hat{\gamma}_2,\ldots,\hat{\gamma}_m)\right)\times b_2^k \tag{3.9}$$

$$\vdots$$

$$\frac{\partial E}{\partial \hat{\gamma}_m} = -\frac{2}{N}\sum_{k=1}^{N}\left(y(k) - \hat{y}(k;\hat{\gamma}_1,\hat{\gamma}_2,\ldots,\hat{\gamma}_m)\right)\times b_m^k \tag{3.10}$$

where $b_j^k = \hat{\beta}_1^k u^j(k-1) + \hat{\beta}_2^k u^j(k-2), \ \ j = 1 \sim m, \ \ k = 1 \sim N.$

By setting Eqs. (3.8) to (3.10) to zero, the nonlinear parameters are solved by:

$$\left[\hat{\gamma}_1, \hat{\gamma}_2, \ldots, \hat{\gamma}_m\right]^T = \mathbf{A}^{-1}\left[c_1, c_2, \ldots, c_m\right]^T \tag{3.11}$$

where

$$\mathbf{A} = \begin{bmatrix} \sum_{k=1}^{N} b_1^k \times b_1^k & \sum_{k=1}^{N} b_2^k \times b_1^k & \cdots & \sum_{k=1}^{N} b_m^k \times b_1^k \\ \sum_{k=1}^{N} b_1^k \times b_2^k & \sum_{k=1}^{N} b_2^k \times b_2^k & \cdots & \sum_{k=1}^{N} b_m^k \times b_2^k \\ \vdots & \vdots & \cdots & \vdots \\ \sum_{k=1}^{N} b_1^k \times b_m^k & \sum_{k=1}^{N} b_2^k \times b_m^k & \cdots & \sum_{k=1}^{N} b_m^k \times b_m^k \end{bmatrix} \tag{3.12}$$

$$c_j = \sum_{k=1}^{N} \left(y(k) - \hat{\alpha}_1^k y(k-1) - \hat{\alpha}_2^k y(k-2) \right) \times b^k{}_j, \quad j = 1 \sim m \tag{3.13}$$

On the other hand, when the nonlinear parameters $\hat{\gamma}_i$ ($i = 1 \sim m$) are known, the intermediate variable $v(k)$ can be obtained from Eq. (3.4). Therefore, the dataset $\{(y(k), \mathbf{x}(k))\}_{k=1\sim N}$ where $\mathbf{x}(k)$ is the regression vector pertaining to the local model chosen for the JITL algorithm can be constructed. For example, $\mathbf{x}(k) = \{y(k-1), v(k-1)\}$ for a first-order linear model, i.e. $n_y = n_v = 1$ and $n_d = 0$. Using $\{(y(k), \mathbf{x}(k))\}_{k=1\sim N}$ as the reference dataset, the parameters of N local models corresponding to N query data $\{y(k), v(k)\}$, i.e. $\hat{\alpha}_i^k$ and $\hat{\beta}_j^k$ (i,j = 1 or 2), can be obtained by using JITL algorithm given in Chapter 2.

The following summarizes the proposed off-line iterative identification procedure for the generalized Hammerstein model:

1. Given the data set $\{y(k), u(k)\}_{k=1\sim N}$ and the parameters of static nonlinear part are initialized as $\hat{\gamma}_1 = 1$ and $\hat{\gamma}_i = 0$ $(i \neq 1)$;

2. Compute $v(k)$ from Eq. (3.4) and construct the reference dataset $\{(y(k), \mathbf{x}(k))\}_{k=1\sim N}$ for JITL algorithm, followed by the computation of the parameters of a set of linear models, $\hat{\alpha}_i^k$ and $\hat{\beta}_j^k$ $(i, j = 1 \text{or } 2,\ k = 1 \sim N)$, by using the JITL algorithm;
3. The parameters of the static nonlinear part are calculated by using Eq. (3.11) and the result obtained in step 2;
4. When the convergence criterion is met, stop; otherwise, go to step 2 by using the updated parameters $\hat{\gamma}_i$ obtained in step 3.

To conclude this section, it is worth pointing out one major difference in the identification and application of the conventional and generalized Hammerstein models. In the former case, both static nonlinear part and linear model obtained during the off-line identification phase naturally complete the construction of Hammerstein model and are subsequently used in the on-line application of such a model, e.g. model-based controller design. In contrast, only the parameters of static nonlinear part of generalized Hammerstein model obtained in the off-line identification procedure are fixed as part of the model parameters, while those of linear dynamics are calculated at the instant when model prediction is required. This main departure from the conventional Hammerstein model is due to the time-varying linear models employed in the generalized Hammerstein model. As a result, only the most up-to-data linear model relevant to the current process data will be computed at each sampling instant by the JITL algorithm for modeling and controller design purposes, after which these model parameters will then be discarded.

The following summarizes how to calculate the predicted output of generalized Hammerstein model:

1. Given the identical dataset $\{y(k),u(k)\}_{k=1\sim N}$ previously obtained in the off-line identification phase and the static nonlinear part obtained by the aforementioned iterative identification procedure;
2. Compute $v(k)$ from Eq. (3.4) and construct the reference dataset $\{(y(k),\mathbf{x}(k))\}_{k=1\sim N}$ for the JITL algorithm;
3. Given the on-line process data $\{y_p(j),u_p(j)\}$ at the j-th sampling instant, compute $v_p(j)$ from Eq. (3.4) and subsequently obtain the predicted output $\hat{y}_p(j+1)$ of generalized Hammerstein model by the JITL algorithm.

3.3 Identification of MIMO Generalized Hammerstein Model

Two possible structures as depicted Figures 3.1 and 3.2 can be used to describe a MIMO Hammerstein model depending on whether the nonlinearities are separate or combined (Lakshminnarayana et al., 1995; Al-Duwaish and Karim, 1997). The combined nonlinearity case is more general, but it can cause a very challenging parameter estimation problem because of the large number of parameters to be estimated. Therefore, the MIMO generalized Hammerstein model with separate nonlinearities will be considered in this research.

Without loss of generality, a multivariable process with two inputs and two outputs will be utilized to detail the proposed identification procedure. For a 2×2 generalized Hammerstein model with separate nonlinearities, it can be described by the following equation:

$$\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} \alpha_{11}^k & \alpha_{12}^k \\ \alpha_{21}^k & \alpha_{22}^k \end{bmatrix} \begin{bmatrix} y_1(k-1) \\ y_2(k-1) \end{bmatrix} + \begin{bmatrix} \beta_1^k & 0 \\ 0 & \beta_2^k \end{bmatrix} \begin{bmatrix} v_1(k-1) \\ v_2(k-1) \end{bmatrix} \quad (3.14)$$

where $\alpha_{11}^k, \alpha_{12}^k, \alpha_{21}^k, \alpha_{22}^k, \beta_1^k$ and β_2^k are the parameters of linear dynamics of MIMO Hammerstein model at the *k*-th sampling instant and the nonlinearities are represented by:

$$v_1(k) = \gamma_{11} u_1(k) + \gamma_{12} u_1^2(k) + \cdots + \gamma_{1m_1} u_1^{m_1}(k) \quad (3.15)$$

$$v_2(k) = \gamma_{21} u_2(k) + \gamma_{22} u_2^2(k) + \cdots + \gamma_{2m_2} u_2^{m_2}(k) \quad (3.16)$$

where $\gamma_{1i} \quad (i = 1 \sim m_1)$, $\gamma_{2j} \quad (j = 1 \sim m_2)$, m_1 and m_2 are parameters of static nonlinearities.

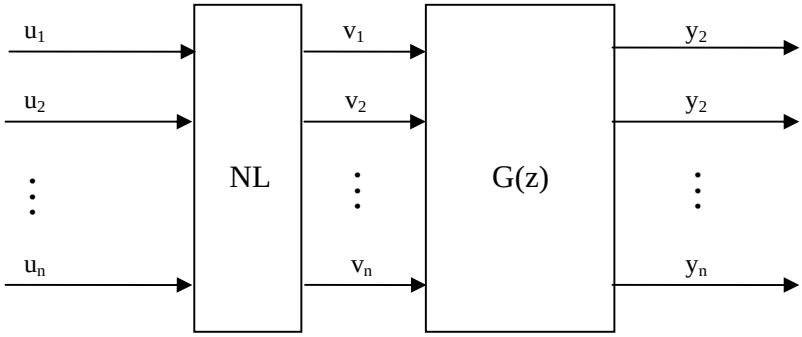


Figure 3.1. MIMO Hammerstein model with combined non-linearities.

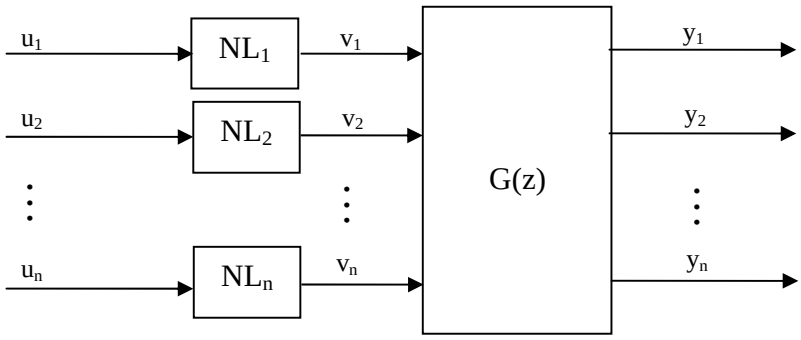


Figure 3.2. MIMO Hammerstein model with separate non-linearities.

Equations (3.14) to (3.16) can be rewritten as follows:

$$\begin{aligned} y_1(k) &= \alpha_{11}^k y_1(k-1) + \alpha_{12}^k y_2(k-1) + \beta_1^k v_1(k-1) \\ v_1(k) &= \gamma_{11} u_1(k) + \gamma_{12} u_1^2(k) + \cdots + \gamma_{1m_1} u_1^{m_1}(k) \end{aligned} \tag{3.17}$$

$$\begin{aligned} y_2(k) &= \alpha_{21}^k y_1(k-1) + \alpha_{22}^k y_2(k-1) + \beta_2^k v_2(k-1) \\ v_2(k) &= \gamma_{21} u_2(k) + \gamma_{22} u_2^2(k) + \cdots + \gamma_{2m_2} u_2^{m_2}(k) \end{aligned} \tag{3.18}$$

As can be seen from Eqs. (3.17) and (3.18), identification of a 2×2 generalized Hammerstein model is reduced to the identification of two individual SISO generalized Hammerstein models. Therefore, the iterative identification procedure developed in section 3.2 can be extended in a straight forward manner to identify the 2×2 generalized Hammerstein model as described by Eqs (3.17) and (3.18), as will be discussed in detail in what follows.

Given the process data $\{y_1(k), y_2(k), u_1(k), u_2(k)\}_{k=1\sim N}$ and parameters of the linear dynamics in Eq. (3.17), the parameters of static nonlinear part in Eq. (3.17) are obtained by solving the following objective function:

$$\underset{\hat{\gamma}_{11},\hat{\gamma}_{12},\cdots,\hat{\gamma}_{1m_1}}{\text{Min}} E_1(\hat{\gamma}_{11},\hat{\gamma}_{12},\cdots,\hat{\gamma}_{1m_1}) = \frac{1}{N}\sum_{k=1}^{N}(y_1(k) - \hat{y}_1(k;\hat{\gamma}_{11},\hat{\gamma}_{12},\cdots,\hat{\gamma}_{1m_1}))^2 \tag{3.19}$$

where $\hat{y}_1(k)$ is the first predicted output of generalized Hammerstein model:

$$\begin{aligned} \hat{y}_1(k) &= \hat{\alpha}_{11}^k y_1(k-1) + \hat{\alpha}_{12}^k y_2(k-1) + \hat{\beta}_1^k \hat{v}_1(k-1) \\ \hat{v}_1(k) &= \hat{\gamma}_{11} u_1(k) + \hat{\gamma}_{12} u_1^2(k) + \cdots + \hat{\gamma}_{1m_1} u_1^{m_1}(k) \end{aligned} \tag{3.20}$$

where $\hat{\alpha}_{11}^k, \hat{\alpha}_{12}^k, \hat{\beta}_1^k$ are the known linear model parameters and $\hat{\gamma}_{1i}$ $(i = 1\sim m_1)$ are the nonlinear parameters to be determined.

By differentiating the objective function E_1 with respect to $\hat{\gamma}_{1i}$ obtains:

$$\frac{\partial E_1}{\partial \hat{\gamma}_{11}} = -\frac{2}{N}\sum_{k=1}^{N}\left(y_1(k) - \hat{y}_1(k;\hat{\gamma}_{11},\hat{\gamma}_{12},\cdots,\hat{\gamma}_{1m_1})\right)\times\ b_{11}^k \tag{3.21}$$

$$\frac{\partial E_1}{\partial \hat{\gamma}_{12}} = -\frac{2}{N}\sum_{k=1}^{N}\left(y_1(k) - \hat{y}_1(k;\hat{\gamma}_{11},\hat{\gamma}_{12},\cdots,\hat{\gamma}_{1m_1})\right)\times b_{12}^k \tag{3.22}$$

$$\vdots$$

$$\frac{\partial E_1}{\partial \hat{\gamma}_{1m_1}} = -\frac{2}{N}\sum_{k=1}^{N}\left(y_1(k) - \hat{y}_1(k;\hat{\gamma}_{11},\hat{\gamma}_{12},\cdots,\hat{\gamma}_{1m_1})\right)\times\ b_{1m_1}^k \tag{3.23}$$

where $b_{1i}^k = \hat{\beta}_1^k u_1^i(k-1)$, $i = 1 \sim m_1$, $k = 1 \sim N$.

By setting Eqs. (3.21) to (3.23) to zero, the nonlinear parameters are solved by:

$$\left[\hat{\gamma}_{11},\hat{\gamma}_{12},\cdots,\hat{\gamma}_{1m_1}\right]^T = \mathbf{A}_1^{-1}\left[c_{11},c_{12},\ldots,c_{1m_1}\right]^T \tag{3.24}$$

where

$$\mathbf{A}_1 = \begin{bmatrix} \sum_{k=1}^{N} b_{11}^k \times b_{11}^k & \sum_{k=1}^{N} b_{12}^k \times b_{11}^k & \cdots & \sum_{k=1}^{N} b_{1m_1}^k \times b_{11}^k \\ \sum_{k=1}^{N} b_{11}^k \times b_{12}^k & \sum_{k=1}^{N} b_{12}^k \times b_{12}^k & \cdots & \sum_{k=1}^{N} b_{1m_1}^k \times b_{12}^k \\ \vdots & \vdots & \cdots & \vdots \\ \sum_{k=1}^{N} b_{11}^k \times b_{1m_1}^k & \sum_{k=1}^{N} b_{12}^k \times b_{1m_1}^k & \cdots & \sum_{k=1}^{N} b_{1m_1}^k \times b_{1m_1}^k \end{bmatrix} \tag{3.25}$$

and

$$c_{1i} = \sum_{k=1}^{N}\left(y_1(k) - \hat{\alpha}_{11}^k y_1(k-1) - \hat{\alpha}_{12}^k y_2(k-1)\right)\times b_{1i}^k,\ i = 1 \sim m_1 \tag{3.26}$$

Similarly, when parameters of the linear dynamics in Eq. (3.18) are available, the parameters of static nonlinear part in Eq. (3.18) are obtained by solving the following objective function:

$$\underset{\hat{\gamma}_{21},\hat{\gamma}_{22},\cdots,\hat{\gamma}_{2m_2}}{\text{Min}}\ E_2(\hat{\gamma}_{21},\hat{\gamma}_{22},\cdots,\hat{\gamma}_{2m_2}) = \frac{1}{N}\sum_{k=1}^{N}(y_2(k) - \hat{y}_2(k;\hat{\gamma}_{21},\hat{\gamma}_{22},\cdots,\hat{\gamma}_{2\mathrm{m}_2}))^2 \tag{3.27}$$

where $\hat{y}_2(k)$ is the second predicted output of generalized Hammerstein model:

$$\begin{aligned}\hat{y}_2(k) &= \hat{\alpha}_{21}^k y_2(k-1) + \hat{\alpha}_{22}^k y_2(k-1) + \hat{\beta}_2^k \hat{v}_2(k-1)\\ \hat{v}_2(k) &= \hat{\gamma}_{21} u_2(k) + \hat{\gamma}_{22} u_2^2(k) + \cdots + \hat{\gamma}_{2m_2} u_2^{m_2}(k)\end{aligned} \tag{3.28}$$

where $\hat{\alpha}_{21}^k, \hat{\alpha}_{22}^k$, $\hat{\beta}_2^k$ are the known linear model parameters and $\hat{\gamma}_{2j}$ $(j = 1 \sim m_2)$ are the nonlinear parameters to be determined.

By differentiating the objective function E_2 with respect to $\hat{\gamma}_{2j}$ obtains:

$$\frac{\partial E_2}{\partial \hat{\gamma}_{21}} = -\frac{2}{N}\sum_{k=1}^{N}\left(y_2(k) - \hat{y}_2(k;\hat{\gamma}_{21},\hat{\gamma}_{22},\cdots,\hat{\gamma}_{2m_2})\right)\times\ b_{21}^k \tag{3.29}$$

$$\frac{\partial E_2}{\partial \hat{\gamma}_{22}} = -\frac{2}{N}\sum_{k=1}^{N}\left(y_2(k) - \hat{y}_2(k;\hat{\gamma}_{21},\hat{\gamma}_{22},\cdots,\hat{\gamma}_{2m_2})\right)\times b_{22}^k \tag{3.30}$$

$$\vdots$$

$$\frac{\partial E_2}{\partial \hat{\gamma}_{2m_2}} = -\frac{2}{N}\sum_{k=1}^{N}\left(y_2(k) - \hat{y}_2(k;\hat{\gamma}_{21},\hat{\gamma}_{22},\cdots,\hat{\gamma}_{2m_2})\right)\times\ b_{2m_2}^k \tag{3.31}$$

where $b_{2j}^k = \hat{\beta}_2^k u_2^j(k-1)$, $j = 1 \sim m_2$, $k = 1 \sim N$.

By setting Eqs. (3.29) to (3.31) to zero, the nonlinear parameters are solved by:

$$\left[\hat{\gamma}_{21},\hat{\gamma}_{22},\cdots,\hat{\gamma}_{2m_2}\right]^T = \mathbf{A}_2^{-1}\left[c_{21},c_{22},\ldots,c_{2m_2}\right]^T \tag{3.32}$$

where

$$\mathbf{A}_2 = \begin{bmatrix} \sum_{k=1}^{N} b_{21}^k \times b_{21}^k & \sum_{k=1}^{N} b_{22}^k \times b_{21}^k & \cdots & \sum_{k=1}^{N} b_{2m_2}^k \times b_{21}^k \\ \sum_{k=1}^{N} b_{21}^k \times b_{22}^k & \sum_{k=1}^{N} b_{22}^k \times b_{22}^k & \cdots & \sum_{k=1}^{N} b_{2m_2}^k \times b_{22}^k \\ \vdots & \vdots & \cdots & \vdots \\ \sum_{k=1}^{N} b_{21}^k \times b_{2m_2}^k & \sum_{k=1}^{N} b_{22}^k \times b_{2m_2}^k & \cdots & \sum_{k=1}^{N} b_{2m_2}^k \times b_{2m_2}^k \end{bmatrix} \tag{3.33}$$

and

$$c_{2j} = \sum_{k=1}^{N}\left(y_2(k) - \hat{\alpha}_{21}^k y_1(k-1) - \hat{\alpha}_{22}^k y_2(k-1)\right)\times b_{2j}^k, \quad j = 1 \sim m_2 \tag{3.34}$$

On the other hand, given the parameters of static nonlinear parts, $\hat{\gamma}_{1i}$ $(i=1\sim m_1)$ and $\hat{\gamma}_{2j}$ $(j=1\sim m_2)$, both intermediate variables $v_1(k)$ and $v_2(k)$ can be obtained from Eqs. (3.17) and (3.18). Subsequently, two reference datasets $\{(y_1(k),\mathbf{x}_1(k))\}_{k=1\sim N}$ and $\{(y_2(k),\mathbf{x}_2(k))\}_{k=1\sim N}$ where $\mathbf{x}_i(k)=\{y_1(k),y_2(k),v_i(k)\}$ $(i=1,2)$ are constructed for JITL algorithm. Consequently, the parameters of $2\times N$ local models corresponding to *N* query data for predicting y_1 and y_2 can be obtained.

To conclude this section, the following summarizes the proposed off-line iterative identification for a 2×2 generalized Hammerstein model:

1. Given the data set $\{y_1(k),y_2(k),u_1(k),u_2(k)\}_{k=1\sim N}$, the parameters of static nonlinear parts are initialized as zero except that $\hat{\gamma}_{11}=\hat{\gamma}_{21}=1$;
2. Compute $v_1(k)$ and $v_2(k)$ from Eqs. (3.17) and (3.18) and construct the reference datasets $\{(y_1(k),\mathbf{x}_1(k))\}_{k=1\sim N}$ and $\{(y_2(k),\mathbf{x}_2(k))\}_{k=1\sim N}$ for JITL algorithm, followed by the computation of the parameters of linear models, $\hat{\alpha}_{11}^k,\hat{\alpha}_{12}^k$, $\hat{\alpha}_{21}^k,\hat{\alpha}_{22}^k,\hat{\beta}_1^k$ and $\hat{\beta}_2^k$, by using the JITL algorithm;
3. The parameters of the static nonlinear parts are calculated by using Eqs. (3.24) and (3.32) and the result obtained in step 2;
4. When the convergence criterion is met, stop; otherwise, go to step 2 by using the updated parameters $\hat{\gamma}_{1i}(i=1\sim m_1)$ and $\hat{\gamma}_{2j}(j=1\sim m_2)$ obtained in step 3.

With the identification result obtained above, the predicted outputs of the 2×2 generalized Hammerstein model is obtained as follows:

1. Given the identical dataset $\{y_1(k),y_2(k),u_1(k),u_2(k)\}_{k=1\sim N}$ and the static nonlinear parameters obtained in the aforementioned iterative identification procedure;

2. Compute $v_1(k)$ and $v_2(k)$ from Eqs. (3.17) and (3.18) and construct the reference datasets $\{(y_1(k), \mathbf{x}_1(k))\}_{k=1\sim N}$ and $\{(y_2(k), \mathbf{x}_2(k))\}_{k=1\sim N}$ for JITL algorithm;
3. Given the on-line process data $\{y_{p,1}(j), y_{p,2}(j), u_{p,1}(j), u_{p,2}(j)\}$ at the *j*-th sampling instant, compute $v_{p,1}(j)$ and $v_{p,2}(j)$ from Eqs. (3.17) and (3.18) and subsequently obtain the predicted outputs of generalized Hammerstein model by using $\{y_{p,1}(j), y_{p,2}(j), v_{p,1}(j)\}$ and $\{y_{p,1}(j), y_{p,2}(j), v_{p,2}(j)\}$ as the query data for JITL algorithm, respectively.

3.4 Examples

Example 1 Consider the free radical polymerization of methyl methacrylate in the CSTR reactor with azo-bis-isobutyronitrile as initiator and toluene as solvent. This process was modeled as a Hammerstein model in the previous study by Ling and Rivera (1998). The model of this process is represented by the following equations (Doyle et al., 1995; Ling and Rivera, 1998; Harris and Palazoglu, 1998):

$$\dot{x}_1 = -k_1 x_1 \sqrt{x_2} - \frac{F x_1}{V} + \frac{F C_{m,in}}{V} \tag{3.35}$$

$$\dot{x}_2 = -k_2 x_2 - \frac{F x_2}{V} + \frac{F_I C_{I,in}}{V} \tag{3.36}$$

$$\dot{x}_3 = k_3 x_2 + k_4 x_1 \sqrt{x_2} - \frac{F x_3}{V} \tag{3.37}$$

$$\dot{x}_4 = M_m k_5 x_1 \sqrt{x_2} - \frac{F x_4}{V} \tag{3.38}$$

$$M_p = \frac{x_4}{x_3} \tag{3.39}$$

where the dimensionless state variables x_i $(i = 1 \sim 4)$ correspond to the concentration of the monomer, concentration of the initiator, molar concentration of the dead polymer chains, and mass concentration of the dead polymer chains, respectively, process output is the number average molecular weight M_p, and process input is the inlet initiator flow rate, F_I. The relevant model parameters and the nominal operating condition are given in Tables 3.1 and 3.2 respectively. The sample time is chosen as 0.05 hr and the operating space under consideration is $M_p \in [1\times10^4 \quad 4\times10^4]$.

Table 3.1 Model parameters for polymerization reactor

V	Reactor volume	0.1 m^3
F	Inlet flow rate of monomer	1.0 m^3
$C_{I,in}$	Inlet concentration of initiator	8.0 kmol/m^3
$C_{m,in}$	Inlet concentration of monomer	6.0 kmol/m^3
M_m	Molecular weight of monomer	100.12 kg/kmol
k_1	Kinetic parameter	2.457 $m^{3/2}$/$kmol^{1/2}$/hr
k_2	Kinetic parameter	0.102 L/hr
k_3	Kinetic parameter	0.122 L/hr
k_4	Kinetic parameter	2.412×10^{-3} $m^{3/2}$/$kmol^{1/2}$/hr
k_5	Kinetic parameter	2.4568 $m^{3/2}$/$kmol^{1/2}$/hr

Table 3.2 Nominal operating condition for polymerization reactor

$x_{1,0}$	5.507 kmol/m^3	$x_{4,0}$	49.38 kmol/m^3
$x_{2,0}$	0.133 kmol/m^3	$F_{I,0}$	0.01673 m^3/hr
$x_{3,0}$	1.975×10^{-3}kmol/m^3	$M_{p,0}$	25000.5

To identify the generalized Hammerstein model, eight hundred input-output data as shown in Figure 3.3 are collected for model identification and construction of database for JITL algorithm, where the parameters $k_{\min} = 25, k_{\max} = 90$ and Ω=0.95 are chosen in the simulation study. By using a third-order polynomial as static nonlinear part and a first-order linear model to construct the generalized Hammerstein model, the proposed iterative identification procedure obtains γ_1 = 0.9037, γ_2 = -0.2875, γ_3 = 0.0304. For comparison purpose, conventional Hammerstein model consisting of a third-order polynomial and first-order linear model is also identified by using the identical process data shown in Figure 3.3. Applying Narendra-Gallman method, the following model parameters are obtained α_1 = 0.7658, β_1 = -0.0436, γ_1 = 2.5848, γ_2 = -1.0734 and γ_3 = 0.1593.

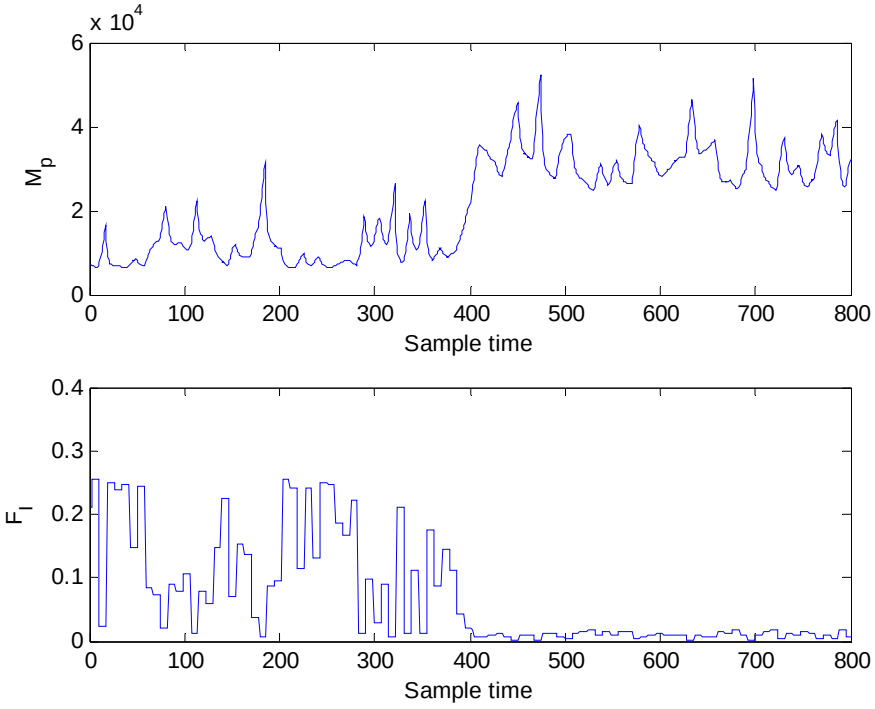


Figure 3.3 Input-output data for polymerization reactor

To evaluate the predictive performance of these two identified models, their respective predicted open-loop responses for 150% and -50% step changes in the initiator flow rate F_I are compared in Figure 3.4. The Mean Absolute Errors (MAE) of generalized Hammerstein model are 33.89% and 32.50% of those obtained by Hammerstein model, respectively. It is apparent that generalized Hammerstein model has better prediction accuracy than the conventional Hammerstein model.

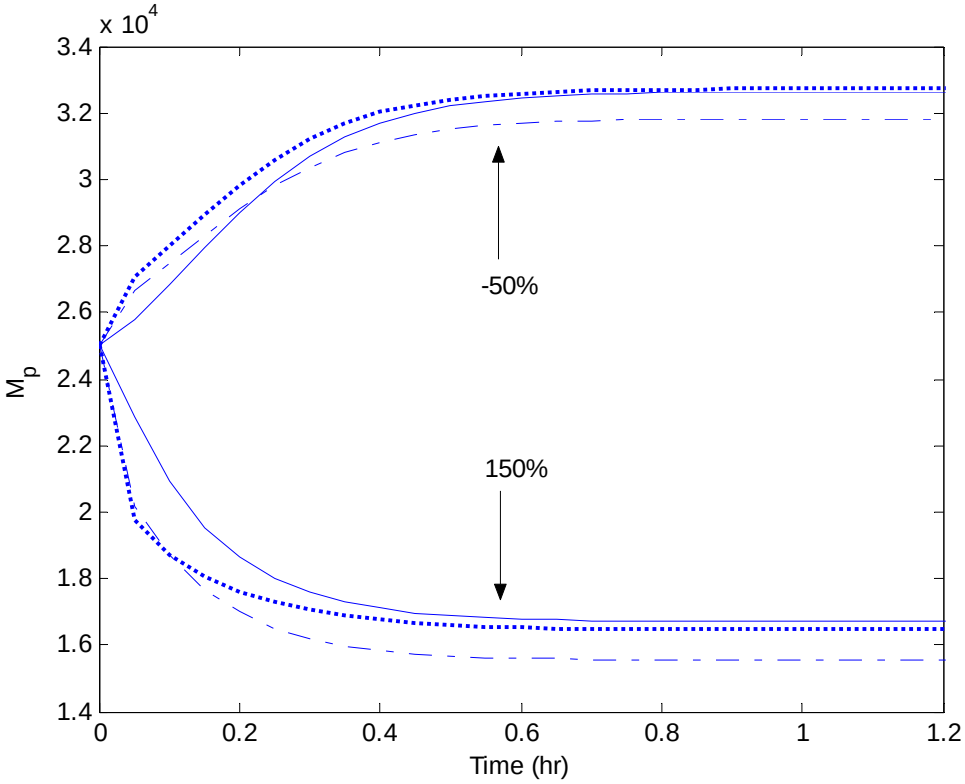


Figure 3.4 Open-loop response for 150% and -50% changes in F_I. Solid line: process; dotted line: generalized Hammerstein model; dash-dot line: Hammerstein model

Example 2 Consider the following van de Vusse reaction kinetic scheme:

$$\mathrm{A} \rightarrow \mathrm{B} \rightarrow \mathrm{C} \tag{3.40}$$

$$2\mathrm{A} \rightarrow \mathrm{D} \tag{3.41}$$

which is carried out in an isothermal CSTR. The mass balances for components A and B are given by (Doyle et al., 1995):

$$\dot{C}_A = -k_1 C_A - k_3 C_A^2 + \frac{F}{V}(C_{Af} - C_A) \tag{3.42}$$

$$\dot{C}_B = k_1 C_A - k_2 C_B - \frac{F}{V} C_B \tag{3.43}$$

where the concentration of component B, C_B, is the process output and the inlet flow rate, F, is the process input. The model parameters used in the simulation study are: k_1 = 50 hr^{-1}, k_2 = 100 hr^{-1}, k_3 = 10 L/(mol hr), C_{Af} = 10 mol/L and V = 1 L and the nominal operation condition is C_{Ao} = 3.0 mol/L, C_{B0} = 1.12 mol/L and F_o = 34.3 L/hr. This process was previously described as Hammerstein-like process (Hahn and Edgar, 2001).

A salient feature of this reactor is that the sign of its steady state gain may change according to the operating condition (see Figure 3.5). In our simulation study, the operating space under consideration is $F \in [8 \;\; 55]$. To apply the proposed identification procedure, one thousand input-output data as shown in Figure 3.6 are collected. The generalized Hammerstein model to be identified consists of a third-order polynomial as static nonlinear part and a second-order linear model. With parameters $k_{\min} = 12$, $k_{\max} = 90$ and Ω= 0.9 chosen for JITL algorithm, the identified nonlinear parameters are γ_1 =1.0237, γ_2 = -0.0004 and γ_3 = 0.0051. Again, for comparison purpose, a conventional

Hammerstein model with $\alpha_1 = 1.7922$, $\alpha_2 = -0.8033$, $\beta_1 = 0.0061$, $\gamma_1 = 0.6034$, $\gamma_2 = -0.3855$ and $\gamma_3 = 0.5109$ are identified by using the Narendra-Gallman method.

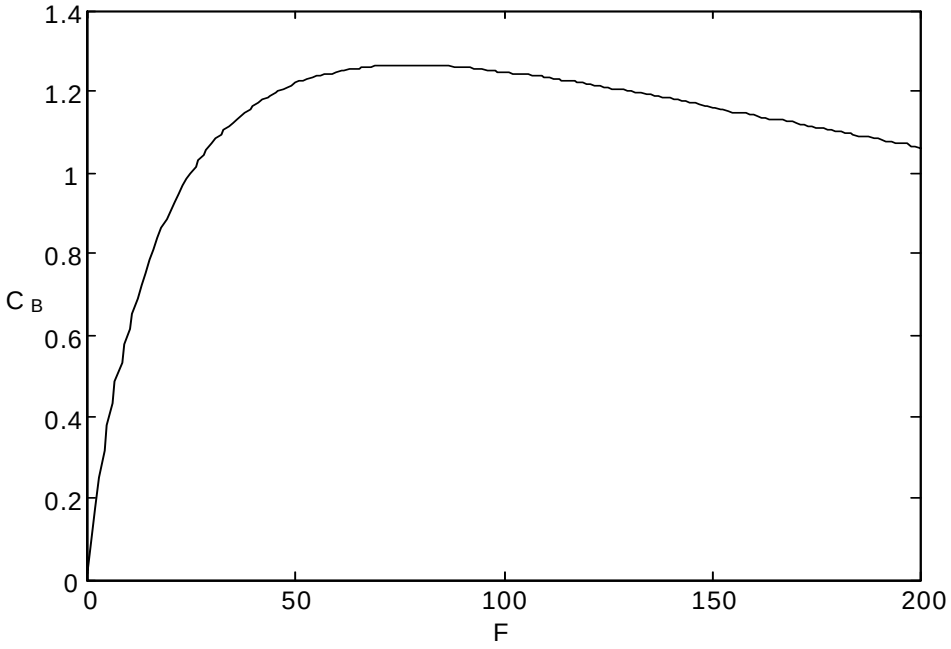


Figure 3.5 Steady-state curve of van de Vusse reactor

Figure 3.7 compares the predictive capability of the generalized Hammerstein and Hammerstein models when *F* is subject to step change of 15 L/hr, while the prediction performance of these two models for open-loop response subject to step change of -25 L/hr in *F* is compared in Figure 3.8. The resulting MAEs of the generalized Hammerstein model are 5.7% and 17.9% of those obtained by Hammerstein model, respectively. Evidently, the former has better prediction accuracy over its conventional counterpart in modeling a process with a wide range of operating space.

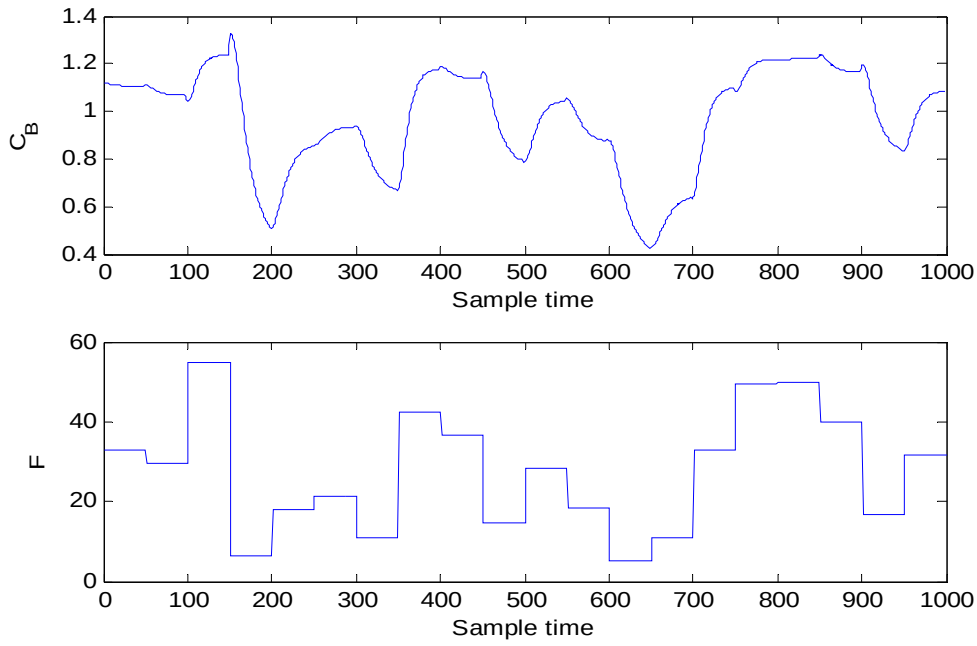


Figure 3.6 Input-output data for van de Vusse reactor

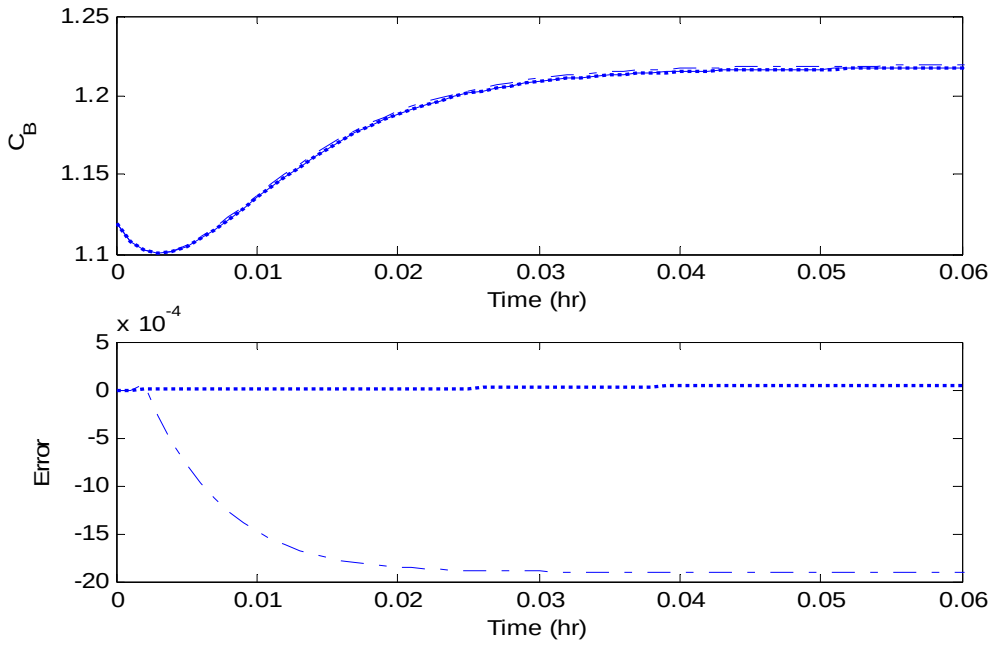


Figure 3.7 Open-loop response for 15 L/hr change in F . Solid line: process; dotted line: generalized Hammerstein model; dash-dot line: Hammerstein model

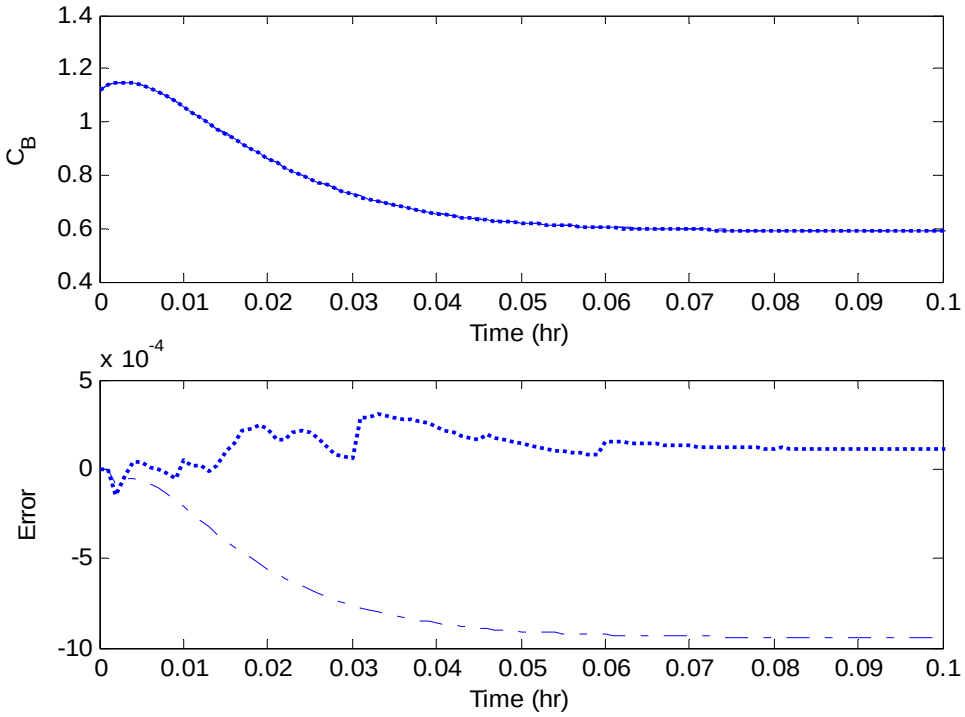


Figure 3.8 Open-loop response for -25 L/hr change in F . Solid line: process; dotted line: generalized Hammerstein model; dash-dot line: Hammerstein model

Example 3 The control of pH is common in the chemical process and biotechnological industries. This process can exhibit severe static nonlinear behavior because the process gain can vary by several orders of magnitude over a modest range of pH values. Moreover, the titration curve may be time varying due to unmeasured change in the buffering capacity.

A simplified schematic diagram of a bench-scale pH neutralization system studied by Henson and Seborg (1994) is shown in Figure 3.9. The process uses NaOH with concentration 0.003 M as the base stream (q_1), $NaHNO_3$ with concentration 0.03 M as the buffer stream (q_2) and HNO_3 with concentration 0.003 M as the acid stream (q_3). The acid stream enters tank 2 which introduces additional flow dynamics. The acid and

base flow rates are regulated with flow control valves, while the buffer flow rate is controlled manually with a rotameter.

The chemical equilibria is modeled by defining two reaction invariants for each inlet stream:

$$W_{ai} = [\mathrm{H}^+]_i - [\mathrm{OH}^-]_i - [\mathrm{HCO}_3^-]_i - 2[\mathrm{CO}_3^{2-}]_i, \quad i = 1 \sim 4 \tag{3.44}$$

$$W_{bi} = [\mathrm{H}_2\mathrm{CO}_3]_i + [\mathrm{HCO}_3^-]_i + [\mathrm{CO}_3^{2-}]_i, \quad i = 1 \sim 4 \tag{3.45}$$

where the invariant W_{ai} is a charge related quantity, while W_{bi} represents the concentration of the CO_3^{2-} ion. Unlike pH, these invariants are conserved quantities. The pH can be determined from W_{a_4} and W_{b_4} using the following relations:

$$W_{b_4} \frac{\frac{K_{a1}}{[\mathrm{H}^+]} + \frac{2K_{a1}K_{a2}}{[\mathrm{H}^+]^2}}{1 + \frac{K_{a1}}{[\mathrm{H}^+]} + \frac{K_{a1}K_{a2}}{[\mathrm{H}^+]^2}} + W_{a_4} + \frac{K_w}{[\mathrm{H}^+]} - [\mathrm{H}^+] = 0 \tag{3.46}$$

$$\mathrm{pH} = -\log([\mathrm{H}^+]) \tag{3.47}$$

The dynamic model of the neutralization process is developed as follows. A mass balance on tank 2 yields,

$$A_2 \frac{dh_2}{dt} = q_3 - q_{3e} \tag{3.48}$$

where h_2 and A_2 are the level and cross-sectional area of tank 2, respectively. The exit flow rate q_{3e} is modeled with the following flow-head relation:

$$q_{3e} = C_{v1} h_2^{0.5} \tag{3.49}$$

where C_{v1} is a constant valve coefficient. An overall mass balance on tank 1 yields:

$$A_1 \frac{dh_1}{dt} = q_1 + q_2 + q_{3e} - q_4 \tag{3.50}$$

where h_1 and A_1 are the level and cross-sectional area of tank 1. The exit flow rate q_4 is model as:

$$q_4 = C_{v4}(h_1 + z)^n \tag{3.51}$$

where C_{v4} is a constant valve coefficient, n is constant valve exponent, and z is the vertical distance between the bottom of tank 1 and the outlet for q_4. By combining mass balances on each of the ionic species in the system, the following differential equations for the effluent reaction invariants, W_{a4} and W_{b4}, can be derived (Henson and Seborg 1994):

$$A_1 h_1 \frac{dW_{a4}}{dt} = q_1(W_{a1} - W_{a4}) + q_2(W_{a2} - W_{a4}) + q_{3e}(W_{a3} - W_{a4}) \tag{3.52}$$

$$A_2 h_2 \frac{dW_{b4}}{dt} = q_1(W_{b1} - W_{b4}) + q_2(W_{b2} - W_{b4}) + q_{3e}(W_{b3} - W_{b4}) \tag{3.53}$$

Table 3.3 Model parameters and nominal operating condition for the pH system

$K_{a1} = 4.47 \times 10^{-7}$	$q_1 = 15.6$ ml/s	$W_{a1} = -3.05 \times 10^{-3}$ M
$K_{a2} = 5.62 \times 10^{-10}$	$q_2 = 0.55$ ml/s	$W_{b1} = 5.00 \times 10^{-5}$ M
$K_w = 1.00 \times 10^{-14}$	$q_3 = 16.6$ ml/s	$W_{a2} = -0.03$ M
$A_1 = 207$ cm^2	$q_4 = 32.75$ ml/s	$W_{b2} = 0.03$ M
$A_2 = 42$ cm^2	$h_1 = 14.0$ cm	$W_{a3} = 3.00 \times 10^{-3}$ M
$z = 11.5$ cm	$h_2 = 3.0$ cm	$W_{b3} = 0$ M
$C_{v1} = 9.58$	pH = 7.0	$W_{a4} = -4.32 \times 10^{-4}$ M
$C_{v4} = 4.58$	$n = 0.607$	$W_{b4} = 5.28 \times 10^{-4}$ M

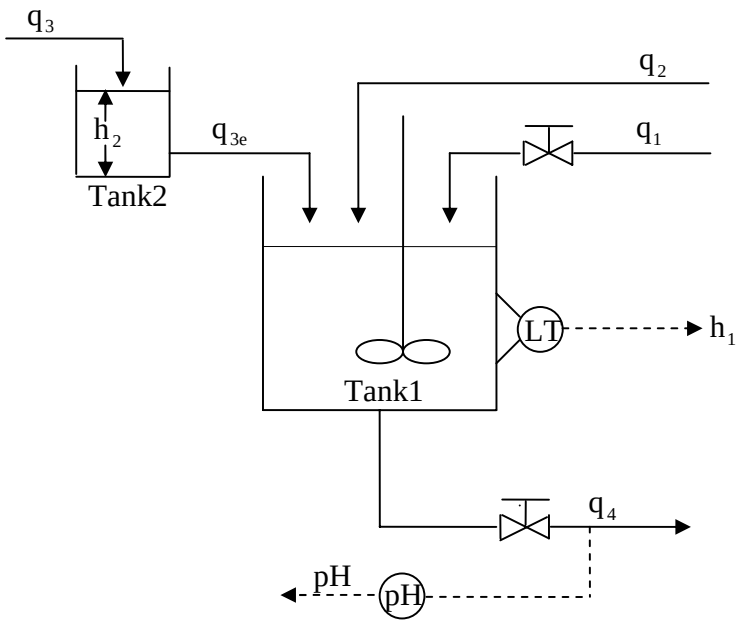


Figure 3.9 The pH neutralization process

This process was previously modeled as a 2×2 Hammerstein model by Lakshminarayanan et al. (1995). Two outputs of this process are h_1 and pH, which will be denoted by y_1 and y_2 in the following development, while the process inputs are q_1 and q_3, which are denoted by u_1 and u_2, respectively. The operating space considered for process modeling is $y_1 \in [10.5\ \ 16.5]$ and $y_2 \in [4.5\ \ 9.5]$.

To proceed with the proposed identification procedure, one thousand independent random signals are collected from input and corresponding process output as shown in Figure 3.10. By using $k_{\min} = 25$, $k_{\max} = 90$ and Ω=0.98 for JITL algorithm, the static nonlinear part of generalized Hammerstein model is identified as:

$$v_1(k) = 2.8745u_1(k) - 0.2689u_1^2(k) + 0.0567u_1^3(k)$$

$$v_2(k) = 1.5495u_2(k) + 6.2594u_2^2(k) + 13.08u_2^3(k)$$

For comparison purpose, the following Hammerstein model is obtained by using Narendra-Gallman method and the identical input and output data given in Figure 3.10:

$$\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} 0.9608 & -0.0395 \\ 0.0607 & 0.9048 \end{bmatrix} \begin{bmatrix} y_1(k-1) \\ y_2(k-1) \end{bmatrix} + \begin{bmatrix} 0.0699 & 0 \\ 0 & -0.1748 \end{bmatrix} \begin{bmatrix} v_1(k-1) \\ v_2(k-1) \end{bmatrix}$$

$$v_1(k) = 0.9782\ u_1(k) + 0.2735u_1^2(k) + 2.3471u_1^3(k)$$

$$v_2(k) = 0.9459u_2(k) - 2.0381u_2^2(k) + 10.869u_2^3(k)$$

The predictive performance of these two models is compared in Figure 3.11 for step changes of 1.5 ml/s and -2.5 ml/s in base flow rate (u_1), respectively, and the corresponding MAEs for prediction error are given in Table 3.4. It is clear that generalized Hammerstein model has superior predictive performance than the conventional Hammerstein model. Likewise, generalized Hammerstein model gives a marked improvement in predicting the open-loop response corresponding to ± 3 ml/s step changes in acid flow rate (u_2), as illustrated in Figure 3.12 and Table 3.4.

Table 3.4 Prediction error for open-loop responses in Figures 3.11 and 3.12

	Hammerstein model		Generalized Hammerstein model	
	y_1	y_2	y_1	y_2
+1.5 ml/s change in u_1	1.38×10^{-1}	1.50×10^{-1}	1.12×10^{-3}	5.76×10^{-3}
-2.5 ml/s change in u_1	3.89×10^{-2}	1.99×10^{-2}	1.74×10^{-3}	1.91×10^{-3}
+3 ml/s change in u_2	9.19×10^{-2}	1.57×10^{-2}	3.51×10^{-3}	4.37×10^{-4}
-3 ml/s change in u_2	6.52×10^{-2}	1.28×10^{-1}	3.21×10^{-3}	9.75×10^{-3}

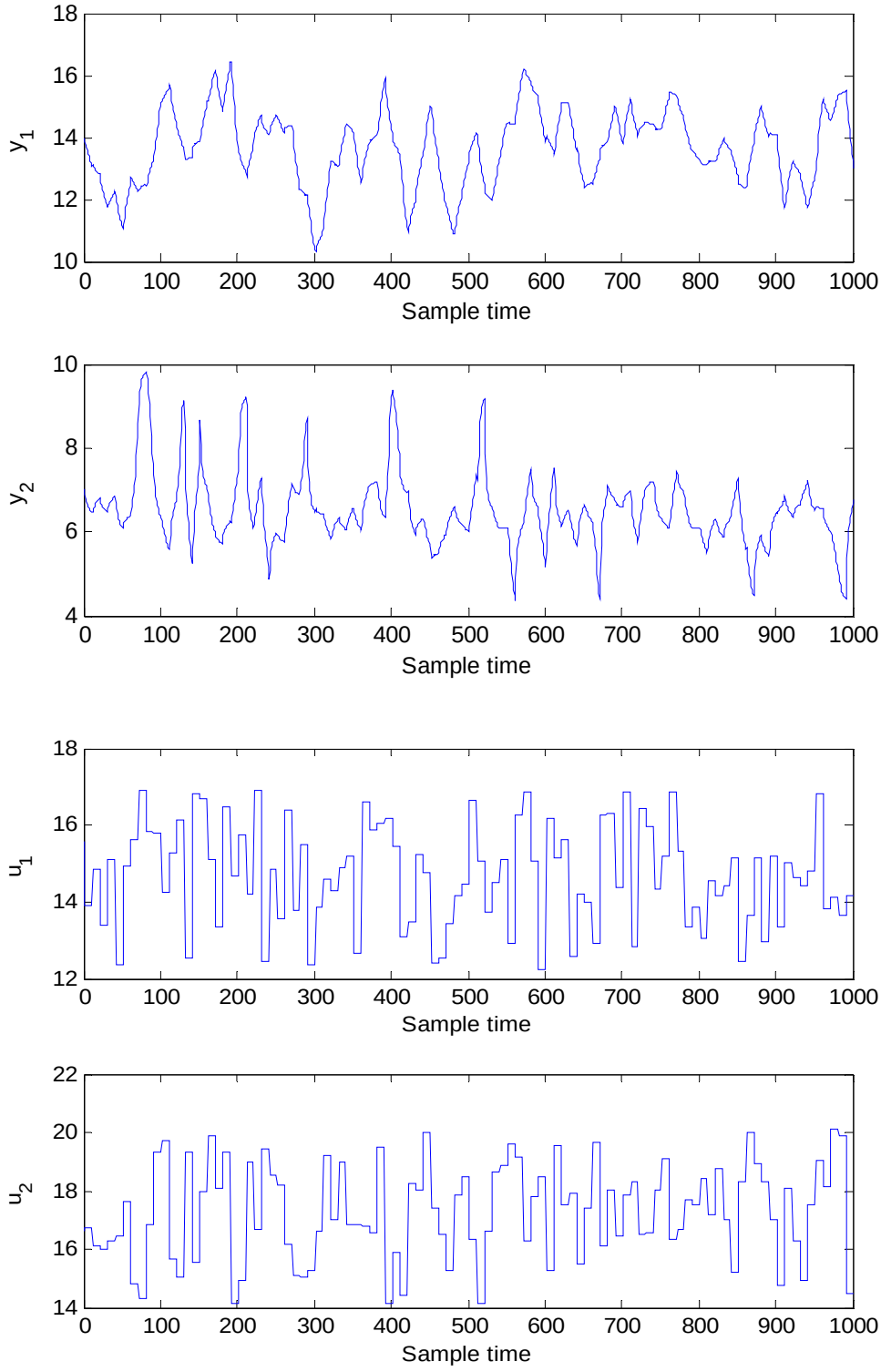
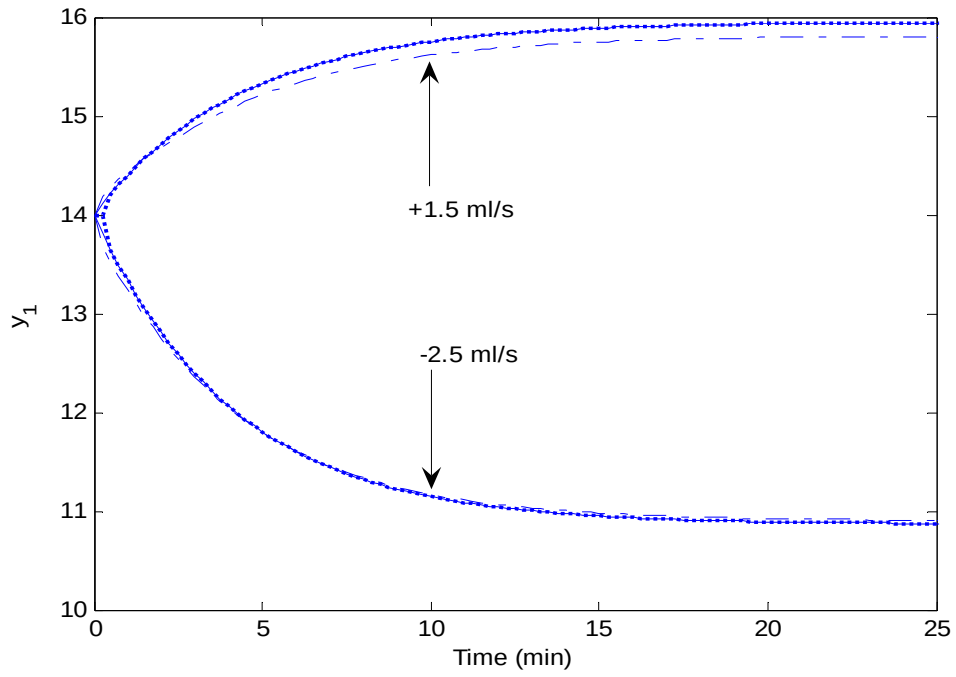
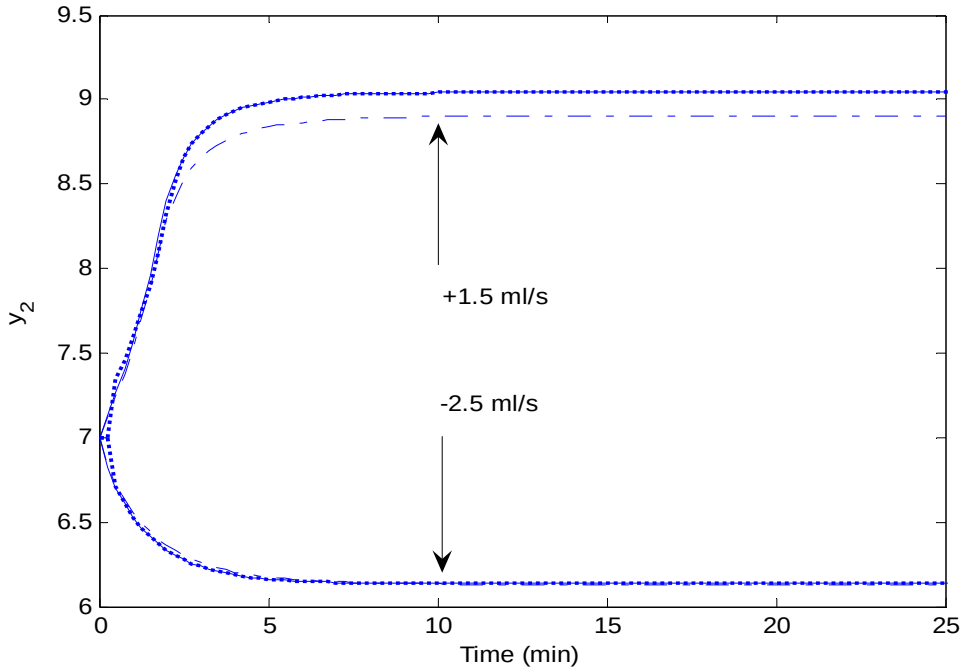


Figure 3.10 Input-output data for pH neutralization process

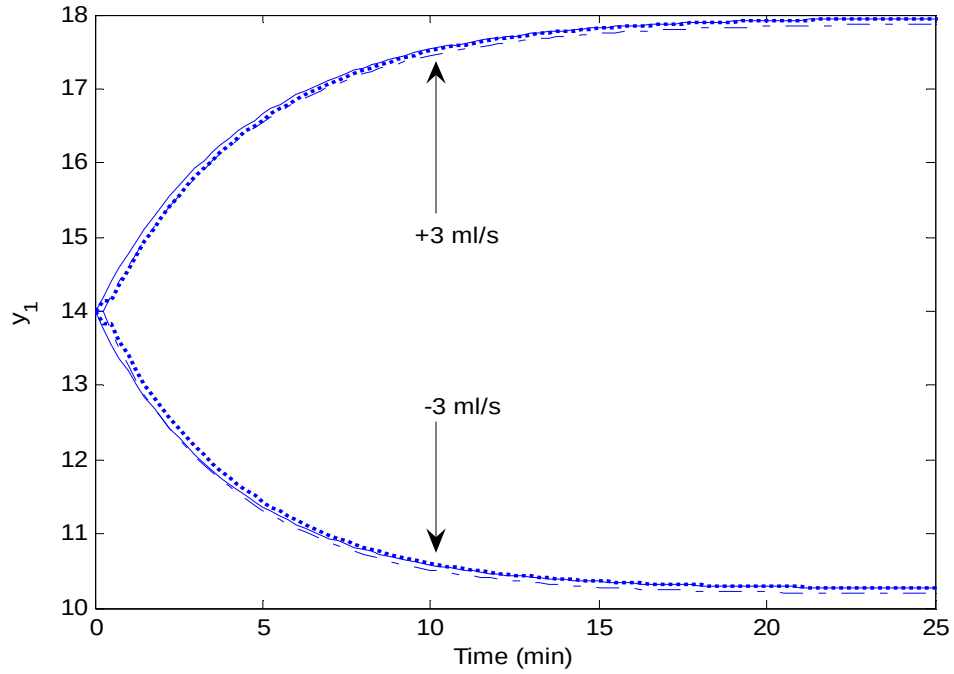


(a)

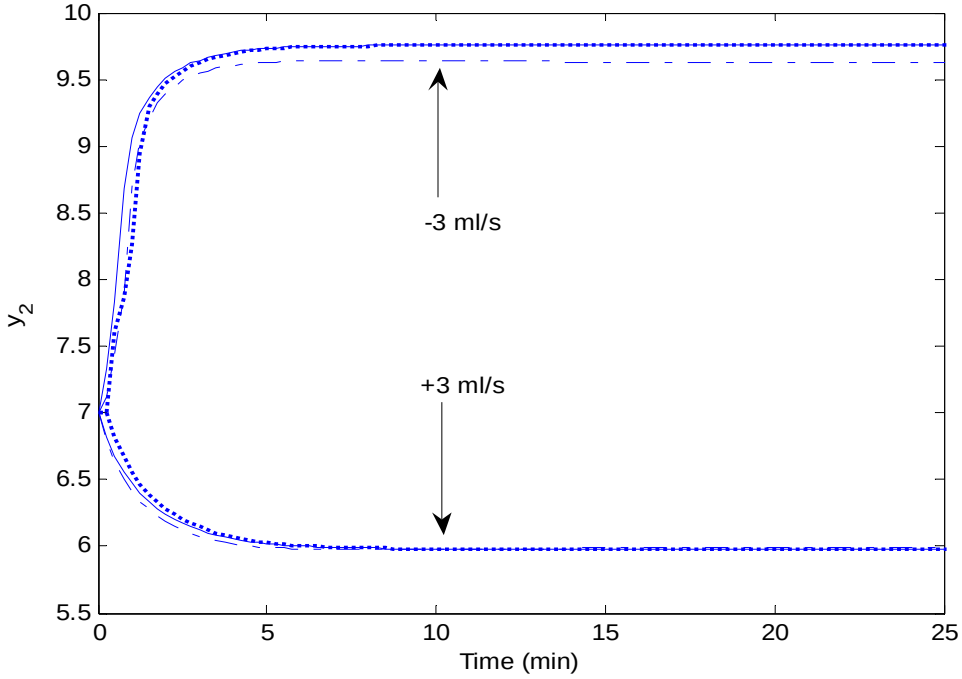


(b)

Figure 3.11 Open-loop response for 1.5 ml/s and -2.5 ml/s changes in q_1 (a) level, (b) pH. Solid line: process; dotted line: generalized Hammerstein model; dash-dot line: Hammerstein model



(a)



(b)

Figure 3.12 Open-loop response for ± 3 ml/s changes in q_3: (a) level, (b) pH. Solid line: process; dotted line: generalized Hammerstein model; dash-dot line: Hammerstein model

Example 4 Considering the following chemical reactions that produce cyclopentanol (B) from cyclopentadiene (A) and the side products are cyclopentanediol (C) and dicyclopentadiene (D) (Stack and Doyle, 1997; Harris and Palazoglu, 1998):

$$\mathrm{A} \xrightarrow{k_1} \mathrm{B} \xrightarrow{k_2} \mathrm{C} \tag{3.54}$$

$$2\mathrm{A} \xrightarrow{k_3} \mathrm{D} \tag{3.55}$$

The above reaction takes place in a jacket-cooled CSTR, where the coolant is introduced by an external heat exchanger. This system can be described by the following equations:

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{Af} - C_A) - k_{1,0}e^{E_1/T}C_A - k_{3,0}e^{E_3/T}C_A^2 \tag{3.56}$$

$$\frac{dC_B}{dt} = -\frac{F}{V}C_B + k_{1,0}e^{E_1/T}C_A - k_{2,0}e^{E_2/T}C_B \tag{3.57}$$

$$\begin{aligned}\frac{dT}{dt} = &-\frac{1}{\rho C_p}[k_{1,0}e^{E_1/T}C_A\Delta H_1 + k_{2,0}e^{E_2/T}C_B\Delta H_2 + k_{3,0}e^{E_3/T}C_A^2\Delta H_3] \\ &+ \frac{F}{V}(T_0 - T) + \frac{k_w A_w}{\rho C_p V}(T_w - T)\end{aligned} \tag{3.58}$$

$$\frac{dT_w}{dt} = \frac{1}{m_w C_{p,w}}\left[Q_w + k_w A_w (T - T_w)\right] \tag{3.59}$$

where C_i is the concentration of species $(i = A, B)$, F is the reactor flow rate, T is the reactor temperature, T_w is the coolant temperature and Q_w is the external heat exchanger duty. The model parameters and nominal operating condition are given in Tables 3.5 and 3.6 respectively. For this 2×2 system, the process outputs are C_B and T (denoted by y_1 and y_2 respectively) and process inputs are F and Q_w (denoted by u_1 and u_2 respectively). The sample time of the system is chosen as 0.001 hr. The operating space considered for process modeling is $y_1 \in [1.12 \;\; 0.5]$ and $y_2 \in [397 \;\; 412]$. This process

Table 3.5 Model parameters for cyclopentanol reactor

V	Reactor volume	10 L
T_0	Inlet temperature	403.15 K
C_{Af}	Feed concentration of component A	5.1 mol/L
C_p	Average heat capacity	30.1 kJ/kg/L
ρ	Average density	0.9342 kg/L
k_w	Coolant conductivity	4032 kJ/h/m^2/K
$C_{p,w}$	Coolant heat capacity	2.0 kJ/kg/K
m_w	Coolant mass	5.0 kg
A_w	Heat exchange area	$0.215\,\mathrm{m}^2$
$k_{1,0}$	Arrhenius constant	1.287×10^{12} l/hr
$k_{2,0}$	Arrhenius constant	1.287×10^{12} l/hr
$k_{3,0}$	Arrhenius constant	9.043×10^{9} L/mol/hr
E_1	Normalized activation energy	-9758.3 K
E_2	Normalized activation energy	-9758.3 K
E_3	Normalized activation energy	-8560 K
ΔH_1	Heat of reaction	4.3 kJ/mol
ΔH_2	Heat of reaction	-11 kJ/mol
ΔH_3	Heat of reaction	-41.85 kJ/mol

Table 3.6 Nominal operating condition for cyclopentanol reactor

C_A	1.235 mol/L	T_w	402.1 K
C_B	0.900 mol/L	F	188.3 L/hr
T	407.3 K	Q_w	-4496 kJ/hr

was previously modeled as Hammerstein-like process (Hahn and Edgar, 2001).

To proceed with the proposed identification procedure, one thousand independent random signals are collected from input and corresponding process output as shown in Figure 3.13. By using $k_{\min} = 12$, $k_{\max} = 90$ and Ω=0.9 for JITL algorithm, the static nonlinear part of generalized Hammerstein model is identified as:

$$v_1(k) = 0.6944u_1(k) + 0.0654u_1^2(k) - 0.0668u_1^3(k)$$

$$v_2(k) = 0.9998u_2(k) + 3\times10^{-4}u_2^2(k) + 1.6\times10^{-4}u_2^3(k)$$

For comparison purpose, the following Hammerstein model is obtained by using Narendra-Gallman method and the identical input and output data given in Figure 3.13:

$$\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} 0.9717 & -0.4714 \\ 0.0010 & 0.9889 \end{bmatrix} \begin{bmatrix} y_1(k-1) \\ y_2(k-1) \end{bmatrix} + \begin{bmatrix} 0.0203 & 0 \\ 0 & -0.0004 \end{bmatrix} \begin{bmatrix} v_1(k-1) \\ v_2(k-1) \end{bmatrix}$$

$$v_1(k) = 0.7714u_1(k) - 0.5957u_1^2(k) + 0.3491u_1^3(k)$$

$$v_2(k) = 1.0044u_2(k) + 1.4\times10^{-3}u_2^2(k) + 4.1\times10^{-4}u_2^3(k)$$

Figures 3.14 and 3.15 compare the predictive performance of these two models for 100 L/hr and -180 L/hr step changes in reactor flow rate (u_1), respectively. Likewise, their respective predictive performance for open-loop response corresponding to 1.9 MJ/hr and -1.5 MJ/hr step changes in external heat exchanger duty (u_2) is illustrated in Figures 3.16 and 3.17. The MAEs of prediction errors for the aforementioned open-loop responses are summarized in Table 3.7. Evidently, generalized Hammerstein model has better accuracy than its conventional counterpart.

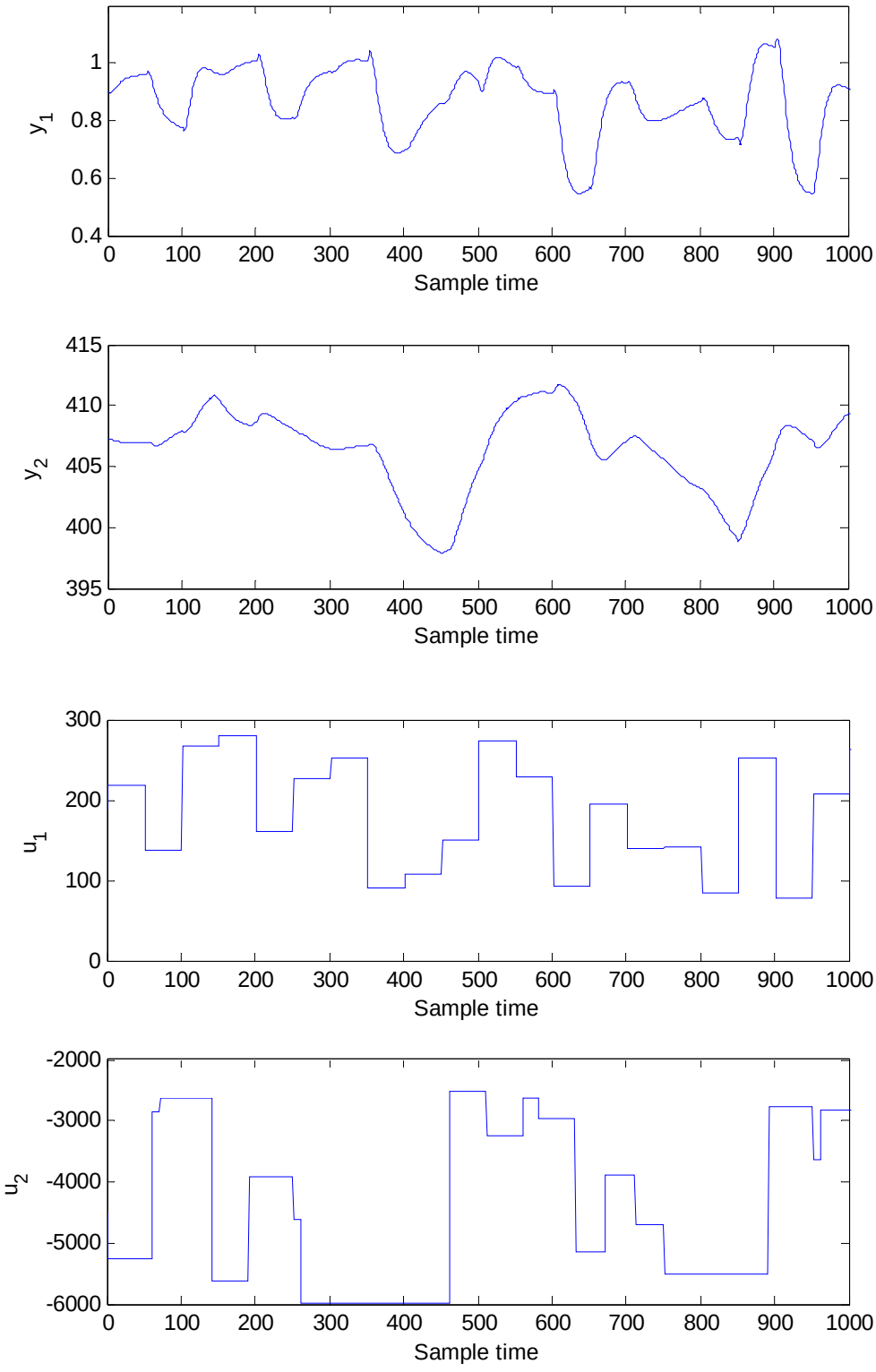


Figure 3.13 Input-output data for cyclopentanol reactor

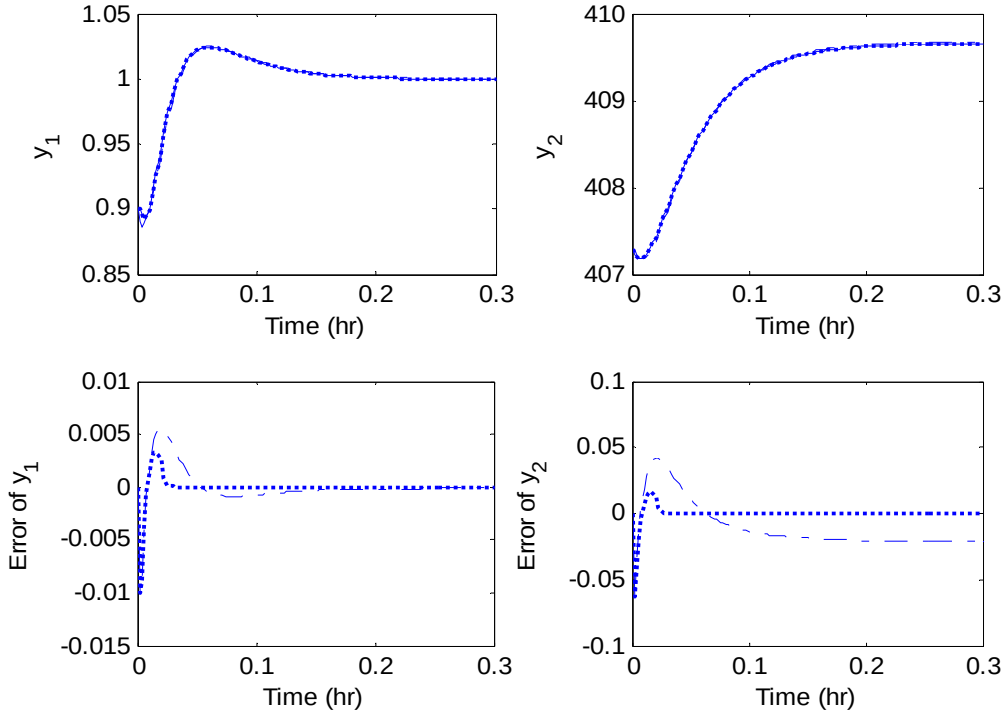


Figure 3.14 Open-loop response for 100 L/hr change in *F*

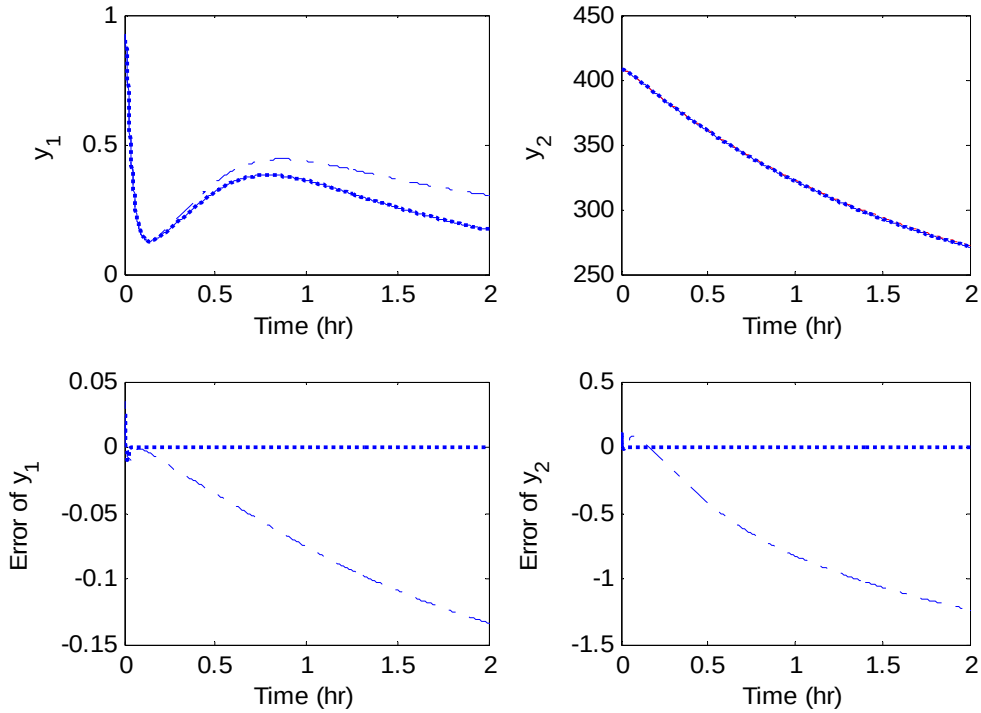


Figure 3.15 Open-loop response for -180 L/hr change in *F*

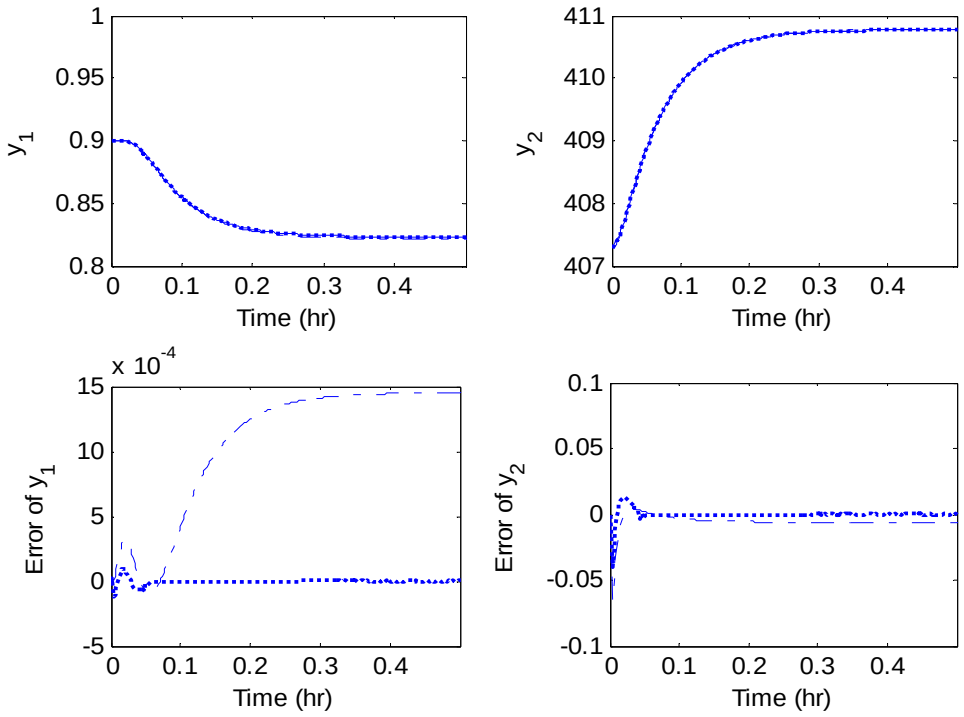


Figure 3.16 Open-loop response for 1.9 MJ/hr change in Q_w

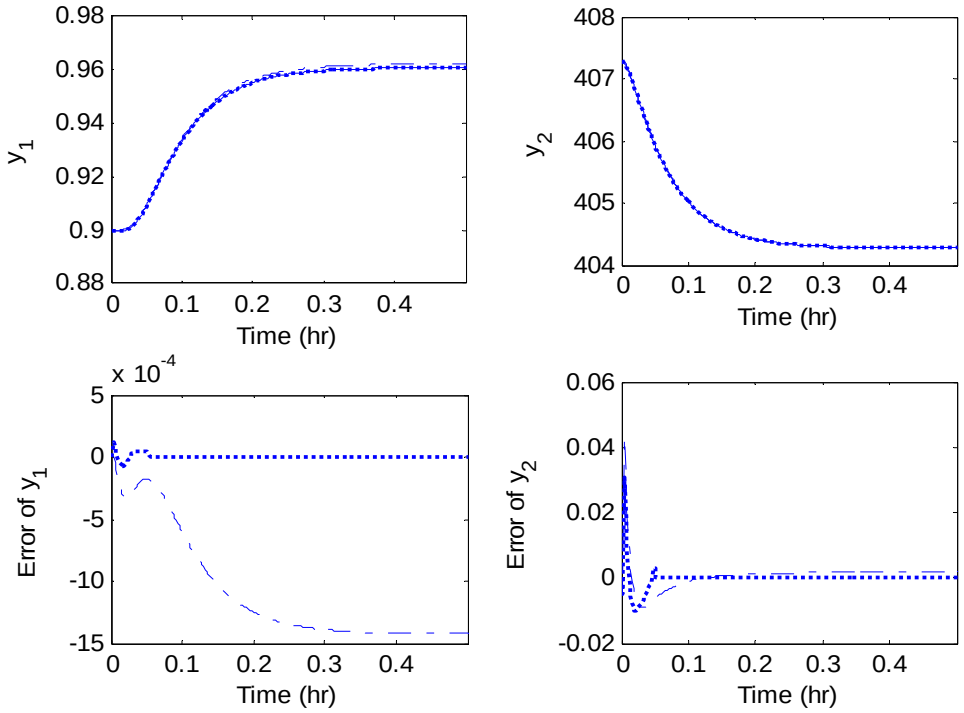


Figure 3.17 Open-loop response for -1.5 MJ/hr change in Q_w

Table 3.7 Prediction error for open-loop responses in Figures 3.14 to 3.17

	Hammerstein model		Generalized Hammerstein model	
	y_1	y_2	y_1	y_2
+100 L/hr change in u_1	8.81×10^{-4}	1.89×10^{-2}	2.77×10^{-4}	1.46×10^{-3}
-180 L/hr change in u_1	7.14×10^{-2}	7.26×10^{-1}	1.50×10^{-4}	3.79×10^{-4}
+1.9 MJ/hr change in u_2	1.06×10^{-3}	5.41×10^{-3}	4.73×10^{-6}	8.86×10^{-4}
-1.5 MJ/hr change in u_2	1.10×10^{-3}	2.31×10^{-3}	7.32×10^{-6}	7.35×10^{-4}

3.5 Conclusions

Generalized Hammerstein model is proposed for modeling the Hammerstein-like processes whose linear dynamics vary over the operating space. Iterative identification procedures for generalized SISO and MIMO Hammerstein models are developed. Unlike the identification of conventional Hammerstein model, only the polynomial function obtained by the proposed identification method will be retained as the static nonlinear part of the generalized Hammerstein model. As a result, on-line application of generalized Hammerstein model requires the computation of linear model by using the JITL technique. Simulations results show that generalized Hammerstein model gives better predictive performance than its conventional counterpart.

CHAPTER 4

Control of Generalized Hammerstein Processes - SISO Cases

The control of Hammerstein processes has been previously studied (Hwang and Hsu, 1995; Ling and Rivera, 2001; Fruzzetti et al., 1997; Lakshminarayana et al., 1997; Sung, 2002) by employing the nonlinear control scheme as depicted in Figure 4.1 (a). Owing to the nonlinear block NL^{-1}, the reciprocal of the static nonlinear part of the Hammerstein model, which is used to cancel the static nonlinearity (NL) of the process, the advantage of this control strategy is that the design of feedback controller is simplified as a linear controller design problem as shown in Figure 4.1 (b). In the same spirit of aforementioned nonlinear control strategy, a nonlinear IMC control system as shown in Figure 4.2 was investigated for Hammerstein processes (Fruzzetti et al., 1997; Ling and Rivera, 2001). In contrast to one static nonlinear block employed in Figure 4.1, two static nonlinear blocks, NL and NL^{-1}, are required for nonlinear IMC design so that linear IMC design procedure can be applied directly to design IMC controller $Q(z)$ based on the linear process model $\tilde{G}(z)$. In this chapter, the aforementioned control strategies for Hammerstein processes will be extended to develop adaptive control strategies for generalized Hammerstein processes.

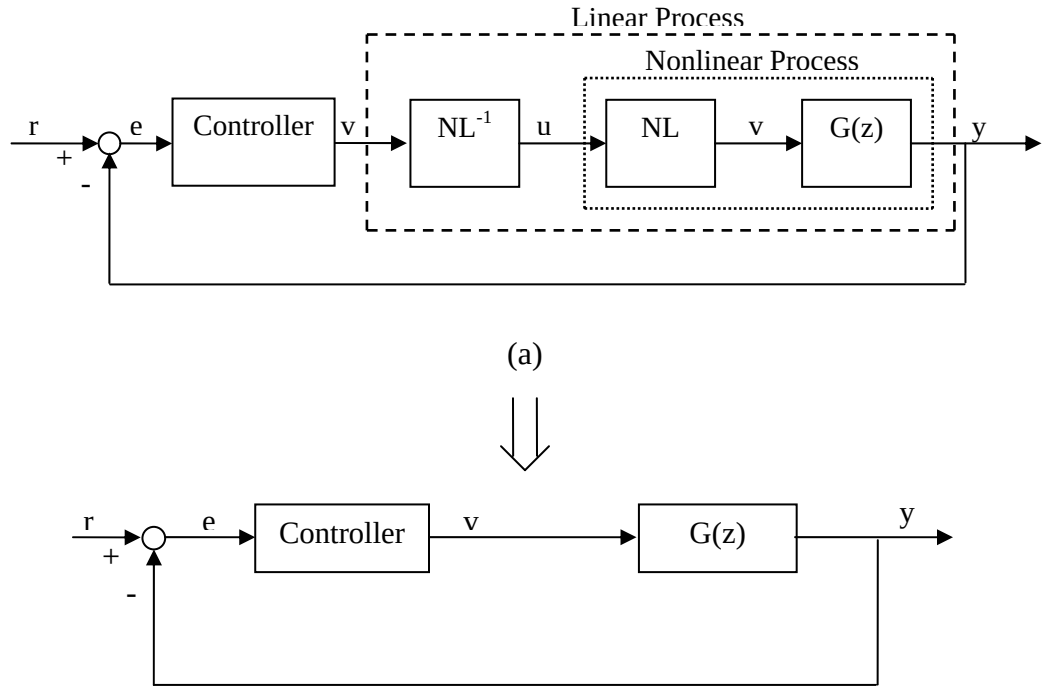


Figure 4.1 (a) Nonlinear controller design for Hammerstein processes, and (b) equivalent linear control system

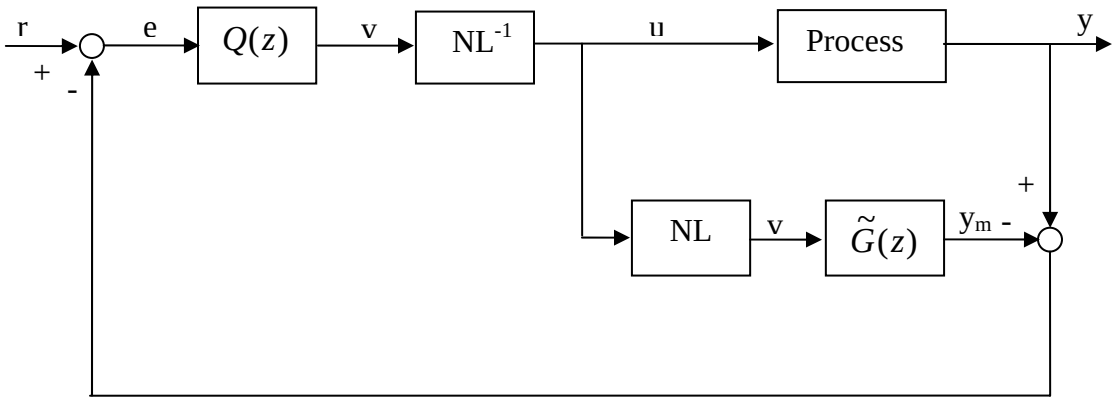


Figure 4.2 Internal model control for Hammerstein processes

4.1 Adaptive IMC Controller Design

In this section, an adaptive IMC control strategy for generalized Hammerstein processes will be developed. Recall that the linear model of generalized Hammerstein processes has the following form:

$$y(k) = \alpha_1^k y(k-1) + \alpha_2^k y(k-2) + \beta_1^k v(k-1-n_d) + \beta_2^k v(k-2-n_d) \tag{4.1}$$

In other words, linear IMC model is given by:

$$\tilde{G}^k(z) = \frac{(\beta_1^k + \beta_2^k z^{-1}) z^{-n_d-1}}{1-\alpha_1^k z^{-1} - \alpha_2^k z^{-2}} \tag{4.2}$$

where the model parameters $\alpha_1^k, \alpha_2^k, \beta_1^k$ and β_2^k are identified by JITL algorithm at each sampling instant. As such, an extension of IMC strategy for Hammerstein processes to generalized Hammerstein processes can be implemented in connection with JITL technique as depicted in Figure 4.3. As can be seen, JITL is employed not only to update the model parameters but also to adjust the parameters of IMC controller $Q^k(z)$ as well. This is because $Q^k(z)$ is designed based on the inversion of process model $\tilde{G}^k(z)$ as dictated by the IMC design procedure discussed in Chapter 2. As a result, those controller parameters pertaining to the model parameters of $\tilde{G}^k(z)$ need to be updated by JITL algorithm at each sampling instant. For illustration purpose, consider the following first-order process model:

$$\tilde{G}^k(z) = \frac{\beta_1^k z^{-1}}{1-\alpha_1^k z^{-1}} \tag{4.3}$$

Using a first-order IMC filter, $Q^k(z)$ is designed by

$$Q^k(z) = \frac{v(z)}{e(z)} = \frac{1-\alpha_1^k z^{-1}}{\beta_1^k} \frac{1-\lambda^k}{1-\lambda^k z^{-1}} \tag{4.4}$$

The control law resulting from Eq. (4.4) is then given by

$$v(k) = \lambda^k v(k-1) + \frac{1-\lambda^k}{\beta_1^k}(e(k) - \alpha_1^k e(k-1)) \tag{4.5}$$

where α_1^k and β_1^k are obtained from JITL algorithm, $e(k)$ is the error between process output and its set-point at the k-th sampling instant, and λ^k is the IMC filter time constant adjusted on-line by the gradient descent algorithm to be discussed below.

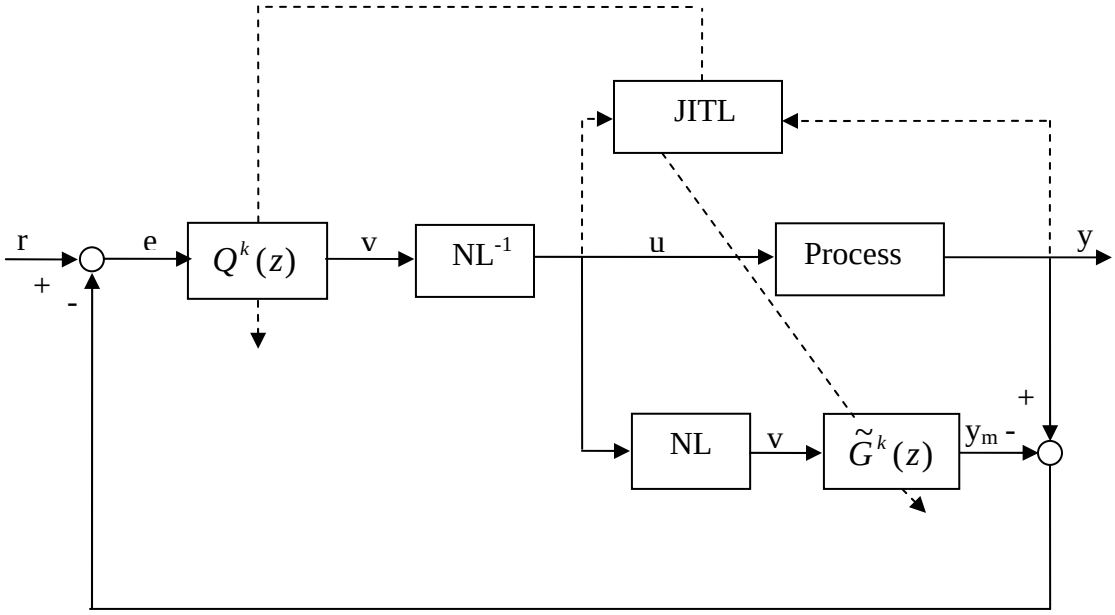


Figure 4.3 Adaptive IMC control system for generalized Hammerstein processes

The following objective function is used to update λ^k :

$$\text{Min } J = (r(k+1) - \hat{y}(k+1))^2 \tag{4.6}$$

where $r(k+1)$ is the set-point and $\hat{y}(k+1)$ is the predicted output of JITL algorithm.

Because λ^k is constrained between 0 and 1, the following mapping function is introduced,

$$\lambda^k = \frac{1}{1+e^{-\varphi(k)}} \tag{4.7}$$

where $\varphi(k)$ is a real number. To adjust the IMC parameter on-line, $\varphi(k+1)$ will be calculated by the updating equation (4.8) and the corresponding λ^{k+1} can then be obtained by Eq. (4.7).

Inspired by the neural network learning strategy, backpropogation method is applied to tune the controller parameter at every sampling time as following:

$$\begin{aligned}\varphi(k+1) &= \varphi(k) - \tilde{\eta}\frac{\partial J}{\partial \varphi(k)} \\ &= \varphi(k) + 2\eta(r(k+1) - \hat{y}(k+1))\frac{z(k)}{1+z^2(k)}\end{aligned} \tag{4.8}$$

where η is a user-specified learning rate and

$$\eta = \tilde{\eta}(1+z^2(k)) \tag{4.9}$$

$$\begin{aligned}z(k) &= \frac{\partial \hat{y}(k+1)}{\partial \varphi(k)} = \frac{\partial \hat{y}(k+1)}{\partial v(k)}\frac{\partial v(k)}{\partial \lambda^k}\frac{\partial \lambda^k}{\partial \varphi(k)} \\ &= \beta_1^{k+1}\lambda^k(1-\lambda^k)(v(k-1) - \frac{1}{\beta_1^k}(e(k) - \alpha_1^k e(k-1)))\end{aligned} \tag{4.10}$$

The convergence property of the parameter updating equation (4.8) was studied by Chen (2001), who proved that the parameter $\varphi(k)$ converges to its local optimum asymptotically provided $0 < \eta < 2$ holds. This explains why the new learning rate η is introduced in Eq. (4.8) to replace the original learning rate $\tilde{\eta}$.

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

1. Initialize λ^k (and $\varphi(k)$ for that matter) and η;
2. Given the current error $e(k)$, compute $v(k)$ from Eq. (4.5) and calculate manipulated variable $u(k)$;
3. Update linear model by using the most current process data and JITL algorithm and subsequently adjust $\varphi(k)$ according to Eq. (4.8);
4. Obtain IMC filter time constant for the next sampling instant by Eq. (4.7) and go to step 2.

4.2 Examples

Example 1 Consider the polymerization of methyl methacrylate in a jacketed CSTR discussed earlier in Chapter 3, where the control objective is to control the number average molecular weight (M_p) by manipulating the inlet initiator concentration, F_I. To proceed with the proposed controller design, the generalized Hammerstein model and the same reference data set used for JITL algorithm in Chapter 3 are incorporated into adaptive IMC control system as depicted in Figure 4.3. For comparison purpose, IMC control system as shown in Figure 4.2 is also designed based on the Hammerstein model identified in Chapter 3.

To evaluate the servo performance of two controllers, $\pm 50\%$ step changes in the set-point as indicated by the dashed line in Figure 4.4 are considered. The controller parameters employed for adaptive IMC design are $\lambda^0 = 0.55$ and $\eta = 0.1$ and for Hammerstein model based IMC design $\lambda = 0.75$. As can be seen from Figure 4.4, adaptive IMC controller has better control performance than that achieved by IMC design, as also evidenced by comparing their respective MAEs given in Table 4.1. Figures 4.5

and 4.6 compare the disturbance rejection capabilities of two controllers when $\pm 10\%$ step changes in $C_{I,in}$ occur at three operating conditions, i.e. $M_p = 38000, 25000.5$, and 12000. It is apparent that the proposed IMC design has consistent and superior control performance over the operating space than its conventional counterpart. The resulting MAEs of load response are summarized in Table 4.1.

Lastly, to test the robustness of the proposed adaptive IMC controller, both process input and output are corrupted by 1% Gaussian white noise. As shown in Figure 4.7, the proposed IMC design can yield reasonably good control performance in the presence of process noise.

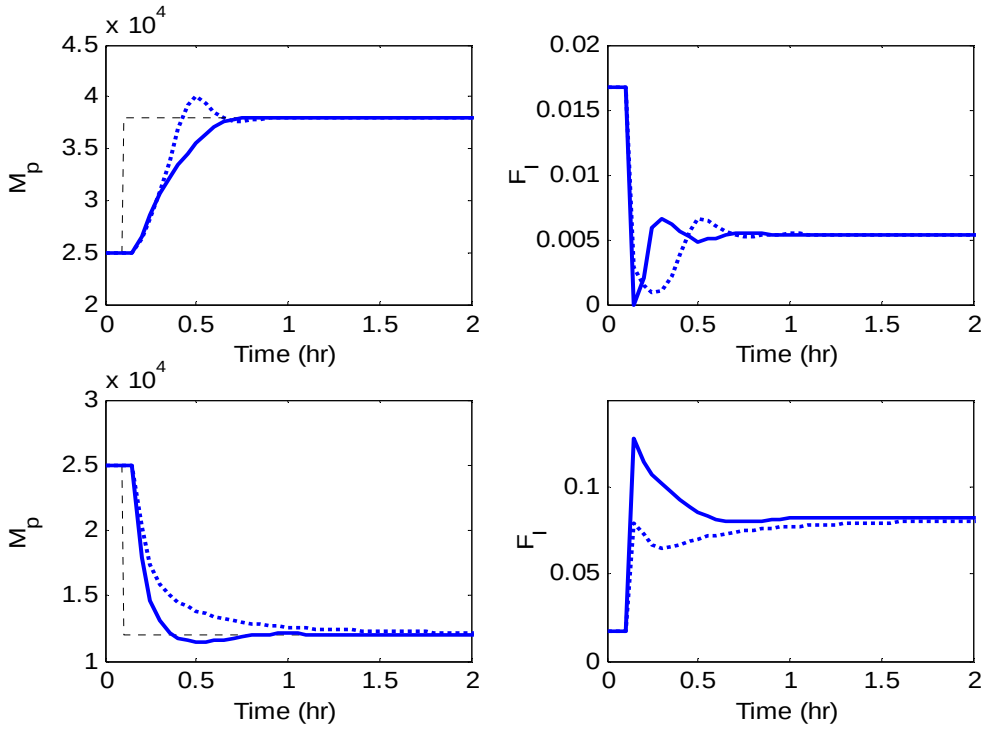


Figure 4.4 Closed-loop response for ± 50 % set-point changes. Solid line: adaptive IMC design; dotted line: Hammerstein model based IMC design

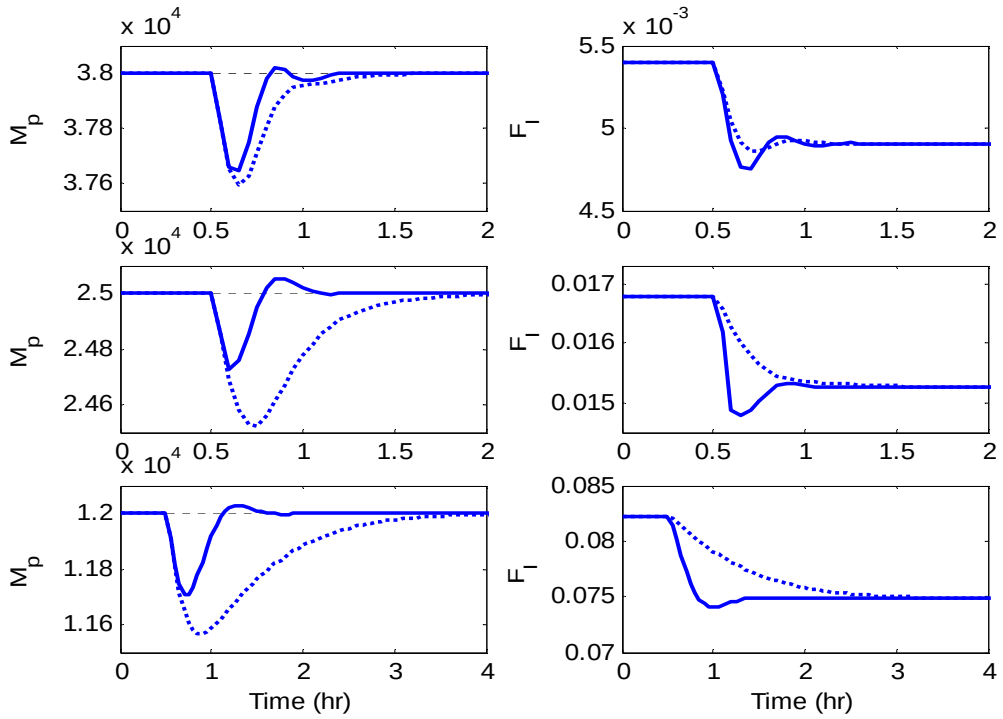


Figure 4.5 Closed-loop response for 10% change in $C_{I,in.}$ Solid line: adaptive IMC design; dotted line: Hammerstein model based IMC design

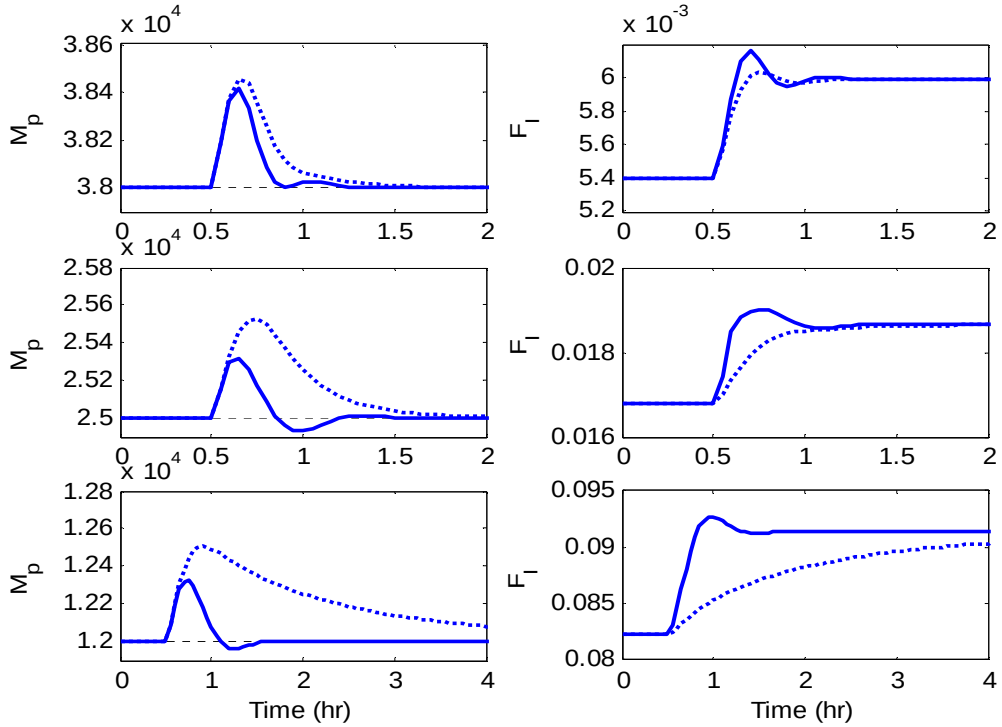


Figure 4.6 Closed-loop response for -10% change in $C_{I,in.}$. Solid line: adaptive IMC design; dotted line: Hammerstein model based IMC design

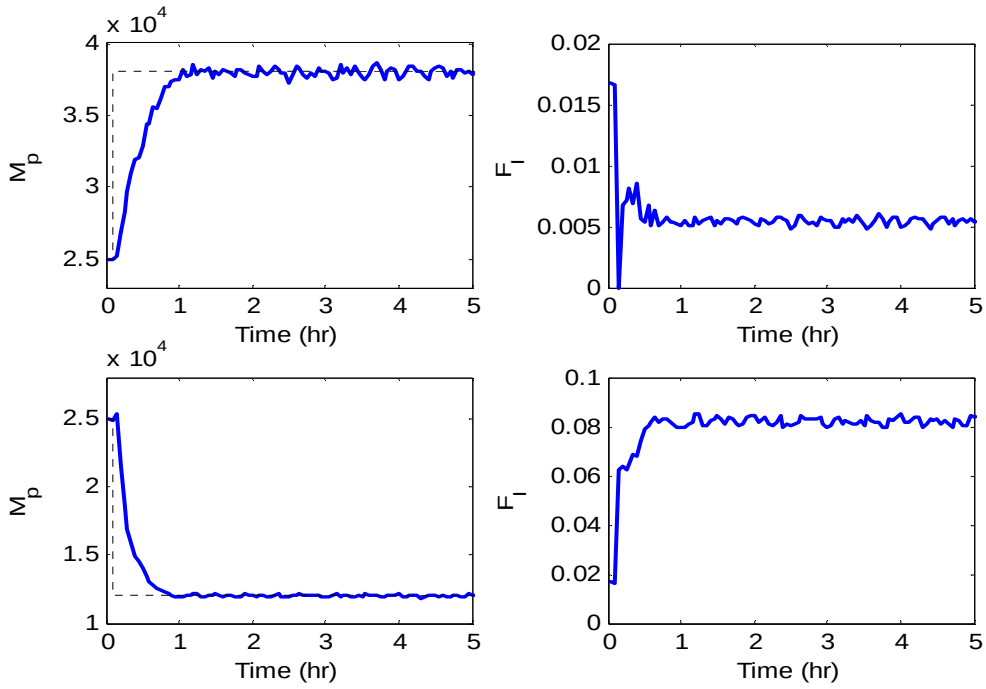


Figure 4.7 Closed-loop response for ± 50 % set-point changes (with process noise)

Table 4.1 Summary of MAEs for closed-loop responses in Figures 4.4 to 4.6

	Hammerstein model based IMC design	Adaptive IMC design
+50% set-point change	1.47×10^{3}	1.39×10^{3}
-50% set-point change	8.24×10^{2}	6.16×10^{2}
+10% in $C_{I,in}$ at $M_p = 38000$	5.71×10^{1}	3.27×10^{1}
+10% in $C_{I,in}$ at $M_p = 25000.5$	1.10×10^{2}	2.49×10^{1}
+10% in $C_{I,in}$ at $M_p = 12000$	1.19×10^{2}	2.67×10^{1}
-10% in $C_{I,in}$ at $M_p = 38000$	6.83×10^{1}	4.01×10^{1}
-10% in $C_{I,in}$ at $M_p = 25000.5$	1.24×10^{2}	3.85×10^{1}
-10% in $C_{I,in}$ at $M_p = 12000$	2.08×10^{2}	2.89×10^{1}

Example 2 Considering the van de Vusse reactor discussed in Chapter 3, where the control problem focuses on regulating the concentration of component B, C_B, by manipulating the inlet flow rate F. Again, the generalized Hammerstein model and the same reference data set used for JITL algorithm in Chapter 3 are incorporated into the proposed adaptive IMC control system. For comparison purpose, IMC control system is also designed based on the Hammerstein model obtained in Chapter 3.

Figure 4.8 compares the servo performance of two controllers for 10% and -50% step changes in the set-point, respectively. The controller parameters employed for adaptive IMC design are $\lambda^0 = 0.92$ and $\eta = 0.5$ and for Hammerstein model based IMC design a value of $\lambda = 0.93$ is used. As can be seen from Figure 4.8 and Table 4.2, adaptive IMC controller outperforms the IMC design. Figures 4.9 and 4.10 compare the disturbance rejection capabilities of two controllers when $\pm 10\%$ step changes in C_{Af} occur at two operating conditions, i.e. $C_B = 1.23$ and 0.56. Evidently, the proposed IMC design has faster and smoother control performance over the operating space than its counterpart based on Hammerstein model. The resulting MAEs of load response are summarized in Table 4.2.

4.3 Adaptive PID Controller Design

The well-known PID controllers are still the most adopted controllers in the process industries. However, its performance may deteriorate when processes exhibit nonlinear behaviour or are operated for a wide range of operating condition. For nonlinear processes that can be described by Hammerstein model, Ling and Rivera (2001)

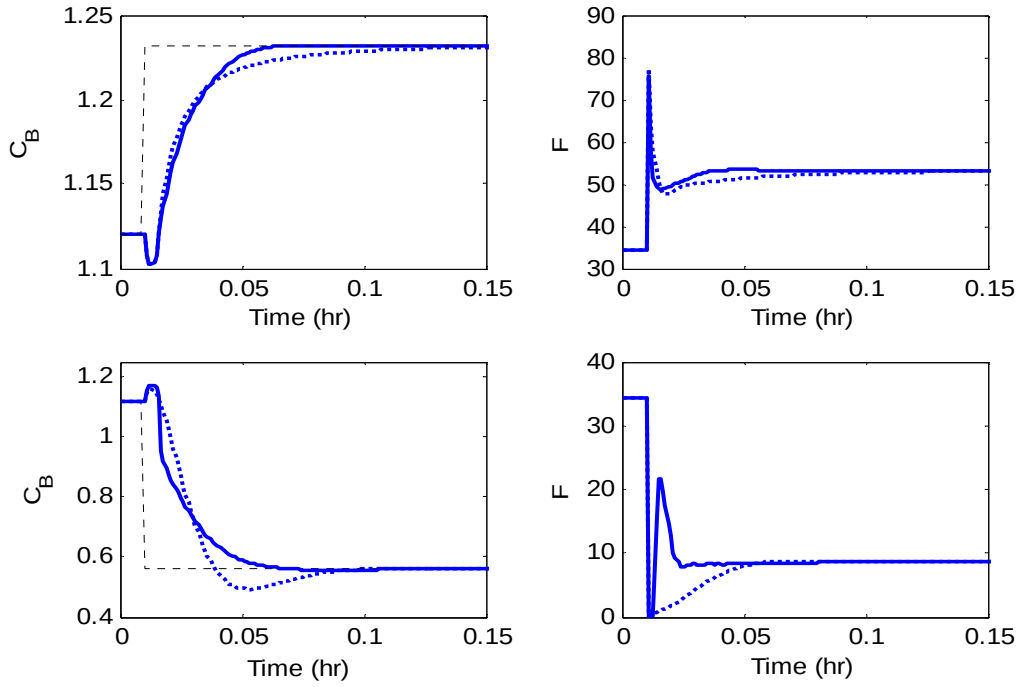


Figure 4.8 Closed-loop response for 10% and -50% set-point changes. Solid line: adaptive IMC design; dotted line: Hammerstein model based IMC design

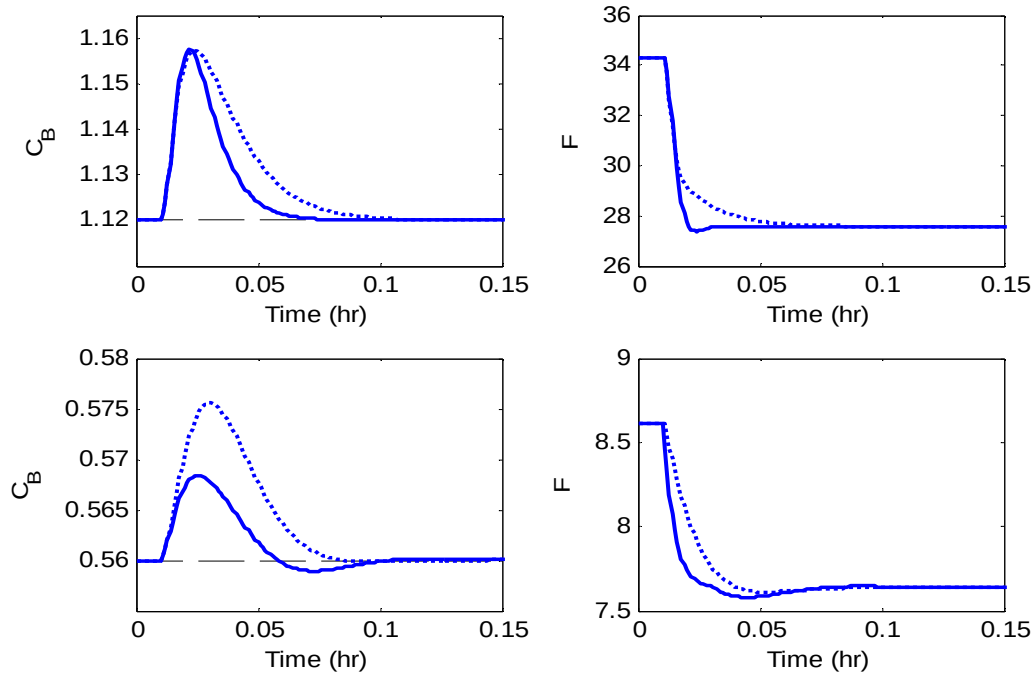


Figure 4.9 Closed-loop response for 10% change in C_{Af} . Solid line: adaptive IMC design; dotted line: Hammerstein model based IMC design

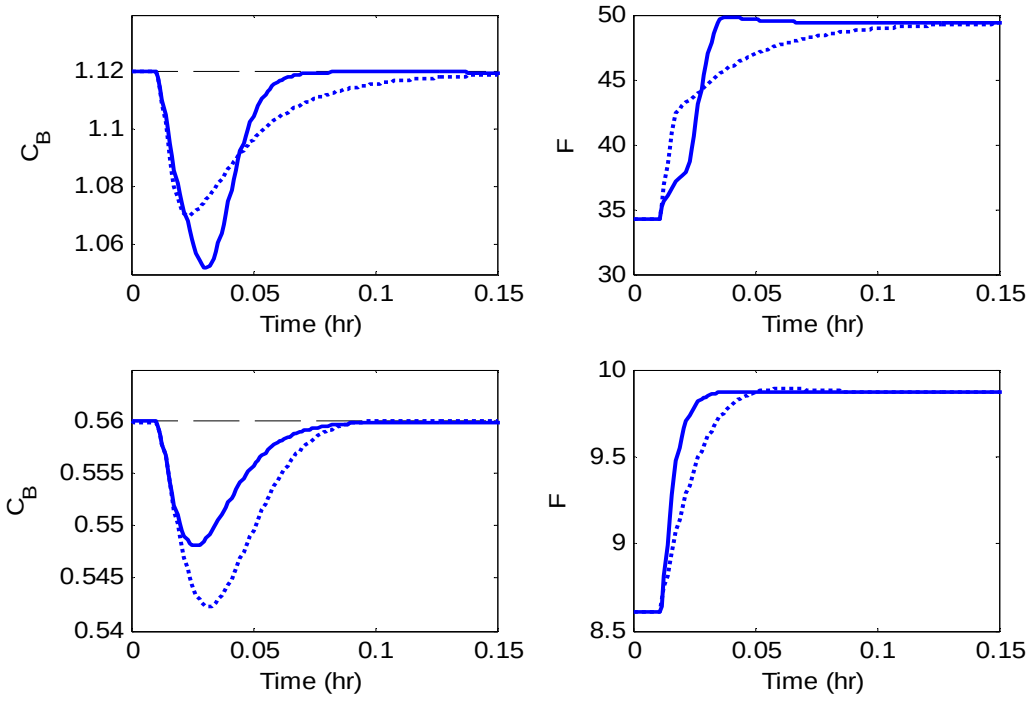


Figure 4.10 Closed-loop response for -10% change in C_{Af} . Solid line: adaptive IMC design; dotted line: Hammerstein model based IMC design

Table 4.2 Summary of MAEs for closed-loop responses in Figures 4.8 to 4.10.

	Hammerstein model based IMC design	Adaptive IMC design
10% set-point change	1.38×10^{-2}	1.35×10^{-2}
-50% set-point change	7.35×10^{-2}	6.11×10^{-2}
+10% in C_{Af} at $C_B = 1.12$	7.43×10^{-3}	4.97×10^{-3}
+10% in C_{Af} at $C_B = 0.56$	3.25×10^{-3}	1.60×10^{-3}
-10% in C_{Af} at $C_B = 1.12$	1.26×10^{-2}	1.09×10^{-2}
-10% in C_{Af} at $C_B = 0.56$	3.40×10^{-3}	2.30×10^{-3}

and Sung (2002) designed nonlinear PID control system as that depicted in Figure 4.2 for controlling a polymerization reactor and a thermal microsystem.

Figure 4.11 illustrates the proposed adaptive PID control system, which is an extension of nonlinear control system in Figure 4.2 to generalized Hammerstein processes. Again, the nonlinear block NL^{-1} in Figure 4.11 is used to remove the static nonlinear part of the process so that the design of PID controller can focus on the linear dynamics part of the process. As a result of time-varying nature of linear dynamics in the generalized Hammerstein processes, JITL is employed as an on-line estimator to provide necessary information to update PID parameters. In this sense, the resulting control system is an adaptive PID control system.

Considering the following PID control algorithm:

$$v(k) = v(k-1) + \Delta v(k) \tag{4.11}$$

$$\Delta v(k) = w_1^k e(k) + w_2^k \Delta e(k) + w_3^k (e(k) - 2e(k-1) + e(k-2)) \tag{4.12}$$

where $\Delta e(k) = e(k) - e(k-1)$, and w_1^k, w_2^k and w_3^k are the tuning parameters to be determined online by the updating formula derived in the sequel.

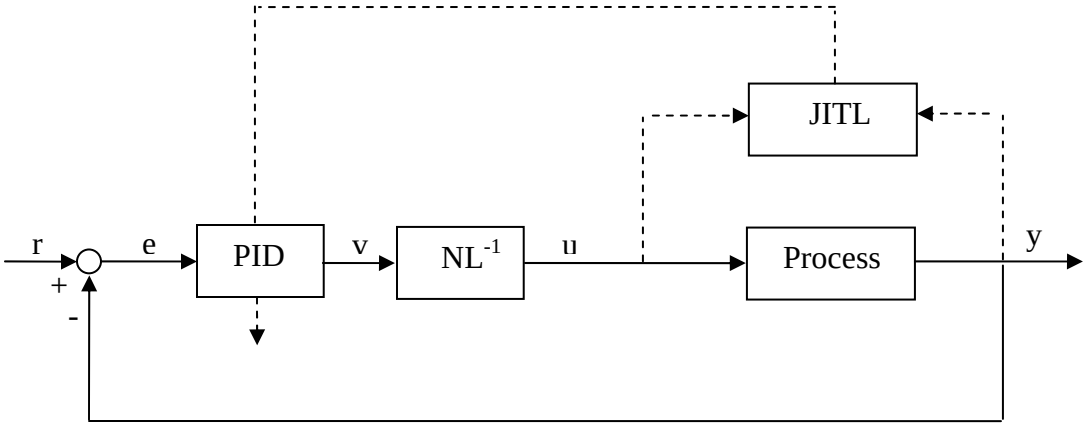


Figure 4.11 Adaptive PID control system for generalized Hammerstein processes

Similar to adaptive IMC design, the following objective function is used to update PID parameters:

$$\text{Min } J = (r(k+1) - \hat{y}(k+1))^2 \tag{4.13}$$

Since the parameter w_i^k is constrained to be positive or negative, the following mapping function is introduced:

$$w_i^k = \begin{cases} e^{\varsigma_i(k)}, & \text{if} \quad w_i^k \geq 0 \\ -e^{\varsigma_i(k)}, & \text{if} \quad w_i^k < 0 \end{cases}, \quad i = 1 \sim 3 \tag{4.14}$$

where ς_i is real number. Henceforth, $\varsigma_i(k)$ will be adjusted by the following updating equation and the respective PID parameters are obtained by Eq. (4.14).

$$\begin{aligned} \varsigma_i(k+1) &= \varsigma_i(k) - \tilde{\eta}_i \frac{\partial J}{\partial \varsigma_i(k)} \\ &= \varsigma_i(k) + 2\eta(r(k+1) - \hat{y}(k+1)) \frac{z_i(k)}{1 + z_i^2(k)}, \quad i = 1 \sim 3 \end{aligned} \tag{4.15}$$

where η is a user-specified learning rate and

$$\tilde{\eta}_i = \frac{\eta}{1 + z_i^2(k)} \tag{4.16}$$

$$z_1(k) = \frac{\partial \hat{y}(k+1)}{\partial v(k)} \frac{\partial v(k)}{\partial w_1^k} \frac{\partial w_1^k}{\partial \varsigma_1(k)} = \beta_1^{k+1} e(k) e^{\varsigma_1(k)} \tag{4.17}$$

$$z_2(k) = \beta_1^{k+1} \Delta e(k) e^{\varsigma_2(k)} \tag{4.18}$$

$$z_3(k) = \beta_1^{k+1} \big(e(k) - 2e(k-1) + e(k-2)\big) e^{\varsigma_3(k)} \tag{4.19}$$

The implementation of the proposed adaptive PID algorithm is summarized as follows:

1. Initialize w_i^k (and $\varsigma_i(k)$ for that matter) and η;

2. Given the current error $e(k)$, compute $v(k)$ from Eq. (4.11) and calculate manipulated variable $u(k)$;
3. Update linear model by using the most current process data and JITL algorithm and subsequently adjust $\varsigma_i(k)$ according to Eq. (4.15);
4. Obtain PID parameters for the next sampling instant using Eq. (4.14) and go back to step 2.

4.4 Examples

Example 1 The first example considered is the control of polymerization reaction studied in section 4.2. The proposed PID design is based on the generalized Hammerstein model and the same reference data set used for JITL algorithm in Chapter 3. In addition, IMC design based on Hammerstein model as given in section 4.2 will serve as a benchmark design for comparison purposes.

With initial controller parameters $w_1^0 = -1.6$, $w_2^0 = -2$, $w_3^0 = -0.01$ and learning rate $\eta = 1.8$ chosen for the proposed adaptive PID controller, Figure 4.12 compares servo performance of two controllers for $\pm 50\%$ step changes in the set-point. As can be seen from Figure 4.12 and corresponding tracking errors given in Table 4.3, adaptive PID controller has better control performance than that achieved by IMC design. Figures 4.13 and 4.14 compare the disturbance rejection capabilities of two controllers when $\pm 10\%$ step changes in $C_{I,in}$ occur at operating conditions $M_p = 38000, 25000.5,$ and 12000. The MAEs of these load responses are also summarized in Table 4.3. It is apparent that the

proposed IMC design has superior control performance over the operating space than its conventional counterpart.

To test the robustness of the proposed adaptive PID controller method, both process input and output are corrupted by 1% Gaussian white noise. As shown in Figure 4.15, the proposed adaptive PID controller can yield reasonably good control performance in the presence of process noise.

Table 4.3 Summary of MAEs for closed-loop responses in Figures 4.12 to 4.14

	Hammerstein model based IMC design	Adaptive PID design
+50% set-point change	1.47×10^{3}	1.39×10^{3}
-50% set-point change	8.24×10^{2}	7.82×10^{2}
+10% in $C_{I,in}$ at $M_p = 38000$	5.71×10^{1}	3.31×10^{1}
+10% in $C_{I,in}$ at $M_p = 25000.5$	1.10×10^{2}	7.38×10^{1}
+10% in $C_{I,in}$ at $M_p = 12000$	1.19×10^{2}	7.18×10^{1}
-10% in $C_{I,in}$ at $M_p = 38000$	6.83×10^{1}	4.02×10^{1}
-10% in $C_{I,in}$ at $M_p = 25000.5$	1.24×10^{2}	8.39×10^{1}
-10% in $C_{I,in}$ at $M_p = 12000$	2.08×10^{2}	5.63×10^{1}

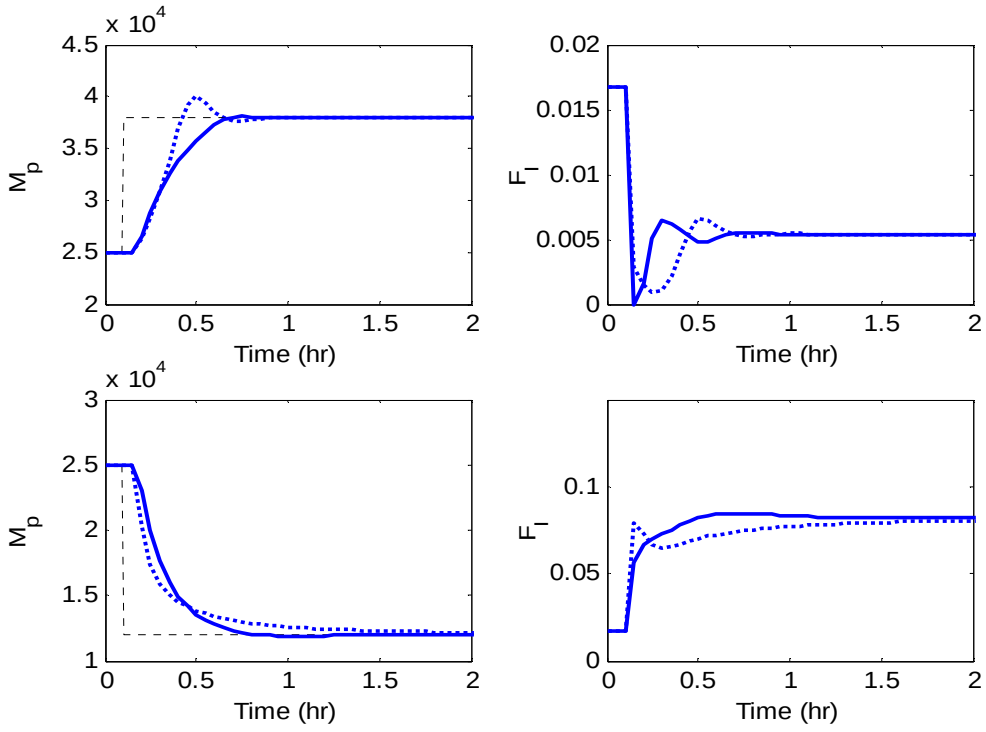


Figure 4.12 Closed-loop response for ± 50 % set-point changes. Solid line: adaptive PID design; dotted line: Hammerstein model based IMC design

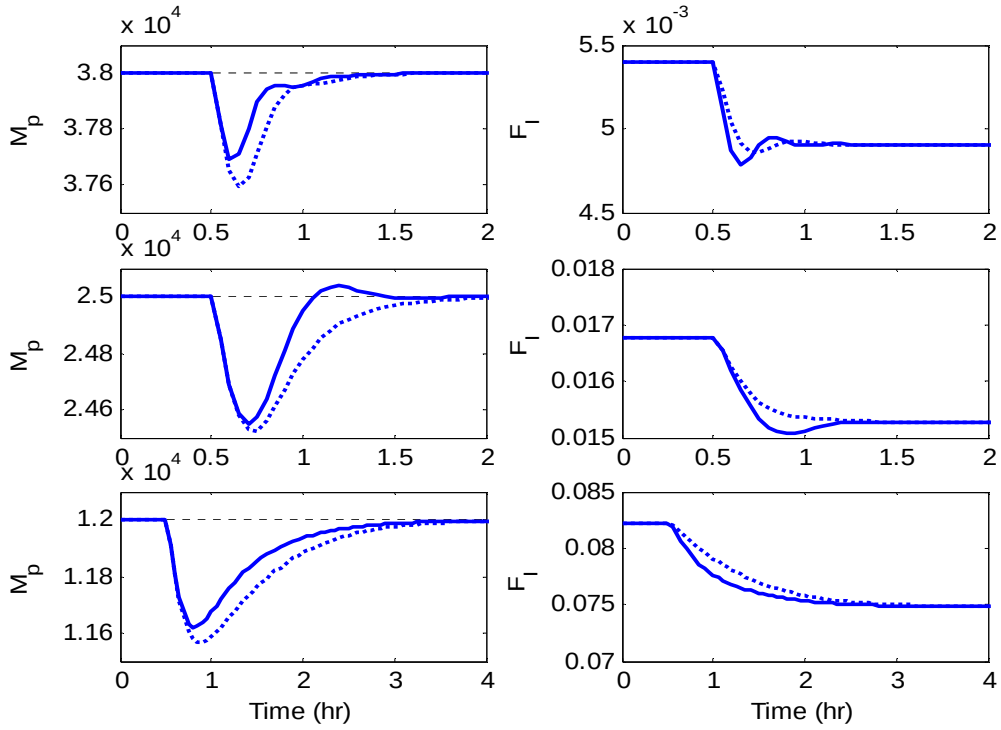


Figure 4.13 Closed-loop response for 10% change in $C_{I,in.}$ Solid line: adaptive PID design; dotted line: Hammerstein model based IMC design

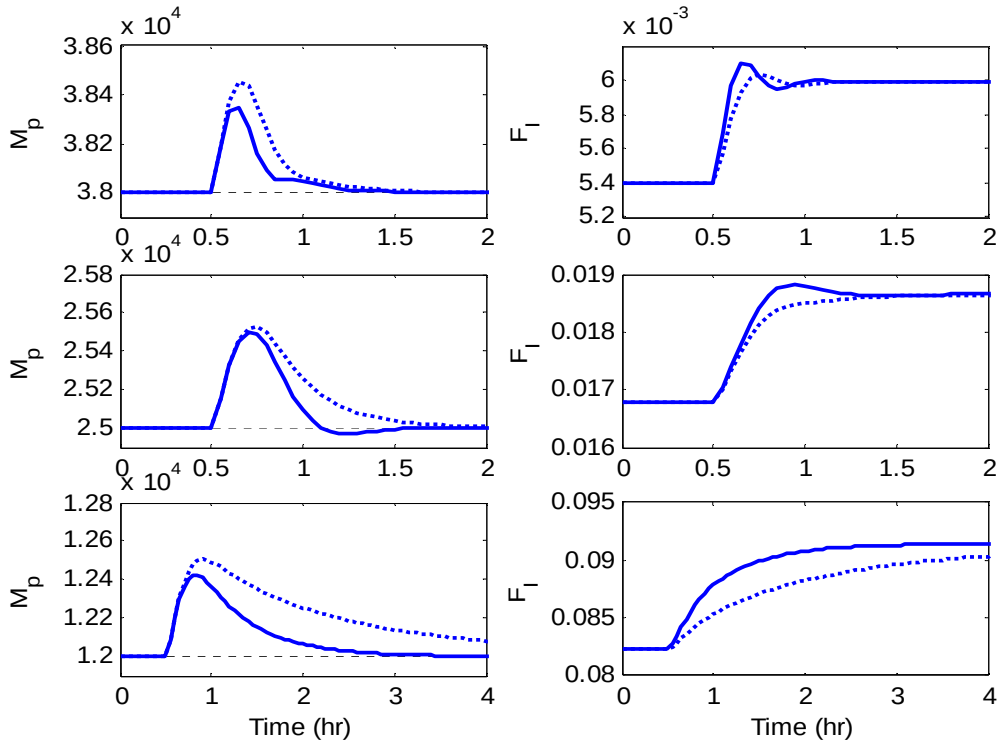


Figure 4.14 Closed-loop response for -10% change in $C_{I,in.}$. Solid line: adaptive PID design; dotted line: Hammerstein model based IMC design .

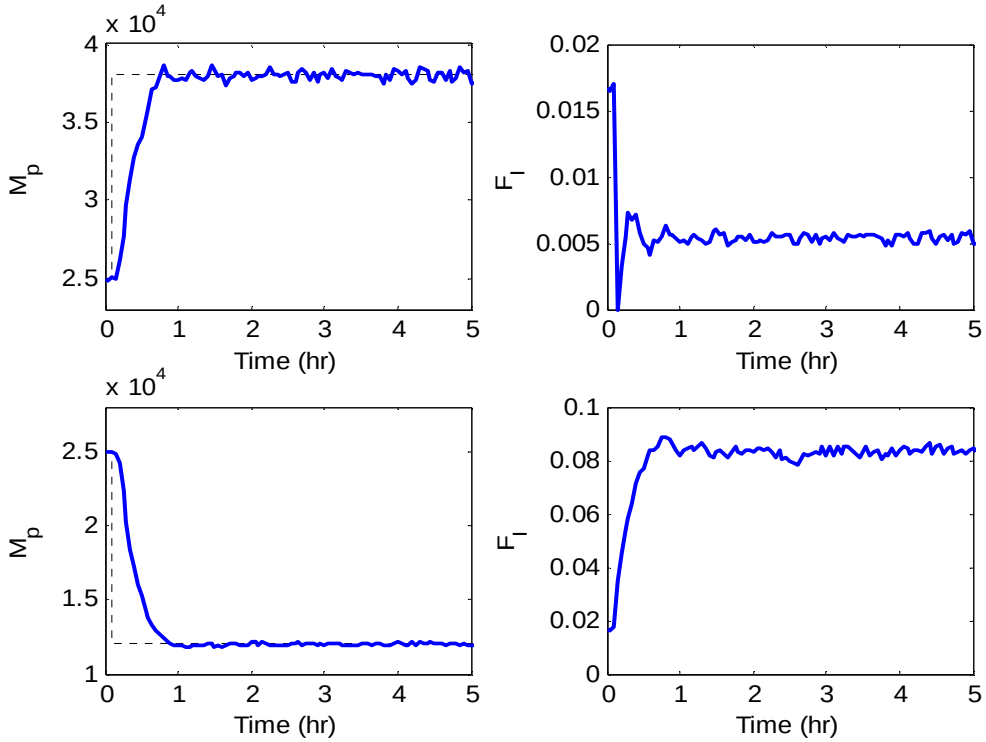


Figure 4.15 Closed-loop response for ± 50 % set-point changes (with process noise)

Example 2 Consider again the control of van de Vusse reactor as studied in section 4.2. With initial controller parameters $w_1^0 = 0.25, w_2^0 = 3.5,\ w_3^0 = 15$ and learning rate $\eta = 1.5$ chosen for adaptive PID controller, Figure 4.16 compares the resulting servo response and that by Hammerstein model based IMC controller for 10% and -50% set-point changes. The corresponding tracking errors are given in Table 4.4. The disturbance rejection capabilities of these two controllers are illustrated in Figures 4.17 and 4.18 and their respective MAEs are summarized in Table 4.4. It is evident that adaptive PID controller outperforms IMC controller designed based on Hammerstein model.

Table 4.4 Summary of MAEs for closed-loop responses in Figures 4.16 to 4.18

	Hammerstein model based IMC controller	Adaptive PID controller
+10% set-point change	1.38×10^{-2}	1.36×10^{-2}
-50% set-point change	7.35×10^{-2}	3.00×10^{-2}
10% increase in C_{Af} at $C_B = 1.12$	7.43×10^{-3}	5.60×10^{-3}
10% increase in C_{Af} at $C_B = 0.56$	3.25×10^{-3}	8.26×10^{-4}
10% decrease in C_{Af} at $C_B = 1.12$	1.26×10^{-2}	9.14×10^{-3}
10% decrease in C_{Af} at $C_B = 0.56$	3.40×10^{-3}	1.08×10^{-3}

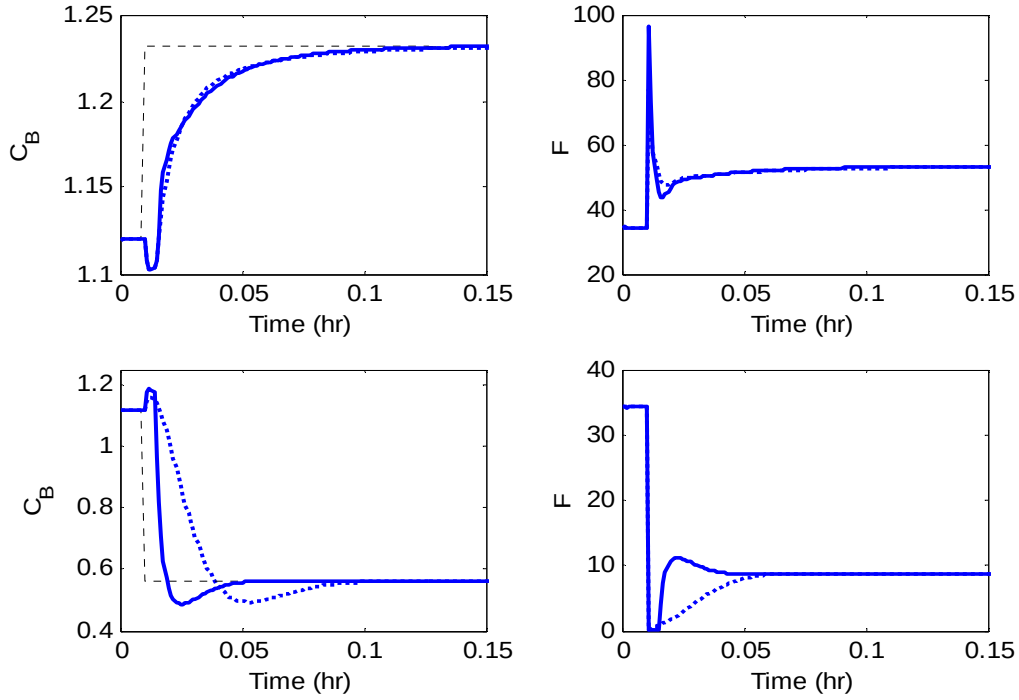


Figure 4.16 Closed-loop response for 10% and -50% set-point changes. Solid line: adaptive PID design; dotted line: Hammerstein model based IMC design

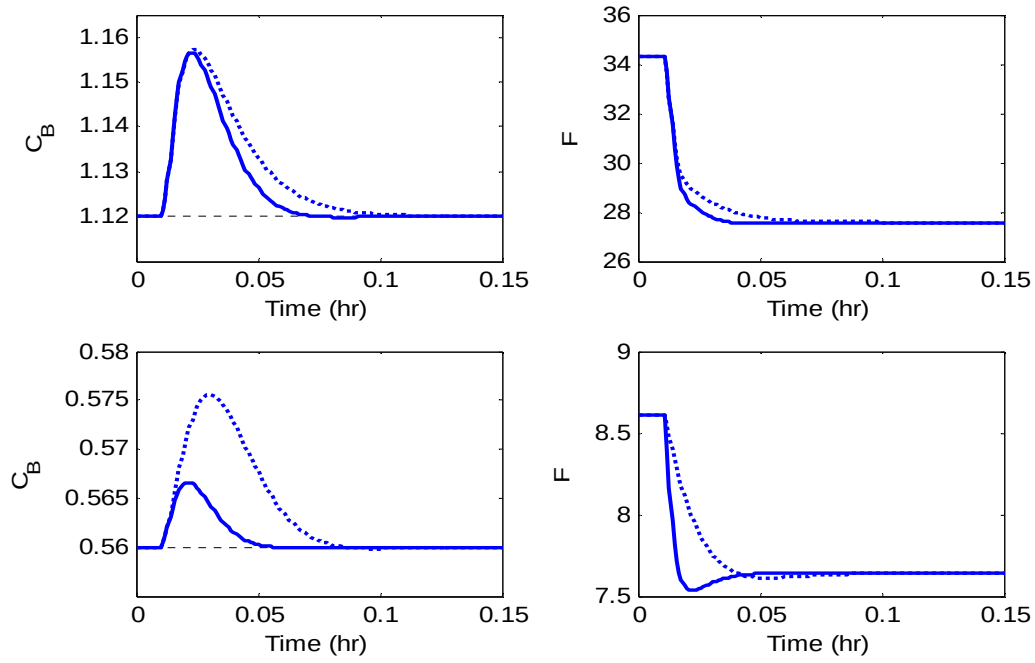


Figure 4.17 Closed-loop response for 10% change in C_{Af} . Solid line: adaptive PID design; dotted line: Hammerstein model based IMC design

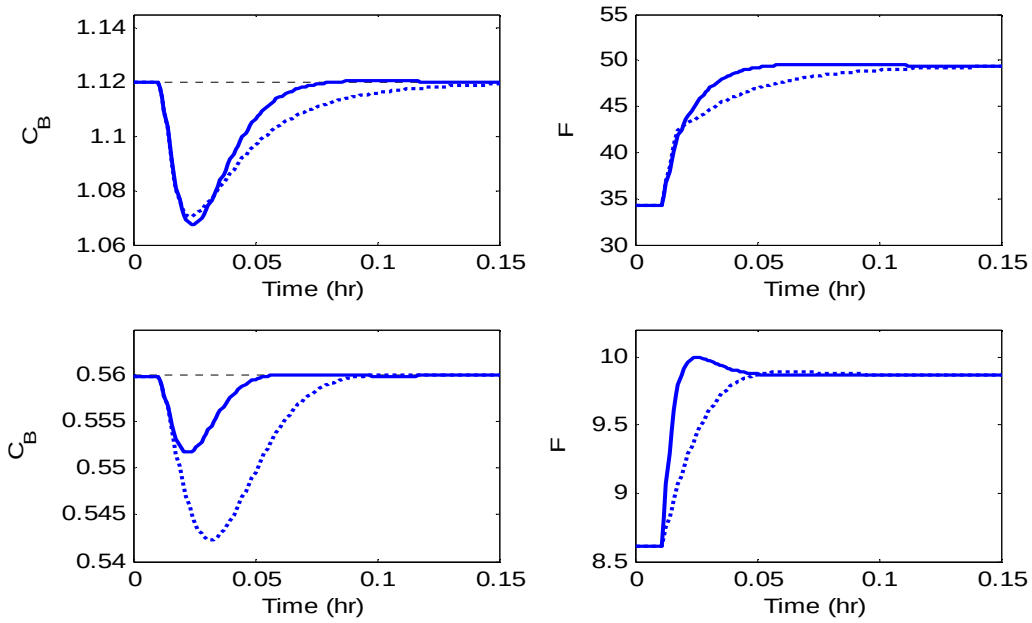


Figure 4.18 Closed-loop response for -10% change in C_{Af} . Solid line: adaptive PID design; dotted line: Hammerstein model based IMC design

4.5 Conclusions

By incorporating generalized Hammerstein model into controller design, adaptive IMC design method and adaptive PID control strategy are developed in this chapter. The IMC and PID parameters are adjusted adaptively by their respective parameter updating equations derived from steepest descent gradient method. Simulation results are presented to demonstrate the advantages of the proposed adaptive IMC and PID designs over the conventional Hammerstein model based IMC design.

CHAPTER 5

Control of Generalized Hammerstein Processes - MIMO Cases

5.1 Introduction

Multi-loop SISO controllers or decentralized controllers are often used to control multivariate chemical processes. The simple controller structure and the easiness to handle loop failure are the most attractive advantages of the decentralized control system. In contrast, the control of multivariable processes using full multivariable controllers involves a formidable cost in the development and maintenance of these controllers. In this chapter, adaptive PID controller design developed in Chapter 4 is extended to the control of multivariable system. In the proposed decentralized adaptive control strategy, controller parameters of individual control loops can be adjusted on-line according to their respective parameter updating equations and information provided by JITL algorithm. In the decentralized control context, an important issue is how process inputs and outputs are paired, i.e. control structure selection problem. In this research, control structure selection is tackled by the relative gain array (RGA) (Bristol, 1966) criterion prior to the proposed adaptive PID control design. The application of this strategy to two literature examples is undertaken to elucidate the design procedure.

5.2 Decentralized Adaptive PID Controller Design

For brevity of the notation used, the proposed controller design method will be presented for 2×2 generalized Hammerstein processes as depicted in Figure 5.1. Because the generalized Hammerstein model considered in Chapter 3 has separate nonlinearities as depicted in Figure 3.2, their respective reciprocals, NL_1^{-1} and NL_2^{-1}, are used in Figure 5.1 to remove the effect of static nonlinear part of the process so that the design of decentralized PID controller can be simply based on the linear dynamics of the process. Because on-line adaptation of PID controller parameters relies on their respective linear models, two JITL algorithms as indicated in Figure 5.1 are needed to provide necessary information for on-line tuning the PID parameters. This point will become clear in the following development.

It is noted that the static nonlinear functions of 2×2 generalized Hammerstein model can be obtained by the iterative identification procedure developed in Chapter 3. In addition, the linear dynamics part of 2×2 generalized Hammerstein model are described by:

$$y_1(k) = \alpha_{11}^k y_1(k-1) + \alpha_{12}^k y_2(k-1) + \beta_1^k v_1(k-1) \tag{5.1}$$

$$y_2(k) = \alpha_{21}^k y_1(k-1) + \alpha_{22}^k y_2(k-1) + \beta_2^k v_2(k-1) \tag{5.2}$$

The control laws of two PID controllers are given by:

$$v_i(k) = v_i(k-1) + \Delta v_i(k) \tag{5.3}$$

$$\Delta v_i(k) = w_{i,1}^k(k) e_i(k) + w_{i,2}^k(k) \Delta e_i(k) + w_{i,3}^k(k)(e_i(k) - 2e_i(k-1) + e_i(k-2)) \tag{5.4}$$

for $i = 1, 2$. $e_i(k)$ is the error between i-th process output and its set-point at the k-th

sampling instant, and $\Delta e_i(k) = e_i(k) - e_i(k-1)$. The PID parameters $w_{i,1}^k, w_{i,2}^k$ and $w_{i,3}^k$ are tuned online by updating formula to be derived in the sequel.

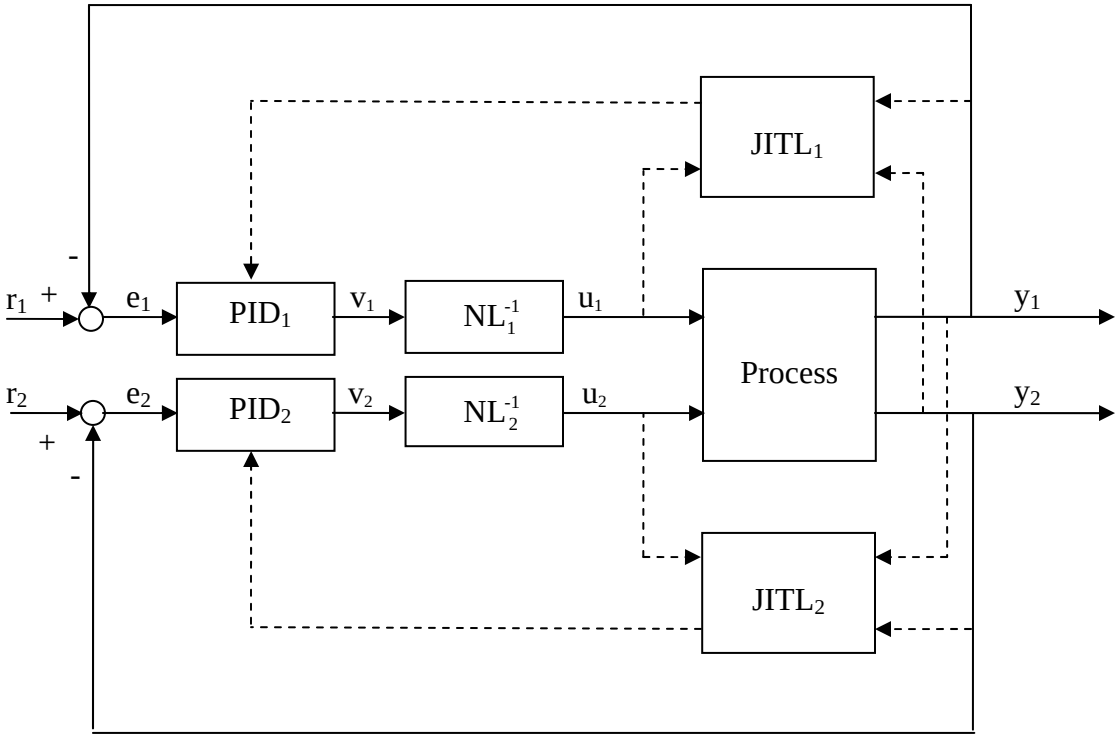


Figure 5.1 Decentralized adaptive PID control system for 2×2 generalized Hammerstein processes

The following objective function is used to update PID parameters:

$$\text{Min } J = (r_1(k+1) - \hat{y}_1(k+1))^2 + (r_2(k+1) - \hat{y}_2(k+1))^2 \qquad (5.5)$$

where $r_1(k+1)$ and $r_2(k+1)$ are the set-points, $\hat{y}_1(k+1)$ and $\hat{y}_2(k+1)$ are the predicted outputs of generalized Hammerstein model.

Since controller parameters are constrained to be positive or negative, the mapping function used in Chapter 4 are also employed here.

$$w_{i,j}^{k} = \begin{cases} e^{\varsigma_{i,j}(k)}, & \text{if} \quad w_{i,j}^{k} \geq 0 \\ -e^{\varsigma_{i,j}(k)}, & \text{if} \quad w_{i,j}^{k} < 0 \end{cases} \; ; \; i = 1,2, \; j = 1 \sim 3 \tag{5.6}$$

where $\varsigma_{i,j}$ is real number. In the same fashion of adaptive PID design discussed in Chapter 4, $\varsigma_{1,j}(k)$ and $\varsigma_{2,j}(k)$ will be adjusted on-line according to their respective updating equations and the informations provided by the JITL algorithms. Subsequently, PID parameters can be obtained by Eq. (5.6).

The updating equation for PID parameters are given as follows:

$$\begin{aligned} \varsigma_{i,j}(k+1) &= \varsigma_{i,j}(k) - \tilde{\eta}_{i,j} \frac{\partial J}{\partial \varsigma_{i,j}(k)} \\ &= \varsigma_{i,j}(k) + 2\eta_i (r_i(k+1) - \hat{y}_i(k+1)) \frac{z_{i,j}(k)}{1 + z_{i,j}^2(k)} \end{aligned} \tag{5.7}$$

for $i = 1,2$ and $j = 1 \sim 3$. The parameters η_1 and η_2 are the learning rates and

$$\tilde{\eta}_{i,j} = \frac{\eta_i}{1 + z_{i,j}^2} \tag{5.8}$$

$$z_{i,1}(k) = \frac{\partial \hat{y}_i(k+1)}{\partial v_i(k)} \frac{\partial v_i(k)}{\partial w_{i,1}^{k}} \frac{\partial w_{i,1}^{k}}{\partial \varsigma_{i,1}(k)} = \beta_i^{k+1} e_i(k) e^{\varsigma_{i,1}(k)} \tag{5.9}$$

$$z_{i,2}(k) = \beta_i^{k+1} \Delta e_i(k) e^{\varsigma_{i,2}(k)} \tag{5.10}$$

$$z_{i,3}(k) = \beta_i^{k+1} \left(e_i(k) - 2e_i(k-1) + e_i(k-2) \right) e^{\varsigma_{i,3}(k)} \tag{5.11}$$

where β_1^{k+1} and β_2^{k+1} are the model parameters of two linear models identified by the identification method developed in Chapter 3.

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

1. Initialize $w_{1,j}^{k}$ and $w_{2,j}^{k}$ $(j = 1 \sim 3)$ and learning rate parameters η_1 and η_2;

2. Given the current errors $e_1(k)$ and $e_2(k)$, compute $v_1(k)$ and $v_2(k)$ from Eq. (5.3) and calculate manipulated variables $u_1(k)$ and $u_2(k)$;
3. Update two linear models by using the most current process data and JITL algorithms and subsequently adjust $\varsigma_{1,j}(k)$ and $\varsigma_{2,j}(k)$ according to Eq. (5.7);
4. Obtain PID parameters for the next sampling instant using Eq. (5.6) and go to step 2.

5.3 Examples

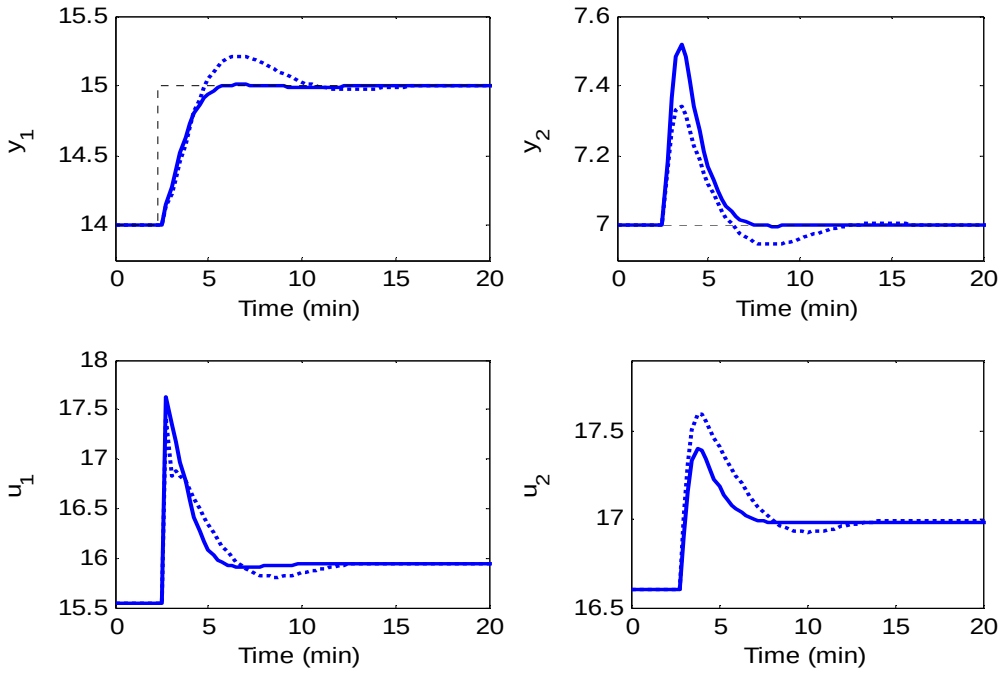
Example 1 Considering the pH neutralization process previously studied in Chapter 3, where the control objective is to control the tank level h_1 and effluent pH by manipulating the base flow rate and acid flow rate. Prior to the proposed decentralized controller design, the pairing between process outputs and inputs needs to be considered. After performing the RGA analysis, tank level h_1 (y_1) is controlled by base flow rate (u_1) and effluent pH (y_2) is controlled by acid flow rate (u_2).

To evaluate the servo performance of two controllers, ± 1 step change in the set-point of y_1 as indicated by the dashed lines in Figure 5.2 and ± 2 set-point changes in y_2 as shown in Figure 5.3 are considered. To design the proposed adaptive decentralized PID controller, the initial controller parameters for the first control loop are $w_{1,1}^0 = 0.6$, $w_{1,2}^0 = 4.5$, and $w_{1,3}^0 = 2$, and for the second control loop $w_{2,1}^0 = -0.09$, $w_{2,2}^0 = -1$, and $w_{2,3}^0 = -0.5$, respectively, whereas the respective learning rates are fixed as $\eta_1 = 1.6$ and $\eta_2 = 1.85$. For comparison purposes, a decentralized PID controller is also design based on the 2×2 Hammerstein model identified in Chapter 3. The controller parameters for

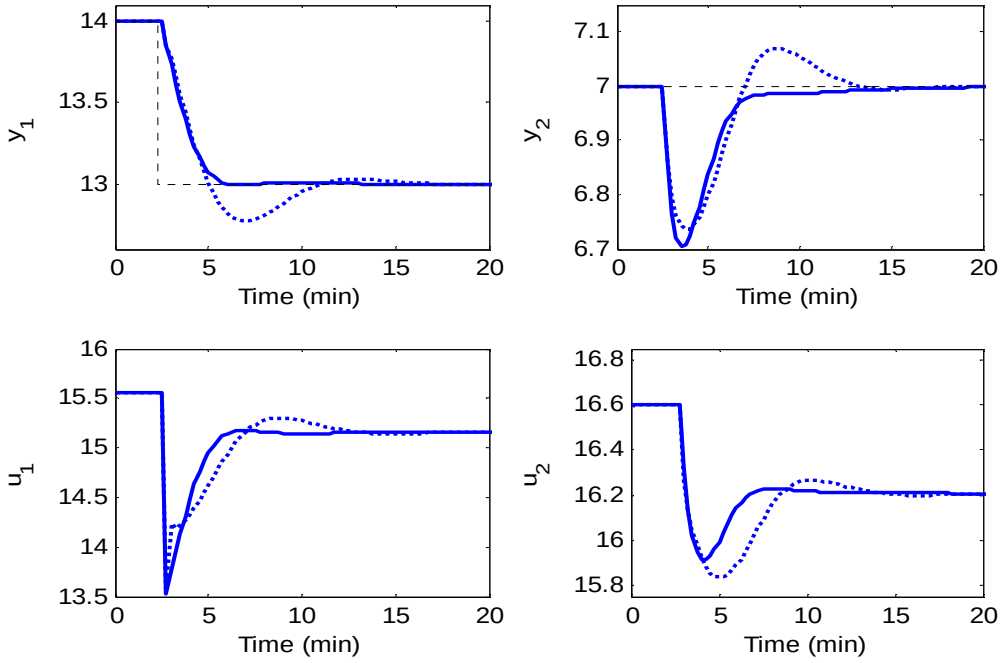
the first control loop are $w_{1,1} = 0.15$, $w_{1,2} = 1.1$, and $w_{1,3} = 0.5$, and for the second control loop $w_{2,1} = -0.08$, $w_{2,2} = -0.8$, and $w_{2,3} = -0.1$. As can be seen from Figures 5.2 and 5.3, decentralized adaptive PID controller has superior control performance than that achieved by Hammerstein model based PID design, as also verified by comparing their respective MAEs given in Table 5.1. In addition, their respective load performance for the step change in buffer flow rate from 0.55 to 0.2 is compared in Figure 5.4 and Table 5.1. It is evident that the proposed decentralized PID controller outperforms its counterpart designed based on the Hammerstein model.

Table 5.1 Summary of MAEs for closed-loop responses in Figures 5.2 to 5.4

	Hammerstein model based PID design		Adaptive PID design	
	y_1	y_2	y_1	y_2
+1 set-point change in y_1	8.94×10^{-2}	3.85×10^{-2}	5.56×10^{-2}	4.39×10^{-2}
-1 set-point change in y_1	9.55×10^{-2}	4.42×10^{-2}	5.72×10^{-2}	3.66×10^{-2}
+2 set-point change in y_2	6.83×10^{-2}	1.14×10^{-1}	7.13×10^{-2}	9.15×10^{-2}
-2 set-point change in y_2	1.81×10^{-1}	1.93×10^{-1}	1.19×10^{-1}	1.63×10^{-1}
-0.35 step change in buffer flow rate	3.50×10^{-2}	1.73×10^{-2}	2.37×10^{-2}	1.54×10^{-2}

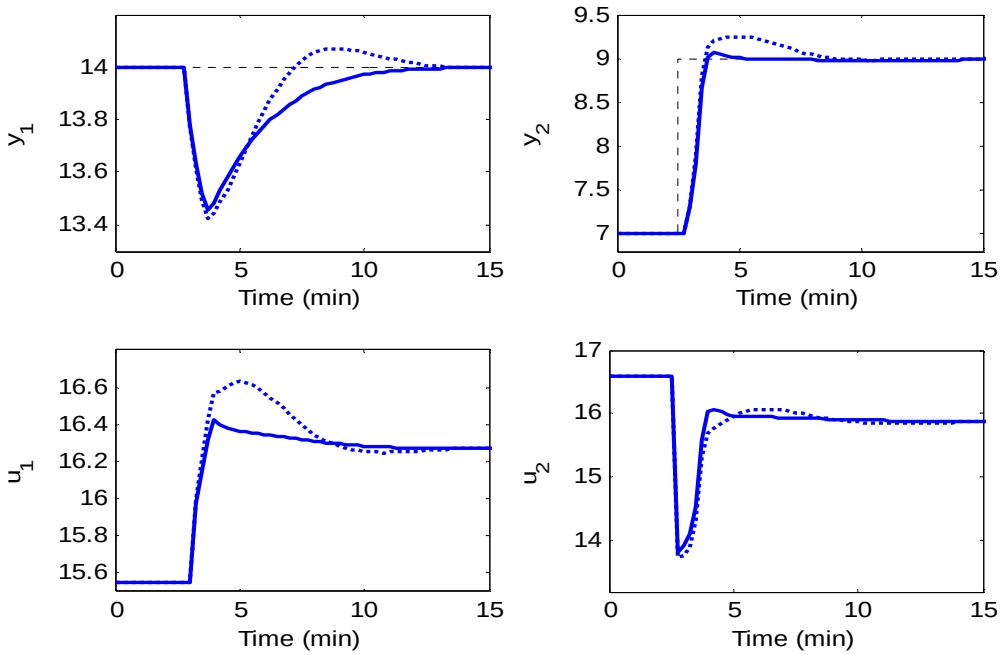


(a)

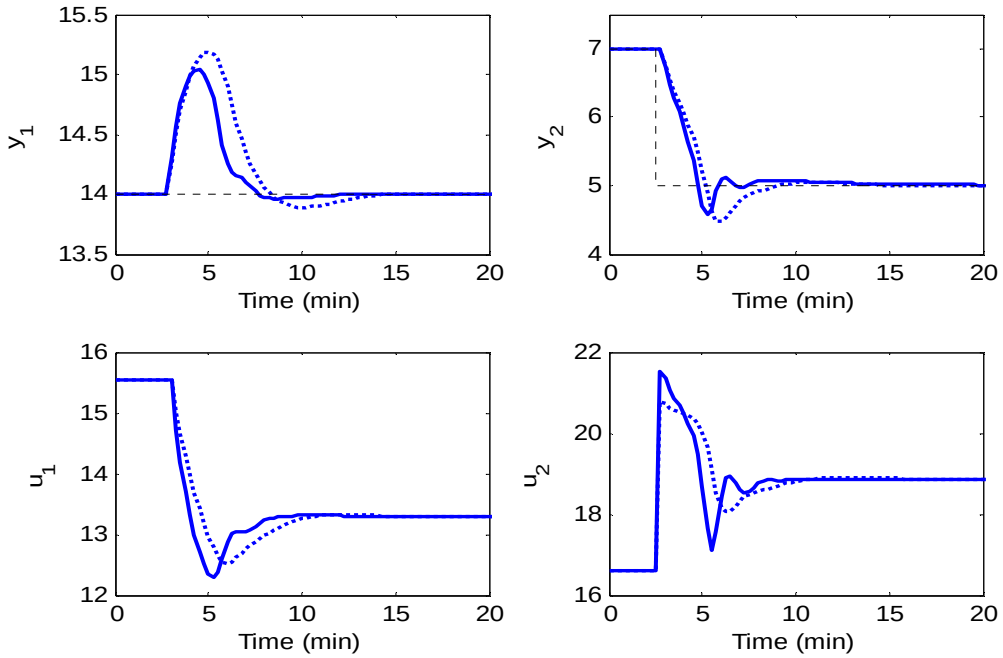


(b)

Figure 5.2 Closed-loop response for set-point changes in y_1: (a) 14 to 15, (b) 14 to 13. Solid line: adaptive PID design; dotted line: Hammerstein model based PID design



(a)



(b)

Figure 5.3 Closed-loop response for set-point changes in y_2: (a) 7 to 9, (b) 7 to 5. Solid line: adaptive PID design; dotted line: Hammerstein model based PID design

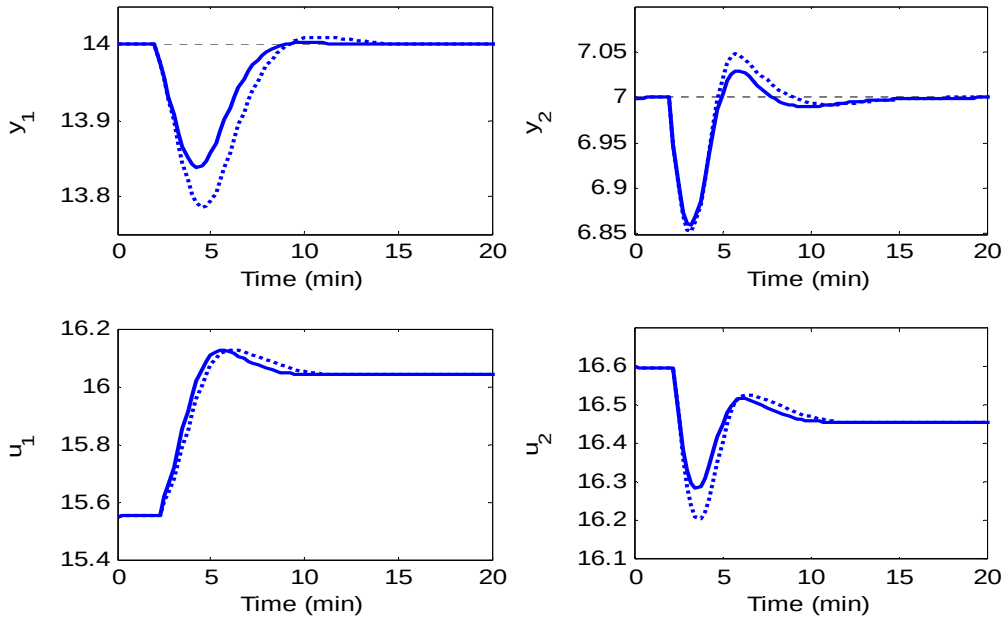


Figure 5.4 Closed-loop response for step disturbance in buffer stream. Solid line: adaptive PID design; dotted line: Hammerstein model based PID design

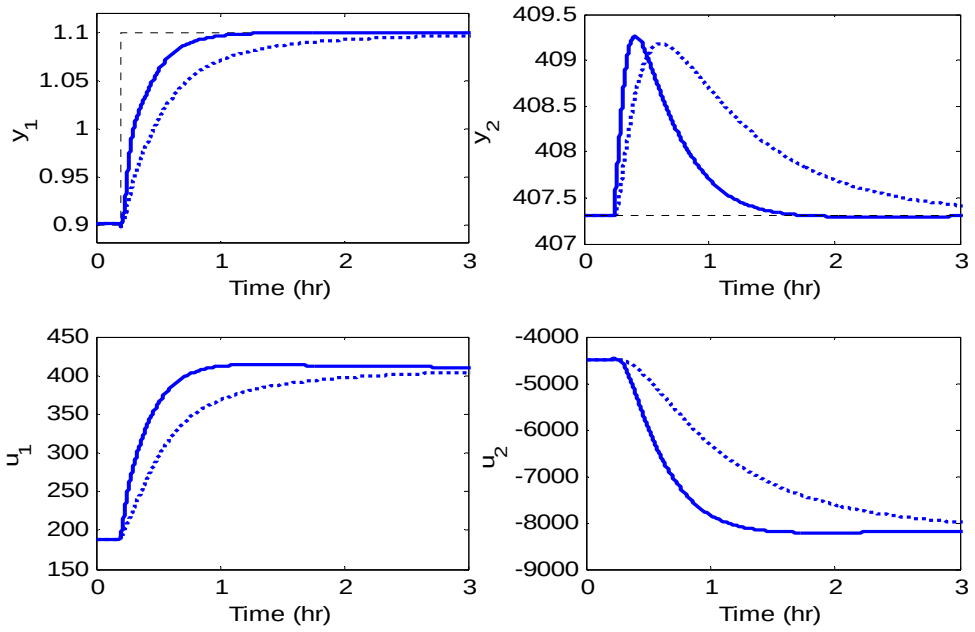
Example 2 The second example focuses on the control of the non-isothermal van de Vusse reactor as described in Chapter 3. The process outputs are the outlet concentration of component B (C_B) and reactor temperature (T) and process inputs are the reactor flow rate F and and the external heat exchanger duty Q_w. After conducting the RGA analysis, C_B (y_1) is controlled by F (u_1) and T (y_2) is controlled by Q_w (u_2) in the decentralized control system to be designed in what follows.

In the simulation studies given in Figure 5.5 to 5.7, the controller parameters for adaptive PID controller are initialized as: $w_{1,1}^0 = 0.05$, $w_{1,2}^0 = 0.01$, $w_{1,3}^0 = 0.5$, $w_{2,1}^0 = 0.7$, $w_{2,2}^0 = 0.01$, $w_{2,3}^0 = 0.7$, and learning rates are specified as $\eta_1 = 1$ and $\eta_2 = 1.5$. For PID design based on the Hammerstein model, the controller parameters are designed as: $w_{1,1} = 0.013$, $w_{1,2} = 0.01$, $w_{1,3} = 0.05$, $w_{2,1} = 0.3$, $w_{2,2} = 1.5$, $w_{2,3} = 0.1$. Figures 5.5 and

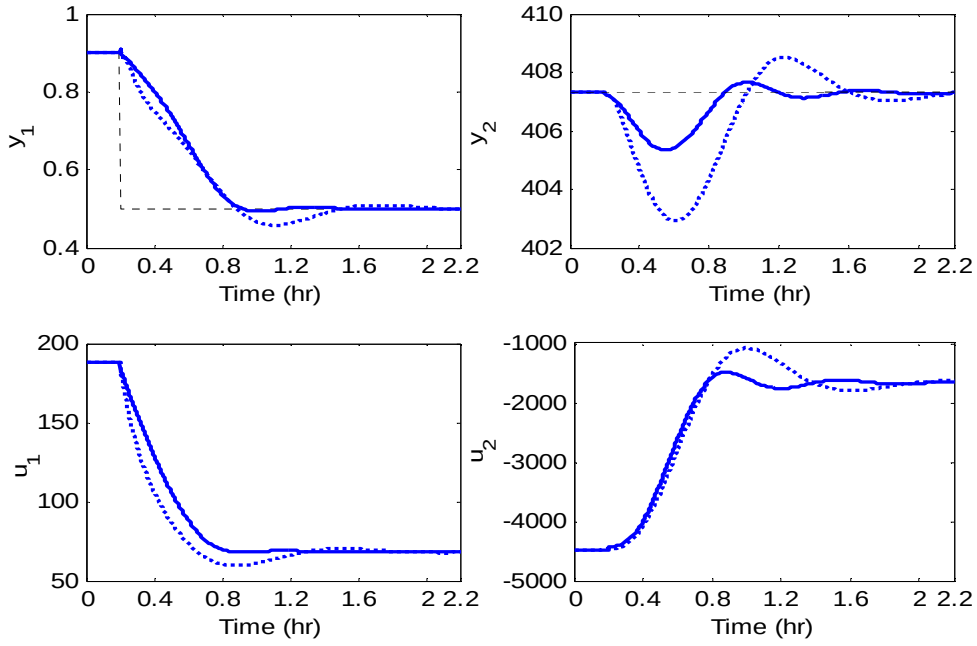
5.6 show that the adaptive PID controller has superior servo response than the PID design based on the Hammerstein model. To evaluate their disturbance rejection capabilities, a step change in the inlet concentration C_{Af} from its nominal value of 5.1 to 6.6 is considered. As can be seen from Figure 5.7, adaptive PID controller has better performance than its conventional counterpart. Table 5.2 summarizes the MAEs of the aforementioned simulation studies.

Table 5.2 Summary of MAEs for closed-loop responses in Figures 5.5 to 5.7

	Hammerstein model based PID design		Adaptive PID design	
	y_1	y_2	y_1	y_2
+0.22 set-point change in y_1	3.81×10^{-2}	7.64×10^{-1}	1.15×10^{-2}	3.05×10^{-1}
-0.4 set-point change in y_1	4.61×10^{-2}	8.23×10^{-1}	4.67×10^{-2}	2.87×10^{-1}
+10 set-point change in y_2	1.68×10^{-2}	1.19×10^{0}	7.49×10^{-3}	5.17×10^{-1}
-10 set-point change in y_2	3.66×10^{-2}	1.74×10^{0}	3.27×10^{-2}	7.99×10^{-1}
+1.5 step change in C_{Af}	2.65×10^{-2}	8.22×10^{-1}	1.78×10^{-2}	3.72×10^{-1}

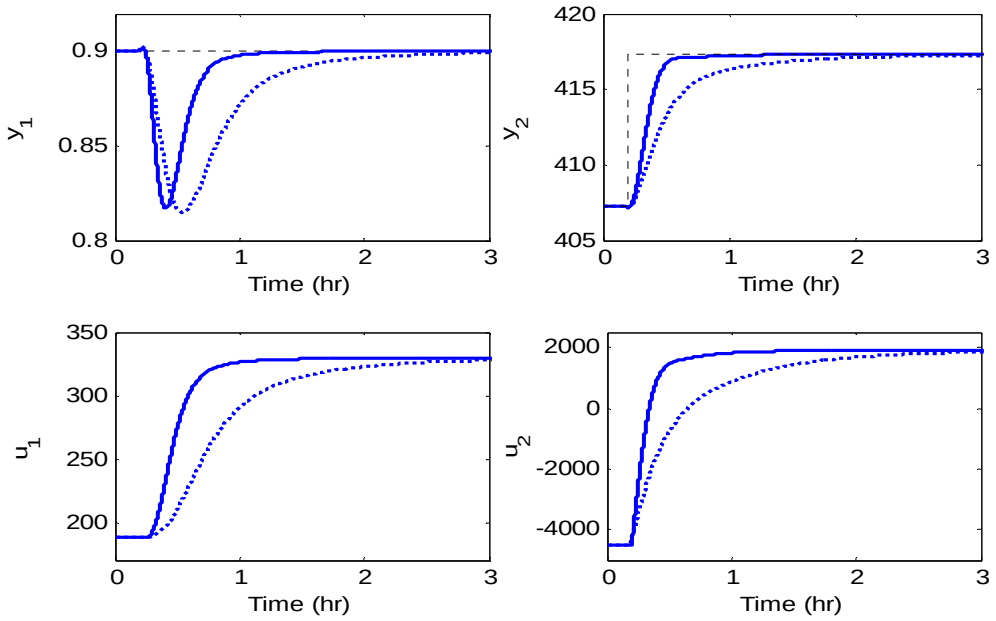


(a)

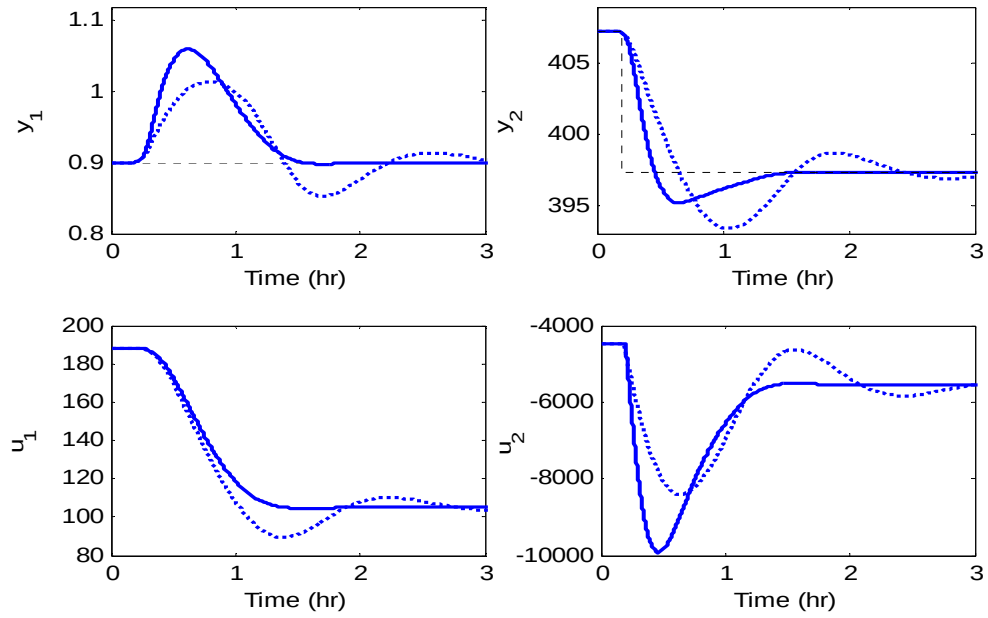


(b)

Figure 5.5 Closed-loop response for set-point changes in y_1 : (a) 0.9 to 1.12, (b) 0.9 to 0.5. Solid line: adaptive PID design; dotted line: Hammerstein model based PID design



(a)



(b)

Figure 5.6 Closed-loop response for set-point changes in y_2 : (a) 407.3 to 417.3, (b) 407.3 to 397.3. Solid line: adaptive PID design; dotted line: Hammerstein model based PID design

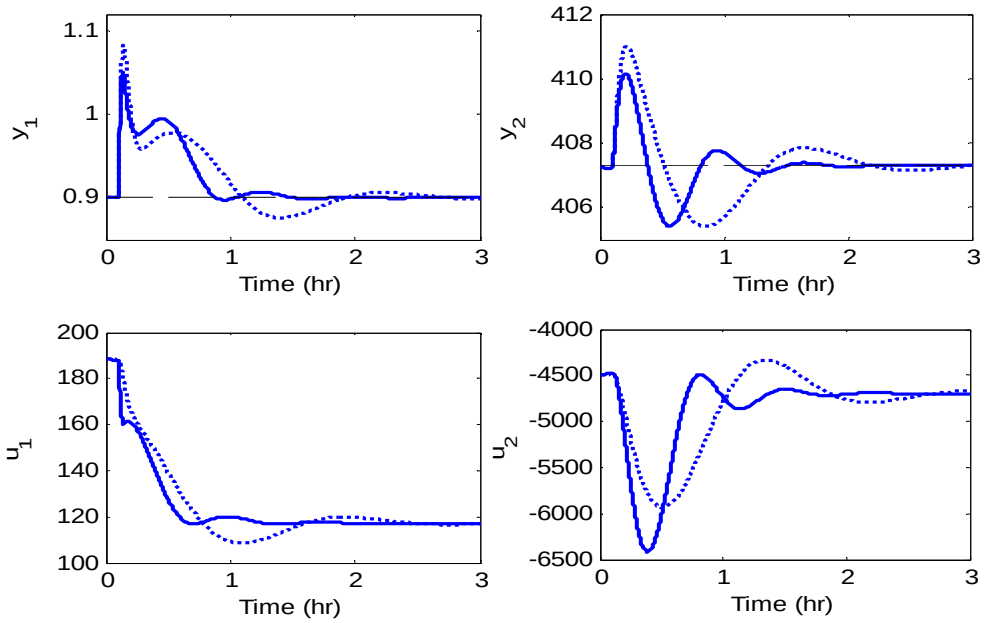


Figure 5.11 Closed-loop responses for step disturbance in C_{Af}. Solid line: adaptive PID design; dotted line: Hammerstein model based PID design

5.4 Conclusions

In this chapter, the previously developed SISO adaptive PID control strategy is extended for adaptive decentralized PID design. Similar to the earlier study in Chapter 4, the inversion of static nonlinear functions is employed to simplify the resulting controller design. By using the parameter updating equation derived and information provided by JITL algorithm, the proposed controller design is evaluated through simulation studies to show better control performance than its counterpart designed based on the Hammerstein model.

CHAPTER 6

Conclusions

In this research work, generalized SISO and MIMO Hammerstein models are proposed. These new models consist of a static nonlinear function in series with time-varying linear dynamics. Consequently, generalized Hammerstein model can be used for modeling the Hammerstein-like processes whose linear dynamics vary over the operating space. Iterative identification procedures for generalized SISO and MIMO Hammerstein models are developed. Unlike the identification of conventional Hammerstein model, only the polynomial function obtained by the proposed identification method will be retained as the static nonlinear part of generalized Hammerstein model. As a result, on-line application of generalized Hammerstein model requires the computation of linear model by using the JITL technique and current process information. Simulation results show that generalized Hammerstein model has better modeling accuracy than the conventional Hammerstein model.

By using the generalized Hammerstein model as the process model, an adaptive IMC control strategy is developed. In the proposed adaptive IMC control design, two static nonlinear blocks, NL and NL^{-1}, are employed to make the resulting IMC design problem amenable to linear IMC analysis. The controller parameters are adjusted on-line

by using gradient descent learning algorithm and the information provided by JITL. Simulation results show that the proposed adaptive IMC design can provide better performance over that designed based on conventional Hammerstein model. Following the design concept of adaptive IMC design, adaptive PID control strategies are developed for both SISO and MIMO generalized Hammerstein processes. Again, PID parameters are adjusted on-line by their respective parameter updating equations developed in Chapters 4 and 5. Simulation results illustrate that the proposed adaptive PID design has better set-point tracking and disturbance rejection performance than its counterpart based on the Hammerstein model.

The suggested future work includes the following points. Firstly, static nonlinear part of the generalized Hammerstein model can be identified by the neural network owing to its ability to model a nonlinear function to any arbitrary accuracy. Furthermore, neural network can be applied straightforwardly to the MIMO generalized Hammerstein processes whose nonlinear part is better described by the combined nonlinear function as illustrated in Figure 3.1. Lastly, the control strategies developed in this thesis does not address the input saturation problem. One remedy to overcome this problem would be to develop a model predictive controller (MPC) based on the generalized Hammerstein model because MPC is one of the few methods for handling constraints and other issues like process interaction in a systematic design framework. Consequently, the resulting MPC has potential to give better control performance than the adaptive decentralized PID controller developed in this thesis.

References

Aha, D. W., D. Kibler, and M. K. Albert, Instant-Based Learning Algorithms, *Machine Learning*, 6(1), pp.37-66, 1991.

Al-Duwaish, H. and M. N. Karim, A New Method for the Identification of Hammerstein Model, *Automatica*, 33, pp.1871-1875, 1997.

Astrom K. J., Theory and Application of Adaptive Control-A Survey, *Automatica*, 19, pp. 471-486, 1983.

Astrom K. J. and B. Wittenmark, *Adaptive Control*, *Addison-Wesley*, 1989.

Atkeson C. G., A. W. Moore, and S. Schaal, Locally Weighted Learning, *Artificial Intelligence Review*, 11, pp. 11-73, 1997.

Bontempi, G., M. Birattari, and H. Bersini, Lazy Learning for Local Modelling and Control Design, *Int. J. Control*, 72, pp. 643-658, 1999.

Bristol, E. H., On a New Measure of Interaction for Multivariable Process Control, *IEEE Trans. on Automatic Control*, 11, pp. 133-134, 1966

Chang, F. H. and R. Luus, A Noniterative Method for Identification Using the Hammerstein Model, *IEEE Trans. on Automatic Control*, 16, pp. 464-468, 1971.

Chen, C. T., Direct Adaptive Control of Chemical Process Systems, *Ind. Eng. Chem. Res.*, 40, pp. 4121-4140, 2001.

Cheng, C. and M. S. Chiu, A New Data-based Methodology for Nonlinear Process Modeling, *Chemical Engineering Science*, 59, pp. 2801-2810, 2004.

Doyle, F. J., B. A. Ogunnaike, and R. K. Pearson, Nonlinear Model-based Control Using Second-order Volterra models, *Automatica*, 31, pp. 697–714, 1995.

Eskinat E., S. H. Johnson, and W. L. Luyben, Use of Hammerstein Models in Identification of Nonlinear Systems, *AIChE J.*, 37, pp. 255–268, 1991.

Fruzzetti K. P., A. Palazoglu, and K. A. McDonald, Nonlinear Model Predictive Control Using Hammerstein Models, *J. Proc. Control*, 7, pp. 31–41, 1997.

Gallman, P., A Comparison of Two Hammerstein Model Identification Algorithms, *IEEE Trans. on Automatic Control*, 20, pp. 771-773, 1976.

Greblicki, W., and M. Pawlak, Identification of Discrete Hammerstein Systems Using Kernel Regression Estimate, *IEEE Trans. on Automatic Control*, 31, pp. 74-77, 1986.

Hahn, J. and T. F. Edgar, A Gramian Based Approach to Nonlinearity Quantification and Model Classification, *Ind. Eng. Chem. Res.*, 40, pp. 5724–5731, 2001.

Harris, R. H. and A. Palazoglu, Studies on the Analysis of Nonlinear Process via Functional Expansions-III: Controller Design, *Chemical Engineering Science*, 53, pp. 4005-4022, 1998.

Henson, M. A. and D. E. Seborg, Adaptive Nonlinear Control of a pH Neutralization Process, *IEEE Trans. on Control Systems Technology*, 2, pp. 169-182, 1994.

Hwang, C. L. and J. C. Hsu, Nonlinear Control Design for a Hammerstein Model system, *IEE Proc.-Control Theory Appl.*, 142, pp. 277-285, 1995.

Lakshminarayanan S., S. L. Shah, and K. Nandakumar, Identification of Hammerstein Models Using Multivariate Statistical Tools, *Chemical Engineering Science*, 50, pp. 3599-3613, 1995.

Lakshminarayanan, S., S. L. Shah and K. Nandakumar, Modeling and Control of Multivariable Process: The Dynamic Projection to Latent Structures Approach, *AIChE J.*, 43, pp. 2307-2323, 1997.

Lang, Z. Q., A Nonparametric Polynomial Identification Algorithm for the Hammerstein System, *IEEE Trans. on Automatic Control*, 42, pp. 1435-1441, 1997.

Ling W. M., and D. E. Rivera, Control Relevant Model Reduction of Volterra Series Models, *J. Proc. Control*, 8, pp. 79-88, 1998.

Ling, W. M. and D. E. Rivera, A Methodology for Control-relevant Nonlinear System Identification Using Restricted Complexity Models, *J. Proc. Control*, 11, pp. 209-222, 2001.

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

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

Narendra, K. S. and P. G. Gallman, An Iterative Method for the Identification of the Nonlinear systems Using the Hammerstein Model, *IEEE Trans. on Automatic Control*, 12, pp. 546-550, 1966.

Pearson, R. K. and M. Pottmann, Gray-box Identification of Block-oriented Nonlinear Models, *J. Proc. Control*, 10, pp. 301-315, 2000.

Pottmann, M., H. Unbehauen, and D. E. Seborg, Application of General Multi-Model Approach for Identification of Highly Nonlinear Processes, *Int. J. Control*, 57, pp. 97-120 , 1993.

Stack A. J. and F. J. Doyle, Application of a Control-Law Nonlinearity Measure to the Chemical Reactor Analysis, *AIChE J.*, 43, pp. 425-439, 1997.

Su H. T. and T. J. McAvoy, Integration of Multilayer Percepton Networks and Linear Dynamic Models: A Hammerstein Modeling Approach, *Ind. Eng. Chem. Res.*, 26, pp. 1927–1936, 1993.

Sung, S. W., System Identification Method for Hammerstein Processes, *Ind. Eng. Chem. Res.*, 41, pp. 4295-4302, 2002.