

Fuzzy Model Predictive Control

Y. L. Huang, Helen H. Lou, J. P. Gong, and Thomas F. Edgar

Abstract—A highly nonlinear system controlled by a linear model predictive controller (MPC) may not exhibit a satisfactory dynamic performance. This has led to the development of a number of nonlinear MPC (NMPC) approaches that permit the use of first principles-based nonlinear models. Such models can be accurate over a wide range of operating conditions, but may be difficult to develop for many industrial cases. Moreover, an NMPC usually requires tremendous computational effort that may prohibit its on-line applications. In this paper, a fuzzy model predictive control (FMPC) approach is introduced to design a control system for a highly nonlinear process. In this approach, a process system is described by a fuzzy convolution model that consists of a number of quasi-linear fuzzy implications (FIs). In controller design, prediction errors and control energy are minimized through a two-layered iterative optimization process. At the lower layer, optimal local control policies are identified to minimize prediction errors in each subsystem. A near optimum is then identified through coordinating the subsystems to reach an overall minimum prediction error at the upper layer. The two-layered computing scheme avoids extensive on-line nonlinear optimization and permits the design of a controller based on linear control theory. The efficacy of the FMPC approach is demonstrated through three examples.

Index Terms—Control system design, fuzzy logic, model predictive control.

I. INTRODUCTION

MODEL predictive control (MPC) has emerged as one of the most attractive control techniques in the chemical and petrochemical industries during the past decade. In MPC, a process dynamic model is used to predict future outputs over a prescribed period [12], [13]. Dynamic matrix control [2], model algorithmic control [11], and simplified model predictive control [1] are excellent examples that have been applied to various industrial processes [3], [4].

Continuous and batch processes in chemical and petrochemical plants are inherently nonlinear and many of them are highly nonlinear. For a highly nonlinear system, a linear MPC algorithm may not give rise to satisfactory dynamic performance. Recently, several researchers [9] have developed nonlinear model predictive control (NMPC) algorithms that accept various kinds of nonlinear models such as nonlinear ordinary differential/algebraic equations, partial differential/al-

gebraic equations, integro-differential equations, and delay equation models. Such models can be accurate over a wide range of operating conditions. However, these models, usually based on the first principles, are very difficult to develop for many industrial cases. Moreover, an NMPC incorporating a nonlinear model may require tremendous computational effort for optimization; this may disqualify it for on-line applications. If a nonlinear process can be precisely described by a set of linear submodels in some way, then the design of a model predictive controller can be greatly simplified.

Reference [15] introduced a novel fuzzy logic-based modeling methodology, where a nonlinear system is divided into a number of linear or nearly linear subsystems. A quasi-linear empirical model is then developed by means of fuzzy logic for each subsystem. The model is a rule-based fuzzy implication (FI). The whole process behavior is characterized by a weighted sum of the outputs from all quasi-linear FIs. The methodology facilitates the development of a nonlinear model that is essentially a collection of a number of quasi-linear models regulated by fuzzy logic. It also provides an opportunity to simplify the design of model predictive controllers.

Reference [10] developed an MPC algorithm using a Takagi–Sugeno (T–S) type model. However, tremendous difficulties have been found in tuning controller parameters since the algorithm requires frequent model updating in control. More recently, [8] proposed an approach for designing a fuzzy model-based state–space feedback controller. A T–S type model is the basis of their fuzzy model. However, they essentially treated the fuzzy model as a set of conventional piecewise linear models. Thus, the uniqueness of a Takagi–Sugeno-type model exhibiting soft transition through any operating regions is lost, causing deterioration in the closed-loop dynamic performance of a system.

In this paper, a fuzzy model predictive control (FMPC) approach is introduced to design a control system for a highly nonlinear process system. The approach utilizes the Takagi–Sugeno modeling methodology to generate a fuzzy convolution model. With this model, a novel hierarchical control design approach is described. Three case studies are provided to demonstrate the attractiveness of the FMPC.

II. FUZZY CONVOLUTION MODEL

Consider a single-input single-output (SISO) highly nonlinear system S . The system is decomposed into p subsystems such that each subsystem demonstrates a linear or nearly linear behavior. By Takagi–Sugeno’s modeling methodology [15], a fuzzy quasi-linear model, R^i , or FI, need be developed for each subsystem. In such a model, the cause–effect relationship between control u and output y at the sampling time n is established in a discrete time representation. The subsystems

Manuscript received February 12, 1999; revised July 12, 2000. This work was supported in part by ACS-PRF, the National Science Foundation under Contract CTS 9 414 944, the American Chemical Society—Petroleum Research Foundation, the Michigan Space Grant Program/NASA, and the Institute of Manufacturing Research at Wayne State University.

Y. L. Huang, H. H. Lou, and J. P. Gong are with the Department of Chemical Engineering and Materials Science, Wayne State University, Detroit, MI 48202 USA (e-mail: yhuang@che.eng.wayne.edu).

T. F. Edgar is with the Department of Chemical Engineering, University of Texas, Austin, TX 78712 USA.

Publisher Item Identifier S 1063-6706(00)10693-9.

are defined in the fuzzy regions, $R^i, i = 1, 2, \dots, p$. Each fuzzy region is characterized by the following Cartesian product:

$$R^i = y(n) \times y(n-1) \times \dots \times y(n-m+1) \\ \times u(n) \times u(n-1) \times \dots \times u(n-l+1) \quad (1)$$

where

$$y(n-j) \quad \text{measured output } y \text{ at time } n-j;$$

$$u(n-j) \quad \text{measured input } u \text{ at time } n-j.$$

An FI is rule based on and consists of a set of symbolic antecedents in the IF part (premise) and a linear numerical expression in the THEN part (consequence). Each FI is generated based on a system response to an impulse signal [5], [6]. Thus, it can be called a fuzzy convolution submodel that has the following structure:

$$R^i: \text{IF } y(n) \text{ is } A_0^i, y(n-1) \text{ is } A_1^i, \dots, y(n-m+1) \\ \text{is } A_{m-1}^i, \text{ and} \\ u(n) \text{ is } B_0^i, u(n-1) \text{ is } B_1^i, \dots, u(n-l+1) \\ \text{is } B_{l-1}^i$$

$$\text{THEN } y^i(n+1) = y(n) + \sum_{j=1}^T h_j^i \Delta u(n+1-j) \quad (2)$$

where

$$A_j^i \quad \text{fuzzy set corresponding to output } y(n-j) \text{ in the} \\ \text{ith FI;} \\ B_j^i \quad \text{fuzzy set corresponding to input } u(n-j) \text{ in the } i\text{th} \\ \text{FI;} \\ h_j^i \quad \text{impulse response coefficient in the } i\text{th FI;} \\ T \quad \text{model horizon;} \\ \Delta u(n) \quad \text{difference between } u(n) \text{ and } u(n-1).$$

A complete fuzzy convolution model for the system consists of p FIs. The system output $y(n+1)$ is inferred as a weighted average value of the outputs estimated by all FIs, i.e.,

$$y(n+1) = \frac{\sum_{j=1}^p w^j y^j(n+1)}{\sum_{j=1}^p w^j} \quad (3)$$

where w^j is the truth value for the j th FI; it can be calculated based on the fuzzy sets in the IF part, i.e.,

$$w^j = \bigwedge_i A_i^j \bigwedge_k B_k^j \quad (4)$$

(3) can be simplified as

$$y(n+1) = \sum_{j=1}^p \omega^j y^j(n+1) \quad (5)$$

where

$$\omega^j = \frac{w^j}{\sum_{j=1}^p w^j}. \quad (6)$$

Apparently

$$\sum_{j=1}^p \omega^j = 1. \quad (7)$$

Note that in each fuzzy convolution submodel (2), output $y^j(n+1)$ is evaluated by utilizing $y(n)$ rather than $y^j(n)$ in order to minimize an estimation error.

III. FUZZY MODEL PREDICTIVE CONTROL

The design goal of an FMPC is to minimize the predictive error between an output and a given reference trajectory in the next N_y steps through the selection of N_u -step optimal control policies.

A. Problem Formulation

The optimization problem can be formulated as

$$\min_{\Delta u(n), \Delta u(n+1), \dots, \Delta u(n+N_u)} J(n) \quad (8)$$

and

$$J(n) = \sum_{i=1}^{N_y} \mu_i (\hat{y}(n+i) - y^r(n+i))^2 + \sum_{i=1}^{N_u} \nu_i \Delta u(n+i)^2 \quad (9)$$

where

$$\mu_i \text{ and } \nu_i \quad \text{respectively, the weighting factors for the predic-} \\ \text{tion error and control energy;} \\ \hat{y}(n+i) \quad \text{ith step output prediction;} \\ y^r(n+i) \quad \text{ith step reference trajectory;} \\ \Delta u(n+i) \quad \text{ith step control action.}$$

The objective function is subject to a fuzzy convolution model, which consists of p FIs as shown in (2).

In (9), the control policy, $\Delta u(n+i), i = 1, 2, \dots, N_u$, can be developed by first generating p sets of local control policies, $\Delta u^j(n+i), i = 1, 2, \dots, N_u, j = 1, 2, \dots, p$, where p is the total number of subsystems. The weighted sum of the local control policies gives the overall control policy. That is

$$\Delta u(n+i) = \sum_{j=1}^p \omega^j \Delta u^j(n+i). \quad (10)$$

In the above equation, the weight for the j th control action is the same as that for the j th submodel. This is reasonable since the contribution of the output estimated by the j th FI to the overall process evaluation should be considered the same as that by the j th local control action to the overall system controller. Substituting (5) and (10) into (9) yields

$$J(n) = \sum_{i=1}^{N_y} \mu_i \left(\sum_{j=1}^p (\omega^j (\hat{y}^j(n+i) - y^r(n+i))) \right)^2 \\ + \sum_{i=1}^{N_u} \nu_i \left(\sum_{j=1}^p \omega^j \Delta u^j(n+i) \right)^2. \quad (11)$$

The minimization of this objective function requires extensive computational effort since various interactions among subsystems exist. To simplify the computation, an alternative objective function is proposed as a satisfactory approximation of (11).

According to the Cauchy inequality, the following relationships hold:

$$\begin{aligned} & \left(\sum_{j=1}^p \omega^j (\hat{y}^j(n+i) - y^r(n+i)) \right)^2 \\ & \leq p \sum_{j=1}^p (\omega^j (\hat{y}^j(n+i) - y^r(n+i)))^2 \end{aligned} \quad (12)$$

$$\left(\sum_{j=1}^p \omega^j \Delta u^j(n+i) \right)^2 \leq p \sum_{j=1}^p (\omega^j \Delta u^j(n+i))^2. \quad (13)$$

The inequalities show that the sum of the weighted squared errors can be the basis for establishing an upper bound of the original objective function. This allows us to define the following alternative objective function:

$$\begin{aligned} \tilde{J}(n) = & \sum_{i=1}^{N_y} \sum_{j=1}^p \mu_i (\omega^j (\hat{y}^j(n+i) - y^r(n+i)))^2 \\ & + \sum_{i=1}^{N_u} \sum_{j=1}^p \nu_i (\omega^j \Delta u^j(n+i))^2. \end{aligned} \quad (14)$$

Note that $p\tilde{J}(n)$ is greater than $J(n)$. However, the nature of minimization of $\tilde{J}(n)$ is the same as that of $p\tilde{J}(n)$. For simplicity, $\tilde{J}(n)$ is used as the objective function in the succeeding text. Equivalently, it can be also written as

$$\begin{aligned} \tilde{J}(n) = & \sum_{j=1}^p \left((\omega^j)^2 \left(\sum_{i=1}^{N_y} \mu_i (\hat{y}^j(n+i) - y^r(n+i))^2 \right. \right. \\ & \left. \left. + \sum_{i=1}^{N_u} \nu_i (\Delta u^j(n+i))^2 \right) \right) \end{aligned} \quad (15)$$

or, more clearly, the optimization problem can be defined as

$$\begin{aligned} & \min_{\Delta u(n), \Delta u(n+1), \dots, \Delta u(n+N_u)} \tilde{J}(n) \\ & = \min_{\Delta u(n), \Delta u(n+1), \dots, \Delta u(n+N_u)} \sum_{j=1}^p (\omega^j)^2 \tilde{J}^j(n) \end{aligned} \quad (16)$$

where

$$\begin{aligned} \tilde{J}^j(n) = & \sum_{i=1}^{N_y} \mu_i (\hat{y}^j(n+i) - y^r(n+i))^2 \\ & + \sum_{i=1}^{N_u} \nu_i (\Delta u^j(n+i))^2. \end{aligned} \quad (17)$$

Note that the difference between the two objective functions (16) and (9) will vanish as $\tilde{J}(n)$ approaches zero after optimization. This means that the output should have a perfect tracking of a reference trajectory by strong control actions, whenever necessary. Using the alternative objective function in (16), we can derive a controller by a hierarchical control design approach.

B. Hierarchical Control Design

By using the basic concept of decomposition-coordination in a large-scale system theory [7], the controller design can be accomplished through a two-layer iterative design process. The whole design is decomposed into the derivation of p local controllers. The subsystems regulated by those local controllers will be coordinated to derive a globally optimal control policy.

1) *Lower Layer Design*: All p subsystems need be considered in the lower layer. For the j th subsystem, the optimization problem is defined as follows:

$$\min_{\Delta u^j(n+1), \Delta u^j(n+2), \dots, \Delta u^j(n+N_u)} \tilde{J}^j(n) \quad (18)$$

subject to

$$\begin{aligned} R^j: & \text{ IF } y(n+k-1) \text{ is } A_0^j, y(n+k-2) \\ & \text{ is } A_1^j, \dots, y(n+k-m) \text{ is } A_{m-1}^j \quad \text{and} \\ & u(n+k-1) \text{ is } B_0^j, u(n+k-2) \\ & \text{ is } B_1^j, \dots, u(n+k-l) \text{ is } B_{l-1}^j \\ & \text{ THEN } \hat{y}^j(n+k) = y^j(n+k-1) \\ & + \sum_{i=1}^T h_i^j \Delta u(n+k-i) + \varepsilon^j(n+k-1) \end{aligned} \quad (19)$$

where $\varepsilon^j(n+k-1)$ serves for system coordination; it is determined at the upper layer. The information to be transmitted to the upper layer is included in the following set:

$$\begin{aligned} S_{L \rightarrow U}^j = & \{ \hat{y}^j(n+i), \Delta u^j(n+l) \mid i = 1, 2, \dots, N_y; \\ & l = 1, 2, \dots, N_u \}. \end{aligned} \quad (20)$$

2) *Upper Layer Design*: The upper layer coordination targets the identification of globally optimal control policies through coordinating $\varepsilon^j(n+k)$ for each of the p local subsystems. Thus, the objective function in this layer can be defined as

$$\min_{\varepsilon(n), \varepsilon(n+1), \dots, \varepsilon(n+N_y)} \tilde{J}(n). \quad (21)$$

Note that $\varepsilon(n)$ in (21) is a vector, where each element is the difference of $y(n)$ and $y^j(n)$. The minimization is accomplished by identifying error variable $\varepsilon^j(n+k-1)$, which forms the following set for each subsystem

$$S_{U \rightarrow L}^j = \{ \varepsilon^j(n+k-1) \mid k = 1, 2, \dots, N_y \}. \quad (22)$$

3) *System Coordination*: Fig. 1 shows a two-layer structure for the fuzzy model-based system coordination. From the lower layer, the local information of output and control in p sets of $S_{L \rightarrow U}^j$ is transmitted to the upper layer. At the upper layer, the error variables $\varepsilon^j(n+k-1)$, $k = 1, 2, \dots, N_y$; $j = 1, 2, \dots, p$ are evaluated as

$$\varepsilon^j(n+k-1) = y(n+k-1) - y^j(n+k-1). \quad (23)$$

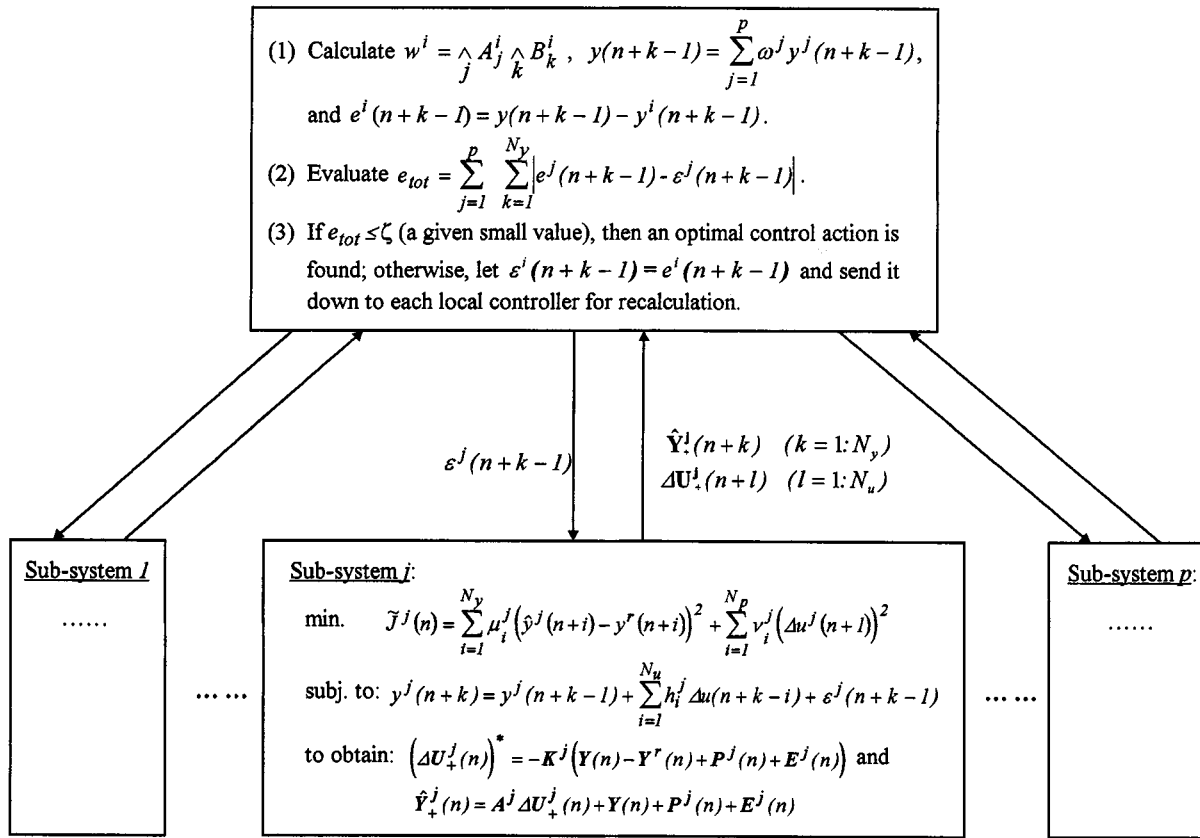


Fig. 1. Structure of a hierarchical controller design.

These values will be compared with those for the same error variables calculated in the last iteration, say $\varepsilon^j(n+k-1)$. If the smallest tolerable error is termed ζ and

$$\sum_{j=1}^p \sum_{k=1}^{N_y} |e^j(n+k-1) - \varepsilon^j(n+k-1)| > \zeta \quad (24)$$

then the control policies are not optimal and need be modified at the local layer. This can be accomplished in a new iterative process by sending down the set $S_{U \rightarrow L}^j$ for each subsystem. If the inequality in (24) does not hold, then the control policies are satisfactory, the predicted output values are reliable, and the coordination process is finished.

4) *Localized Controller Design:* At the lower layer, the task is to identify optimal local control policies and output estimations by all FIs. For clarity, the objective function defined in (17) can be rewritten in a matrix form as follows:

$$\tilde{J}^j(n) = (\hat{\mathbf{Y}}_+^j(n) - \mathbf{Y}^r(n))^T \mathbf{W}_1^j (\hat{\mathbf{Y}}_+^j(n) - \mathbf{Y}^r(n)) + (\Delta \mathbf{U}_+^j(n))^T \mathbf{W}_2^j (\Delta \mathbf{U}_+^j(n)) \quad (25)$$

where

$$\hat{\mathbf{Y}}_+^j(n) = (\hat{y}^j(n+1) \hat{y}^j(n+2) \dots \hat{y}^j(n+N_y))^T \quad (26)$$

$$\mathbf{Y}^r(n) = (y^r(n+1) y^r(n+2) \dots y^r(n+N_y))^T \quad (27)$$

$$\Delta \mathbf{U}_+^j(n) = (\Delta u^j(n+1) \Delta u^j(n+2) \dots \Delta u^j(n+N_u))^T \quad (28)$$

$$\mathbf{W}_1^j = \text{diag}\{\mu_1^j, \mu_2^j, \dots, \mu_{N_y}^j\} \quad (29)$$

$$\mathbf{W}_2^j = \text{diag}\{\nu_1^j, \nu_2^j, \dots, \nu_{N_u}^j\}. \quad (30)$$

The N_y -step prediction of the output by the j th FI can be derived from (19). These predictions can be expressed in a matrix form

$$\hat{\mathbf{Y}}_+^j(n) = \mathbf{A}^j \Delta \mathbf{U}_+^j(n) + \mathbf{Y}(n) + \mathbf{P}^j(n) + \mathbf{E}_+^j(n) \quad (31)$$

where

$$\mathbf{A}^j = \begin{pmatrix} a_1^j & 0 & 0 & \dots & 0 \\ a_2^j & a_1^j & 0 & & 0 \\ a_3^j & a_2^j & a_1^j & & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N_y}^j & a_{N_y-1}^j & a_{N_y-2}^j & \dots & a_{N_y-N_u+1}^j \end{pmatrix} \quad (32)$$

$$a_i^j = \sum_{k=1}^i h_k^j \quad (33)$$

$$\mathbf{Y}(n) = (y(n) y(n) \dots y(n))^T \quad (34)$$

$$\mathbf{P}^j(n) = (P_1^j(n) P_2^j(n) \dots P_{N_y}^j(n))^T \quad (35)$$

$$\mathbf{E}_+^j(n) = \left(0 \sum_{k=1}^2 \varepsilon^j(n+k) \dots \sum_{k=1}^{N_y} \varepsilon^j(n+k) \right)^T \quad (36)$$

$$P_i^j(n) = \sum_{k=1}^i \sum_{l=k+1}^T h_l^j \Delta u(n+l-k). \quad (37)$$

The resulting control policy for the j th subsystem can be derived as

$$\begin{aligned} \tilde{J}^j(n) = & (\Delta \mathbf{U}_+^j(n))^T \left(\mathbf{A}^{jT} \mathbf{W}_1^j \mathbf{A} + \mathbf{W}_2^j \right) \Delta \mathbf{U}_+^j(n) \\ & + (\Delta \mathbf{U}_+^j(n))^T \mathbf{A}^{jT} \mathbf{W}_1^j \mathbf{Z}^j(n) \\ & + (\mathbf{Z}^j(n))^T \mathbf{W}_1^j \mathbf{A}^j \Delta \mathbf{U}_+^j(n) \\ & + (\mathbf{Z}^j(n))^T \mathbf{W}_1^j \mathbf{Z}^j(n) \end{aligned} \quad (38)$$

where

$$\mathbf{Z}^j(n) = \mathbf{Y}(n) - \mathbf{Y}^r(n) + \mathbf{P}^j(n) + \mathbf{E}_+^j(n). \quad (39)$$

Minimizing (38) yields

$$\begin{aligned} \frac{\partial \tilde{J}^j(n)}{\partial \Delta \mathbf{U}_+^j(n)} = & 2 \left(\mathbf{A}^{jT} \mathbf{W}_1^j \mathbf{A}^j + \mathbf{W}_2^j \right) \Delta \mathbf{U}_+^j(n) \\ & + 2 \mathbf{A}^{jT} \mathbf{W}_1^j \mathbf{Z}^j(n) = 0. \end{aligned} \quad (40)$$

Then, the control law by the j th FI can be identified as

$$(\Delta \mathbf{U}_+^j(n))^* = -\mathbf{K}^j \mathbf{Z}^j(n) \quad (41)$$

where \mathbf{K}^j is the feedback gain matrix for the j th subsystem; it can be derived as

$$\mathbf{K}^j = \left(\mathbf{A}^{jT} \mathbf{W}_1^j \mathbf{A}^j + \mathbf{W}_2^j \right)^{-1} \mathbf{A}^{jT} \mathbf{W}_1^j. \quad (42)$$

5) *Global Control Policy*: As the optimal local control policies at the lower layer are identified through optimization, the optimal global control policies can be accordingly derived at the upper layer. That is

$$\Delta \mathbf{U}_+(n) = (\Delta u(n+1) \ \Delta u(n+2) \ \dots \ \Delta u(n+N_u))^T \quad (43)$$

where $\Delta u(n+i)$ is evaluated by (10).

C. Implementation Procedure for Fuzzy MPC

Based on the definition of the two-layer optimization problems and the computational mechanism of identifying optimal control policies, a procedure is introduced to implement the hierarchical control algorithm.

- Step 1) Set the error variables $\varepsilon^j(n+k-1), k = 1, 2, \dots, N_y; j = 1, 2, \dots, p$ to zero at the upper layer and send the sets $S_{U \rightarrow L}^j$ down to the corresponding subsystems at the lower layer. This initial setting comes from the consideration of zero bias between $y^j(n+k-1)$ and $y(n+k-1), k = 1, 2, \dots, N_y$.
- Step 2) Determine the N_u -step control policy $\Delta u^j(n+k-1), k = 1, 2, \dots, N_u$ and estimate the N_y steps of output $y^j(n+k-1), k = 1, 2, \dots, N_y$ for all p subsystems based on their FIs at the lower layer. In this determination process, $\varepsilon^j(n+k-1), k = 1, 2, \dots, N_y; j = 1, 2, \dots, p$ are fixed. The local optimal output and control values form the set $S_{L \rightarrow U}^j$, which is transmitted to the upper layer.
- Step 3) Calculate new $y(n+k-1), k = 1, 2, \dots, N_y$, based on (5) and (6) and, further, calculate the new errors

$\varepsilon^j(n+k-1), j = 1, 2, \dots, p$ between the currently estimated output and the one estimated in the last time, according to (23).

Step 4) Examine the total error e_{tot} of all p subsystems

$$e_{\text{tot}} = \sum_{j=1}^p \sum_{k=1}^{N_y} |\varepsilon^j(n+k-1) - \varepsilon^j(n+k-1)|. \quad (44)$$

Step 5) If $e_{\text{tot}} > \zeta$ (a prespecified error tolerance), then let

$$\begin{aligned} \varepsilon^j(n+k-1) = & \varepsilon^j(n+k-1), \\ k = & 1, 2, \dots, N_y; \quad j = 1, 2, \dots, p \end{aligned} \quad (45)$$

and establish the new sets $S_{U \rightarrow L}^j, j = 1, 2, \dots, p$. These sets should be sent down to all subsystems. Then go to Step 2.

Step 6) If $e_{\text{tot}} \leq \zeta$, then an optimal control policy $\Delta \mathbf{U}_+(n)$ and system output $\hat{\mathbf{Y}}_+(n)$ can be evaluated as

$$\begin{aligned} \Delta \mathbf{U}_+(n) = & \left(\sum_{j=1}^p \omega^j \Delta u^j(n+1) \ \sum_{j=1}^p \omega^j \Delta u^j(n+2) \ \dots \right. \\ & \left. \sum_{j=1}^p \omega^j \Delta u^j(n+N_u) \right)^T \end{aligned} \quad (46)$$

$$\begin{aligned} \hat{\mathbf{Y}}_+(n) = & \left(\sum_{j=1}^p \omega^j y^j(n+1) \ \sum_{j=1}^p \omega^j y^j(n+2) \ \dots \right. \\ & \left. \sum_{j=1}^p \omega^j y^j(n+N_y) \right)^T. \end{aligned} \quad (47)$$

Note that the first step of the derived optimal control policy, i.e., $\sum_{j=1}^p \omega^j \Delta u^j(n+1)$, is the output of the controller. All other steps of the control policy are used to predict future outputs. Also note that the superscript T is a transpose operator, not the model horizon.

IV. PARAMETER TUNING

In controller design, the difficulty encountered is how to quickly minimize the upper bound of the objective function so that the control actions can force the process to track a specified trajectory as close as possible. Like the design of a regular MPC, the parameters to be tuned in the FMPC include model horizon, control horizon, prediction horizon, and weighting factors \mathbf{W}_1^j and $\mathbf{W}_2^j, j = 1, 2, \dots, p$.

So far, there has been no rigorous solution to the selection of optimal model horizon (T), control horizon (V), and prediction horizon (U) for MPC design. In this work, a number of rules of thumb are used to select three horizons [13]. In this work, T is selected so that $T\Delta t \geq$ open-loop settling time, which is equal to the time for the open-loop step response to be 99% complete. Note that increasing V results in a more conservative control action that has a stabilizing effect but also increases the computational effort. On the other hand, U is the number of future control actions that are calculated in the optimization step to reduce the predicted errors. The computational effort increases as U is

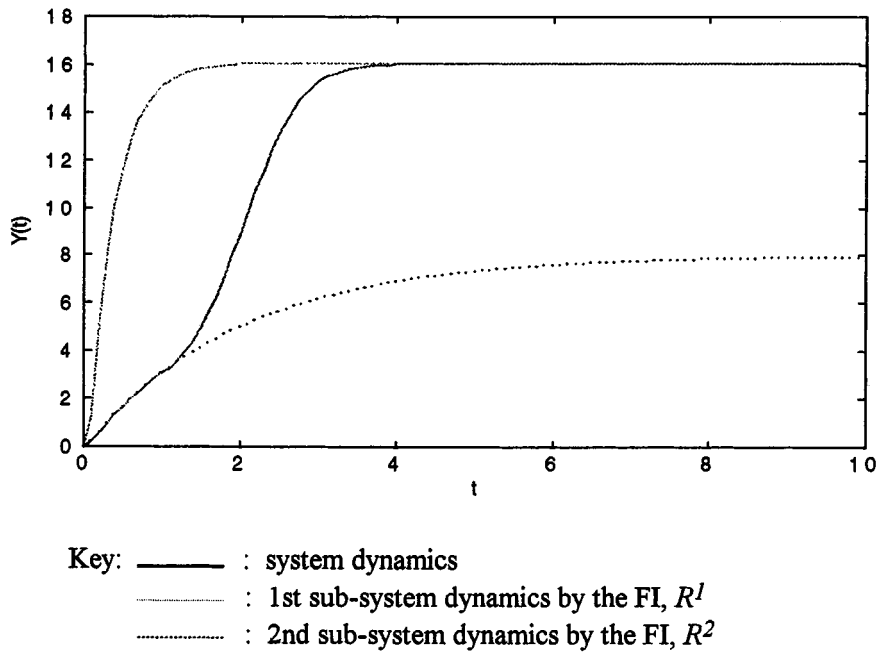


Fig. 2. Open-loop dynamic response of a nonlinear system to a unit set change (Example 1).

increased. A smaller value of U leads to a robust controller that is relatively insensitive to model errors. Tradeoff must be taken in selecting V and U based on dynamic responses and computational errors. Computational time is actually not a problem. This fuzzy logic-based MPC design avoids considerably computational burden caused by traditionally used highly nonlinear models and nonlinear optimization.

The ranges of weighting factors \mathbf{W}_1^j and \mathbf{W}_2^j can be very wide. There have been no systematic and rigorous approaches available for optimally determining these factors. In this work, a heuristic approach is proposed. The basic idea of the approach is delineated below.

For the system being decomposed into p subsystems, there are $2p$ weighting factors to be determined. For any subsystem j , the importance is not the magnitudes of \mathbf{W}_1^j and \mathbf{W}_2^j , but their relative magnitudes. Thus, to simplify their selection, we can set all $\mathbf{W}_1^j, j = 1, 2, \dots, p$, to the same, say \mathbf{W}_1 .

The remaining p weighting factors (\mathbf{W}_2^j) should be determined independently through optimizing each subsystem. These factors need be retuned when a global system optimization is considered. This can be a time-consuming task because there is no systematic approach to follow. There is no guarantee that a solution with the total tolerable errors less than ζ is globally optimal by this approach [referred to (24)]. A systematic three-step procedure is proposed for turning weighting factors. The change of \mathbf{W}_2 value gradually will help identify better solutions, but still not guarantee the global optimality. Alternatively, the tuning can be accomplished by initially setting all \mathbf{W}_2^j to the same, say W_2 which is equal to $\max\{\mathbf{W}_2^j | i = 1, 2, \dots, p\}$. Then \mathbf{W}_2 is gradually reduced during global optimization. Therefore, we propose the following three-step procedure to tune the weighting factors.

Step 1) Select a value for \mathbf{W}_1 and assign it to all p local controllers. Then, determine \mathbf{W}_2^j independently for

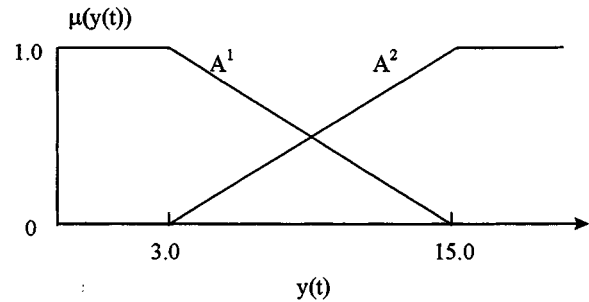


Fig. 3. Definition of fuzzy sets A^1 and A^2 for FIs R^1 and R^2 , respectively (Example 1).

each local controller in order to minimize the objective function for that subsystem.

- Step 2) Identify the largest \mathbf{W}_2^j that is denoted as \mathbf{W}_2 . Then, assign \mathbf{W}_2 to all p subsystems.
- Step 3) Examine the system's closed-loop dynamic performance. If not satisfied, then reduce the value of \mathbf{W}_2 gradually until the most desirable dynamic performance is identified.

V. CASE STUDIES

Three highly nonlinear systems are selected for studying the proposed design approach. The first system is modeled by two FIs. The second system's input-output data contains various noises. A fuzzy convolution model consisting of three FIs is developed. The third example is about the control of a continuous-stirred tank reactor (CSTR), which was studied by [9]. Fuzzy model predictive controllers are designed to realize closed-loop control for all these systems.

Example 1:

a) Process modeling: The process demonstrates nonlinear behavior in responding to a unit step change (Fig. 2, solid

TABLE I
MODEL COEFFICIENTS (h_i^j) FOR EXAMPLE 1

No.	h_i^1	h_i^2
1	0.2000	0.2000
2	0.1460	1.7320
3	0.1398	1.1551
4	0.1329	0.8663
5	0.1264	0.6411
6	0.1202	0.4751
7	0.1143	0.3520
8	0.1087	0.2608
9	0.1034	0.1933
10	0.0983	0.1432
11	0.0935	0.1061
12	0.0889	0.0786
13	0.0846	0.0583
14	0.0804	0.0432
15	0.0765	0.0320
16	0.0728	0.0237
17	0.0692	0.0176
18	0.0658	0.0130
19	0.0626	0.0096
20	0.0595	0.0071
21	0.0566	0.0053
22	0.0538	0.0039
23	0.0512	0.0029
24	0.0487	0.0022
25	0.0463	0.0016
26	0.0440	0.0012
27	0.0419	0.0009
28	0.0398	0.0006
29	0.0379	0.0005
30	0.0360	0.0004
31	0.0343	0.0003
32	0.0326	0.0002
33	0.0310	0.0001
34	0.0295	0.0001
35	0.0280	0.0001

No.	h_i^1	h_i^2
36	0.0267	0.0001
37	0.0254	0.0000
38	0.0241	0.0000
39	0.0229	0.0000
40	0.0218	0.0000
41	0.0207	0.0000
42	0.0197	0.0000
43	0.0188	0.0000
44	0.0178	0.0000
45	0.0170	0.0000
46	0.0161	0.0000
47	0.0153	0.0000
48	0.0146	0.0000
49	0.0139	0.0000
50	0.0132	0.0000
51	0.0126	0.0000
52	0.0119	0.0000
53	0.0114	0.0000
54	0.0108	0.0000
55	0.0103	0.0000
56	0.0098	0.0000
57	0.0093	0.0000
58	0.0088	0.0000
59	0.0084	0.0000
60	0.0080	0.0000
61	0.0076	0.0000
62	0.0072	0.0000
63	0.0069	0.0000
64	0.0065	0.0000
65	0.0062	0.0000
66	0.0059	0.0000
67	0.0056	0.0000
68	0.0053	0.0000
69	0.0051	0.0000
70	0.0048	0.0000

line). The system responds nearly exponentially, although quite slowly, during the first 1.1 min. The output is then increased quickly until $t = 2.8$ min, where the response becomes sluggish. For this system, a simple fuzzy convolution model consisting of two FIs (R^1 and R^2) is developed as follows:

R^1 : IF $y(n)$ is A^1

$$\text{THEN } y^1(n+1) = y(n) + \sum_{i=1}^{70} h_i^1 u(n+1-i) \quad (48)$$

R^2 : IF $y(n)$ is A^2

$$\text{THEN } y^2(n+1) = y(n) + \sum_{i=1}^{70} h_i^2 u(n+1-i). \quad (49)$$

The fuzzy sets A^1 and A^2 in the FIs are defined in Fig. 3. The coefficients $h_i^j, i = 1, 2, \dots, 70, j = 1, 2$ of the two FIs are listed in Table I. The model horizon is set to 70. Note that when $y^1(n+1)$, for instance, is evaluated by $R^1, y(n)$ rather

than $y^1(n)$ is used. This is more desirable since $y(n)$ reflects an error correction based on both $y^1(n)$ and $y^2(n)$ in the n th step.

b) *Controller design*: In designing the FMPC controller, predictive horizon and control horizon are set to three and two, respectively. Weighting factors are selected as follows:

$$W_1^j = \text{diag}\{20, 20, 10\} \quad (50)$$

$$W_2^j = \text{diag}\{25, 25\}. \quad (51)$$

The two local controllers are synthesized and the feedback gain matrices are

$$K^1 = \begin{bmatrix} 0.1324 & 0.2141 & 0.1478 \\ -0.0150 & 0.1239 & 0.1114 \end{bmatrix} \quad (52)$$

$$K^2 = \begin{bmatrix} 0.0323 & 0.2775 & 0.0828 \\ -0.0345 & -0.2328 & 0.2177 \end{bmatrix}. \quad (53)$$

c) *Simulation*: System simulation is conducted to study how the change of the weighting factors and the selection of reference trajectories affect the system's dynamic performance

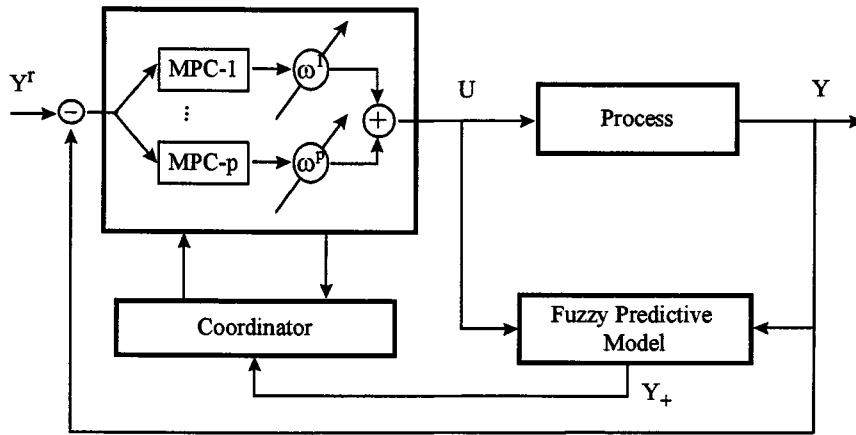


Fig. 4. Structure of the FMPC controller.

and to compare the dynamic responses by the FMPC and a conventional MPC.

It is known that in designing a model predictive controller, the reference trajectory $\mathbf{Y}^r(n)$ should be adjustable according to the control requirement [14]. In this case, a first-order response is selected as the reference trajectory which is described as

$$y^r(t) = y^{sp} \left(1 - e^{-\frac{t}{\tau}} \right) \quad (54)$$

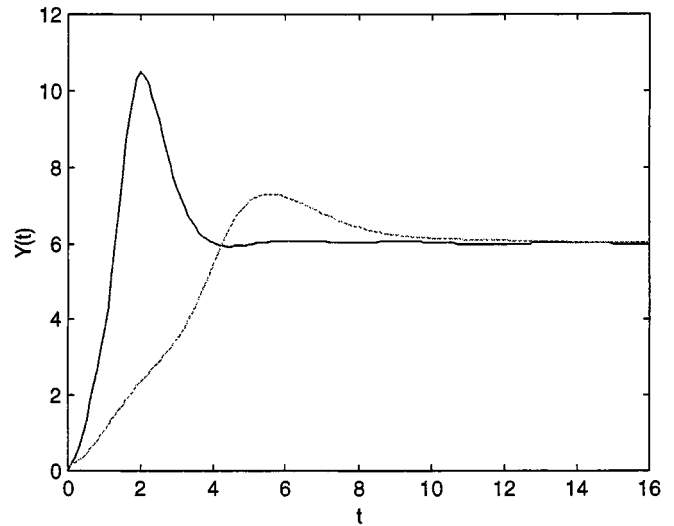
where y^{sp} is the magnitude of a step change; τ is the time constant which is the only adjustable parameter. In this case, y^{sp} is equal to one, and τ is set to 2.5 (Case I) and 0.5 (Case II). Fig. 4 gives the structure of the FMPC controller. Note that all local controllers will be used all the time. This means that there is no switch from one local controller to the other in operation. As shown in (5), the system output $y(n+1)$ is inferred as a weighted average value of the outputs of all subsystems. On the other hand, the overall control policy to the process under control is the weighted sum of all local control policies, as shown in (10). This kind of design not only eliminates the controller switch problem and thus possible system instability, but also provides a much more smooth control performance in process operation. Fig. 5 provides the closed-loop dynamic response of the system under different values of τ , which shows how a speedy response can be adjusted.

Fig. 6 gives the comparison of the closed-loop dynamic performance of the system when parameter matrix \mathbf{W}_2 is set differently. In this case, τ is maintained at 2.5. The dynamic response with the smallest norm of \mathbf{W}_2 (Case III) is the most desirable. If the norm is reduced further, the response will become worse due to the appearance of oscillation (not plotted in the figure).

To demonstrate the superiority of the FMPC design methodology, we also conducted a series of simulations by using an optimally designed conventional MPC. Fig. 7 illustrates the different control qualities of the system when FMPC and MPC are both optimally designed (with different \mathbf{W}_2). In this example, the FMPC demonstrates a much better control performance.

Example 2:

a) *Process modeling*: The open-loop dynamic response of the system is shown in Fig. 8, where large noise exists. A



Key: —: Case I ($\tau = 2.5$); - - - -: Case II ($\tau = 0.5$)

Fig. 5. Closed-loop dynamic responses with the same parameter design [$\mathbf{W}_1 = \text{diag}\{20, 20, 10\}$ and $\mathbf{W}_2 = \text{diag}\{25, 25\}$].

fuzzy dynamic model containing three FIs is developed as follows:

$$R^1: \text{IF } y(n) \text{ is } B^1 \\ \text{THEN } y^1(n+1) = y(n) + \sum_{i=1}^{80} h_i^1 u(n+1-i) \quad (55)$$

$$R^2: \text{IF } y(n) \text{ is } B^2 \\ \text{THEN } y^2(n+1) = y(n) + \sum_{i=1}^{80} h_i^2 u(n+1-i) \quad (56)$$

$$R^3: \text{IF } y(n) \text{ is } B^3 \\ \text{THEN } y^3(n+1) = y(n) + \sum_{i=1}^{80} h_i^3 u(n+1-i). \quad (57)$$

Fuzzy sets $B^i, i = 1, 2, 3$ are defined in Fig. 9. The model horizon is set to 80. The coefficients $h_i^j, i = 1, 2, \dots, 80, j =$

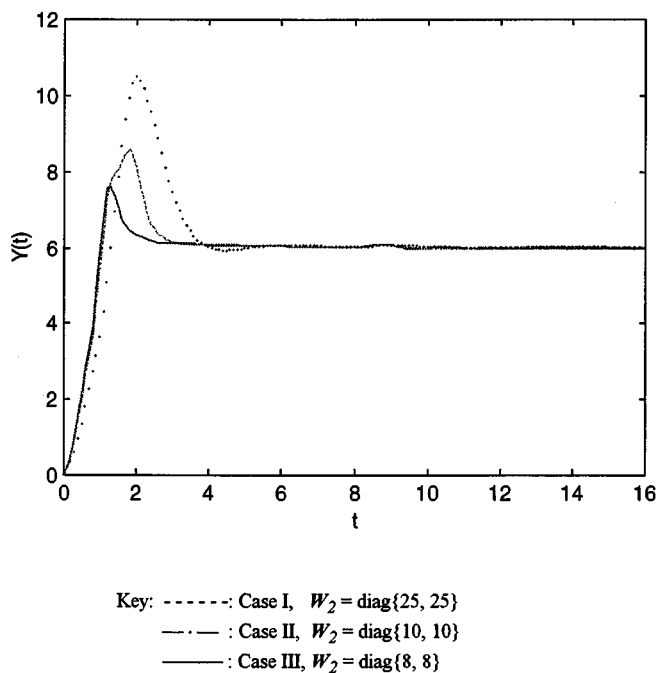


Fig. 6. Closed-loop dynamic responses of the system using the same $W_1 (= \text{diag}\{20, 20, 10\})$ and the different W_2 when $\tau = 2.5$.

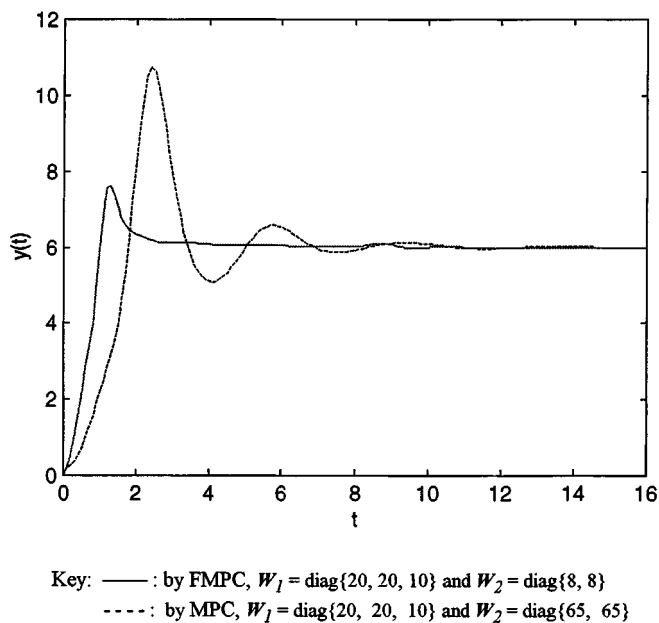


Fig. 7. Comparison of the closed-loop dynamic responses by the FMPC and an MPC when $\tau = 2.5$.

1, 2, 3 of the three FIs are derived in Table II. The system dynamics derived by the model is depicted in Fig. 8 (see the smooth curve).

b) Controller design: In designing an FMPC controller, predictive horizon and control horizons are set to three and two, respectively. Weighting factors are selected below:

$$W_1^j = \text{diag}\{20, 20, 10\} \quad (58)$$

$$W_2^j = \text{diag}\{25, 25\}. \quad (59)$$

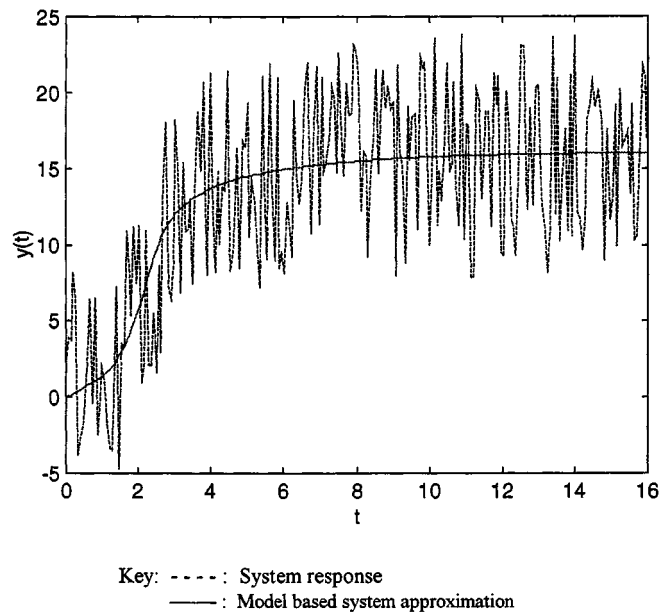


Fig. 8. Open-loop dynamic response of a nonlinear system to a unit set change (Example 2).

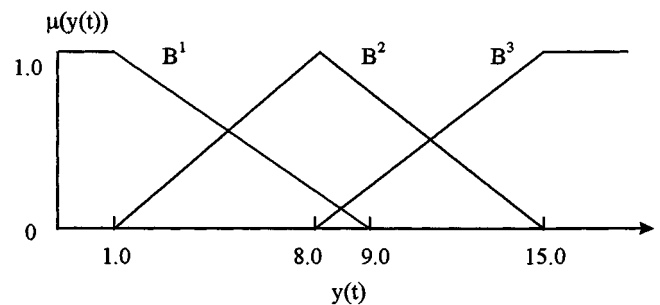


Fig. 9. Definition of fuzzy sets B^1 through B^3 for FIs R^1 through R^3 , respectively (Example 2).

The three local controllers are synthesized. Their feedback gain matrices are obtained as follows:

$$K^1 = \begin{bmatrix} 0.1016 & 0.1514 & 0.1021 \\ -0.0071 & 0.0982 & 0.0777 \end{bmatrix} \quad (60)$$

$$K^2 = \begin{bmatrix} 0.0307 & 0.2447 & 0.1025 \\ -0.0257 & -0.1454 & 0.1748 \end{bmatrix} \quad (61)$$

$$K^3 = \begin{bmatrix} 0.0773 & 0.2205 & 0.1625 \\ -0.0161 & -0.0533 & 0.1179 \end{bmatrix}. \quad (62)$$

c) Simulation: Fig. 10 depicts the dynamic responses in different values of W_2 when τ for the reference trajectory is set to 0.88. When W_2 is equal to $\text{diag}\{25, 25\}$, the dynamic response is the most desirable (Case I). As a comparison, the dynamic response under a smaller norm of W_2 (Case II) is plotted in the same figure. In this case, the closed-loop response becomes unstable. It should be pointed out that the unstable stage does not happen in other two examples. The number of trials for a satisfactory W_2 cannot be determined prior to looking at system dynamics.

The designed controller can also properly control the system when different set point changes occur. Fig. 11 demonstrates the closed-loop dynamic responses using four different set point

TABLE II
MODEL COEFFICIENTS (h_i^j) FOR EXAMPLE 2

No.	h_i^1	h_i^2	h_i^3
1	0.2000	0.2000	0.2000
2	0.1120	1.5620	0.4120
3	0.1115	1.0469	0.3577
4	0.1093	0.8159	0.3392
5	0.1072	0.6223	0.3177
6	0.1051	0.4760	0.2980
7	0.1030	0.3639	0.2795
8	0.1010	0.2783	0.2622
9	0.0990	0.2128	0.2459
10	0.0971	0.1627	0.2306
11	0.0952	0.1244	0.2163
12	0.0933	0.0951	0.2029
13	0.0915	0.0727	0.1903
14	0.0897	0.0556	0.1785
15	0.0879	0.0425	0.1674
16	0.0862	0.0325	0.1570
17	0.0845	0.0249	0.1473
18	0.0828	0.0190	0.1381
19	0.0812	0.0145	0.1296
20	0.0796	0.0111	0.1215
21	0.0781	0.0085	0.1140
22	0.0765	0.0065	0.1069
23	0.0750	0.0050	0.1003
24	0.0736	0.0038	0.0941
25	0.0721	0.0029	0.0882
26	0.0707	0.0022	0.0827
27	0.0693	0.0017	0.0776
28	0.0680	0.0013	0.0728
29	0.0666	0.0010	0.0683
30	0.0653	0.0008	0.0640
31	0.0640	0.0006	0.0601
32	0.0628	0.0004	0.0563
33	0.0616	0.0003	0.0528
34	0.0604	0.0003	0.0496
35	0.0592	0.0002	0.0465
36	0.0580	0.0002	0.0436
37	0.0569	0.0001	0.0409
38	0.0558	0.0001	0.0384
39	0.0547	0.0001	0.0360
40	0.0536	0.0001	0.0337

No.	h_i^1	h_i^2	h_i^3
41	0.0525	0.0000	0.0316
42	0.0515	0.0000	0.0297
43	0.0505	0.0000	0.0278
44	0.0495	0.0000	0.0261
45	0.0485	0.0000	0.0245
46	0.0476	0.0000	0.0230
47	0.0467	0.0000	0.0215
48	0.0457	0.0000	0.0202
49	0.0449	0.0000	0.0190
50	0.0440	0.0000	0.0178
51	0.0431	0.0000	0.0167
52	0.0423	0.0000	0.0156
53	0.0414	0.0000	0.0147
54	0.0406	0.0000	0.0138
55	0.0398	0.0000	0.0129
56	0.0390	0.0000	0.0121
57	0.0383	0.0000	0.0114
58	0.0375	0.0000	0.0106
59	0.0368	0.0000	0.0100
60	0.0361	0.0000	0.0094
61	0.0354	0.0000	0.0088
62	0.0347	0.0000	0.0082
63	0.0340	0.0000	0.0077
64	0.0333	0.0000	0.0073
65	0.0327	0.0000	0.0068
66	0.0320	0.0000	0.0064
67	0.0314	0.0000	0.0060
68	0.0308	0.0000	0.0056
69	0.0302	0.0000	0.0053
70	0.0296	0.0000	0.0049
71	0.0290	0.0000	0.0046
72	0.0284	0.0000	0.0043
73	0.0279	0.0000	0.0041
74	0.0273	0.0000	0.0038
75	0.0268	0.0000	0.0036
76	0.0263	0.0000	0.0034
77	0.0258	0.0000	0.0032
78	0.0253	0.0000	0.0030
79	0.0248	0.0000	0.0028
80	0.0243	0.0000	0.0026

changes ($y^{SP} = 3.5, 6, 10,$ and 12), all yielding satisfactory performance.

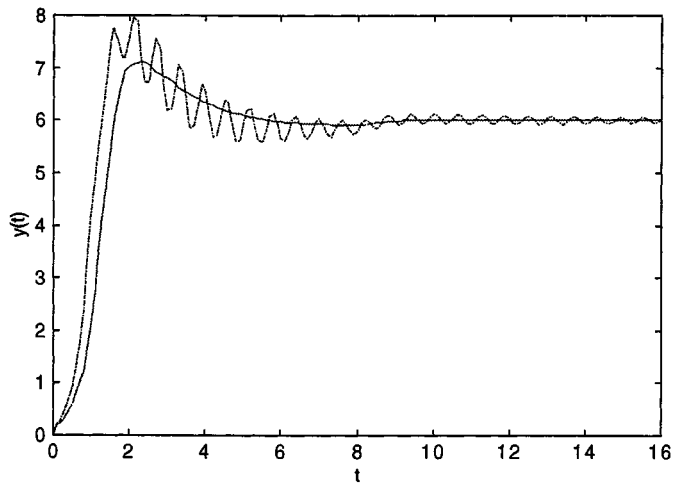
1) *Example 3—Control of a continuous-stirred tank reactor (CSTR):* [9] studied the control of a highly nonlinear CSTR process, which is very common in chemical and petrochemical plants. The control problem is selected here for testing the FMPC approach. In the process, an irreversible, exothermic reaction $A \rightarrow B$ occurs in a constant volume reactor that is cooled by a single coolant stream. The process is modeled by the following equations [9]:

$$\frac{dC_A(t+d)}{dt} = \frac{q(t)}{V}(C_{A0}(t) - C_A(t+d)) - k_0 C_A(t+d) \times \exp\left(\frac{-E}{RT(t)}\right) \quad (63)$$

$$\begin{aligned} \frac{dT(t)}{dt} = & \frac{q(t)}{V}(T_0(t) - T(t)) - \frac{(-\Delta H)k_0 C_A(t+d)}{\rho C_p} \\ & \times \exp\left(\frac{-E}{RT(t)}\right) \\ & + \frac{\rho_c C_{pc}}{\rho C_p V} q_c(t) \left\{ 1 - \exp\left(\frac{-hA}{q_c(t)\rho C_p}\right) \right\} \\ & \times (T_{c0}(t) - T(t)). \end{aligned} \quad (64)$$

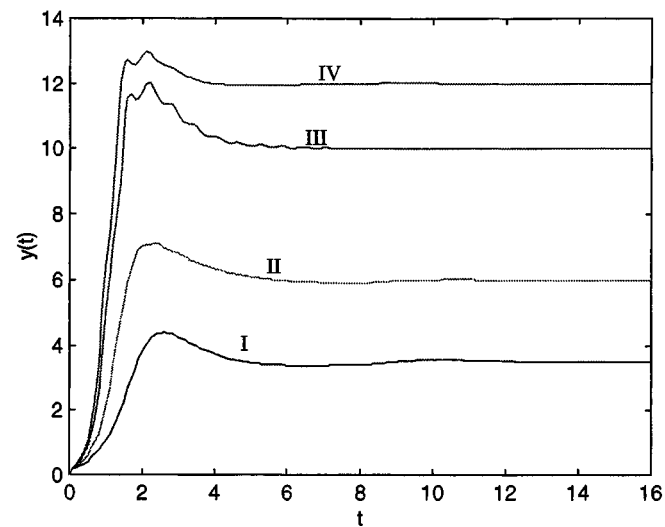
The objective of the design is to control the measured concentration of A , $C_A(t)$ by manipulating coolant flow rate $q_c(t)$. The nominal parameter values of the process appear in Table III.

b) *Fuzzy modeling:* In our study, the above rigorous model is used to generate a series of input–output time-series



Key: — : Case I, $W_2 = \text{diag}\{25, 25\}$
 - - - : Case II, $W_2 = \text{diag}\{15, 15\}$

Fig. 10. Closed-loop dynamic responses of the system using the same $W_1 (= \text{diag}\{20, 20, 10\})$ and the different W_2 when $\tau = 0.88$.



Key: Case I: $y^{SP} = 3.5$; Case II: $y^{SP} = 6.0$
 Case III: $y^{SP} = 10.0$; Case IV: $y^{SP} = 12.0$

Fig. 11. Closed-loop dynamic responses under different set point change (y^{SP}) when $W_1 = \text{diag}\{20, 20, 10\}$ and $W_2 = \text{diag}\{25, 25\}$ and $\tau = 0.88$.

data. The sampling time of the process measurements is set to 0.083 min (5 s). The data is then used to develop a fuzzy convolution model as follows:

$$\begin{aligned}
 R^1: & \text{ IF } q_c(n) \text{ is } Q^1 \\
 & \text{ THEN } C_A^1(n+1) \\
 & = C_A(n-1) + \sum_{i=1}^{100} h_i^1 q_c(n-i+1) \quad (65)
 \end{aligned}$$

TABLE III
 SPECIFICATION OF THE CSTR—EXAMPLE 3 [9]

Process variable	Normal operating condition
Measured product concentration (C_A)	0.1 mol/l
Reactor temperature (T)	438.54 K
Coolant flow rate (q_c)	103.41 l/min.
Process flow rate (q)	100.0 l/min.
Feed concentration (C_{A0})	1 mol/l
Feed temperature (T_0)	350.0 K
Inlet coolant temperature (T_{c0})	350.0 K
CSTR volume (V)	100 l
Heat transfer term (hA)	7×10^5 cal/(min·K)
Reaction rate constant (k_0)	7.2×10^{10} min ⁻¹
Activation energy term (E/R)	1×10^4 K
Heat of reaction ($-\Delta H$)	-2×10^5 cal/mol
Liquid density (ρ, ρ_c)	1×10^3 g/l
Specific heats (C_p, C_{pc})	1 cal/(g·K)

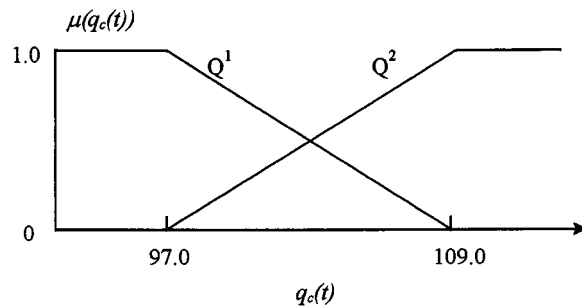


Fig. 12. Definition of fuzzy sets Q^1 and Q^2 for FIs R^1 and R^2 , respectively (Example 3).

$$\begin{aligned}
 R^2: & \text{ IF } q_c(n) \text{ is } Q^2 \\
 & \text{ THEN } C_A^2(n+1) \\
 & = C_A(n-1) + \sum_{i=1}^{100} h_i^2 q_c(n-i+1). \quad (66)
 \end{aligned}$$

The fuzzy model is structurally very simple, which requires only two FIs. The fuzzy sets Q^1 and Q^2 in the model are defined in Fig. 12. The open-loop response with various step changes in the coolant flow rate shows that the fuzzy convolution model can nearly perfectly describe the process dynamic behavior (Fig. 13). It also indicates that the process is indeed highly nonlinear. For each FI, the model horizon based on the impulse responses is set to 100 to ensure the completion of the dynamic response.

c) *Controller Design:* According to the FMPC controller design approach, each FI defines a subsystem. A localized controller need be designed for each subsystem. In design, the predictive horizon and control horizon are set to eight and five, respectively. The weighting factors in W_1^j are all 20 000 on the diagonal, and those in W_2^j are all 6.658 also on the diagonal.

The feedback gain matrices of the two local controller are listed in (67) and (68), shown at the bottom of the next page.

d) *Simulation:* A series of simulations are conducted to examine the control quality by the FMPC controller. In testing the set point tracking capability, the set point of C_A was changed from the nominal operating point 0.1 mol/l to 0.135, to 0.12, to 0.105, to 0.75, and then to 0.09 (see the dash line in Fig. 14). The

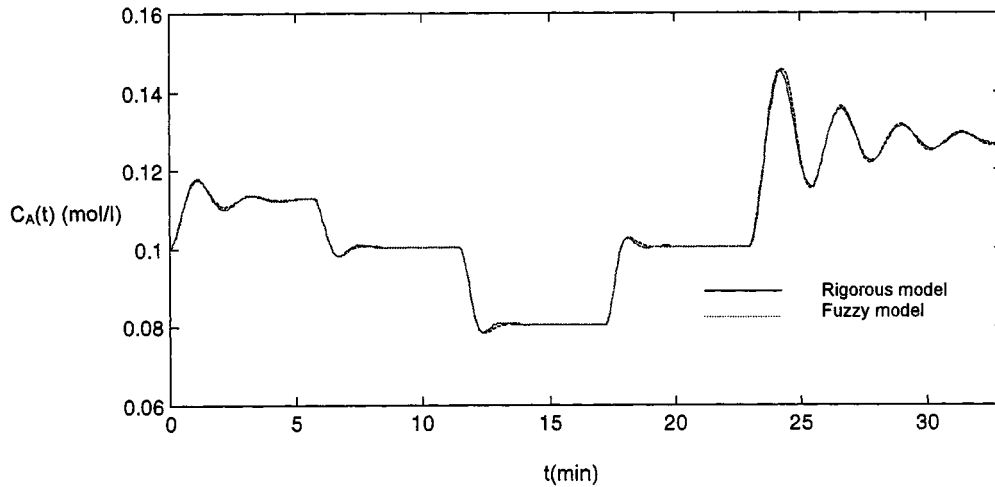


Fig. 13. Open-loop composition response of the CSTR process (Example 3).

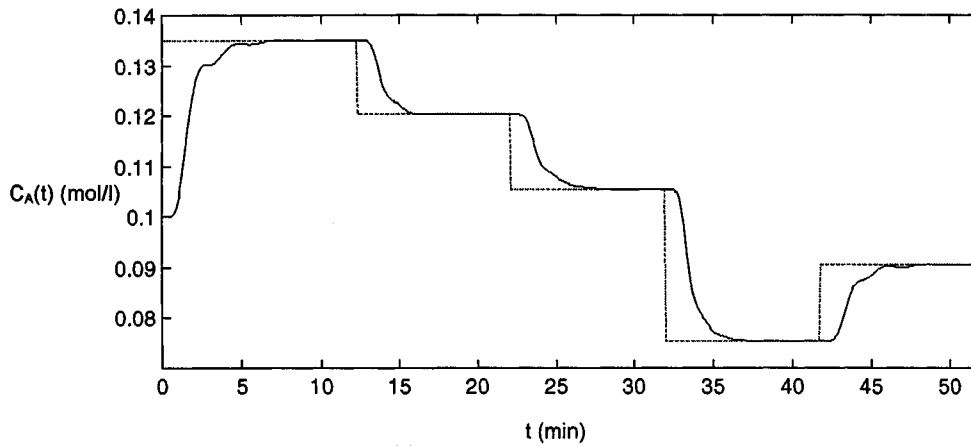


Fig. 14. Concentration set-point tracking of the CSTR process.

dynamic response of the system is depicted in the same figure. Apparently, the control dynamics is as good as [9].

Fig. 15 illustrates the disturbance rejection performance of the FMPC controller. In simulation, the disturbances of the feed concentration (C_{A0}) and the coolant temperature (T_{c0}) are added to the system. The feed concentration changes from 0.1 mol/l to 0.095 at 1 min, and back to 0.1 at 7 min. The coolant temperature is decreased by 10 °C at 18.5 min and gets back

to the nominal value at 28 min. The dynamic response in the figure shows that the FMPC system has a strong disturbance rejection capability.

VI. CONCLUDING REMARKS

A highly nonlinear system can be modeled by Takagi–Sugeno's fuzzy modeling methodology. If the model is de-

$$\mathbf{K}^1 = \begin{bmatrix} 0.604 & 1.167 & 1.775 & 2.443 & 3.155 & 3.891 & 4.615 & 5.294 \\ -0.042 & 0.538 & 1.086 & 1.689 & 2.362 & 3.097 & 3.863 & 4.626 \\ -0.033 & -0.092 & 0.481 & 1.029 & 1.641 & 2.339 & 3.106 & 3.910 \\ -0.024 & -0.067 & -0.129 & 0.448 & 1.008 & 1.646 & 2.375 & 3.178 \\ -0.016 & -0.046 & -0.088 & -0.143 & 0.448 & 1.033 & 1.699 & 2.460 \end{bmatrix} \quad (67)$$

$$\mathbf{K}^2 = \begin{bmatrix} 10.452 & 17.089 & 20.989 & 22.499 & 21.853 & 20.164 & 17.548 & 14.362 \\ -3.815 & 4.279 & 9.667 & 13.341 & 15.592 & 17.372 & 18.427 & 18.738 \\ -2.885 & -8.423 & -1.128 & 4.340 & 8.944 & 13.775 & 18.359 & 22.326 \\ -2.033 & -6.059 & -11.987 & -4.338 & 2.223 & 9.483 & 17.107 & 24.527 \\ -1.311 & -3.996 & -8.076 & -13.435 & -4.584 & 4.589 & 14.526 & 24.814 \end{bmatrix} \quad (68)$$

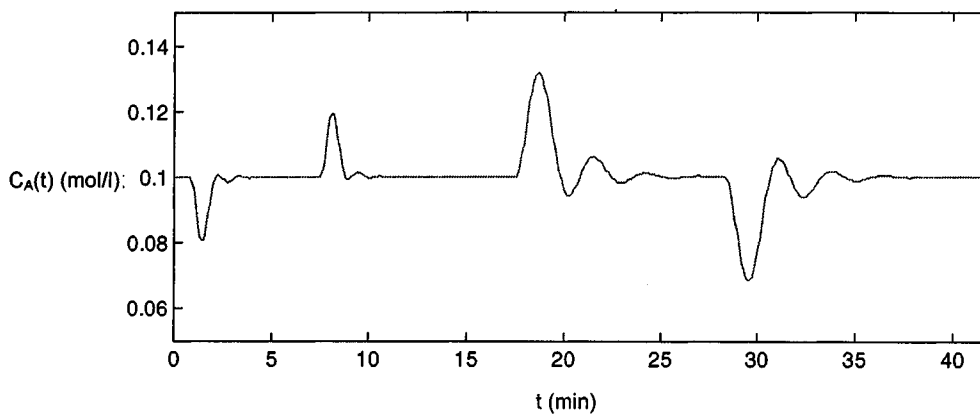


Fig. 15. Concentration disturbance rejection of the CSTR process.

veloped using impulse signal, the resultant one is a fuzzy convolution model. With this type of model, a novel FMPC methodology is developed in this paper. By this methodology, a controller is designed through a hierarchical control design, which can readily identify a near optimal system structure and parameters. The approach effectively avoids extensive optimization steps usually encountered in designing a NMPC, while the computational time is nearly negligible. This greatly advances the feasibility of on-line applications.

REFERENCES

- [1] G. R. Arulalan and P. B. Deshpande, "Simplified model predictive control," *Ind. Eng. Chem. Res.*, vol. 26, pp. 356–362, 1987.
- [2] C. R. Cutler and B. L. Ramaker, "Dynamic matrix control—A computer control algorithm," presented at the AIChE Spring Nat. Meet., Houston, TX, 1979.
- [3] J. W. Eaton and J. B. Rawlings, "Model-predictive control of chemical processes," *Chem. Eng. Sci.*, vol. 47, pp. 705–720, 1992.
- [4] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice—A survey," *Automatica*, vol. 25, pp. 335–348, 1989.
- [5] Z. P. Liu and Y. L. Huang, "Fuzzy model-based optimal dispatching for NO_x reduction in power plants," *Int. J. Elect. Power Energy Syst.*, vol. 20, no. 3, pp. 169–176, 1998.
- [6] H. H. Lou and Y. L. Huang, "Fuzzy logic based process modeling using limited experimental data," *Int. J. Eng. Applicat. Artificial Intell.*, vol. 13, no. 2, pp. 121–135, 2000.
- [7] M. Jamshidi, *Large-Scale Systems: Modeling and Control*. New York: Elsevier Science, 1983, vol. 9, ser..
- [8] S. W. Kim, E. T. Kim, and M. Park, "A new adaptive fuzzy controller using the parallel structure of fuzzy controller and its application," *Fuzzy Sets Syst.*, vol. 81, pp. 205–226, 1996.
- [9] J. D. Morningred, B. E. Paden, D. E. Seborg, and D. A. Mellichamp, "An adaptive nonlinear predictive controller," *Chem. Eng. Sci.*, vol. 47, pp. 755–762, 1992.
- [10] Y. Nakamori, K. Suzuki, and T. Yamanaka, "Model predictive control using fuzzy dynamic models," in *Proc. IFSA'91 Brussels*, vol. 135, Brussels, Belgium, July 1991, p. 138. vol. Engrg..
- [11] J. A. Richalet, A. Rault, J. D. Testud, and J. Papon, "Model predictive heuristic control: Applications to industrial processes," *Automatica*, vol. 14, pp. 413–420, 1978.
- [12] N. L. Ricker, "Model-predictive control: State of the art," in *Proc. Chemical Process Control IV*, 1991, pp. 271–296.
- [13] D. Seborg, T. F. Edgar, and D. A. Mellichamp, *Process Dynamics and Control*. New York: Wiley, 1989, ch. 27.
- [14] R. Soeterboek, *Predictive Control: A Unified Approach*. Englewood Cliffs, NJ: Prentice-Hall, 1992, ch. 3.
- [15] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, pp. 116–132, Jan./Feb. 1985.



Y. L. Huang received the B.S. degree from Zhejiang University, China, in 1982, and the M.S. and Ph.D. degrees from Kansas State University, Manhattan, KS, in 1988 and 1990, respectively, all in chemical engineering.

He joined Wayne State University, Detroit, MI, as an Assistant Professor in 1993, after one year of postdoctoral research at the University of Texas at Austin. He was promoted to an Associate Professor in 1997. He has published widely in all the areas listed below and the developed theories,

methodologies, algorithms, and computer-aided tools have been widely applied in the chemical, petrochemical, electroplating, and automotive coating industries for process improvement, energy recovery, chemical and water reduction, quality control, and process pollution prevention. His research has been mainly focused on process synthesis, modeling, control, and optimization using large-scale system theories, artificial intelligence, fuzzy logic, and neural networks and has been supported by NSF, EPA, NASA, DOE, ACS, AESF, and various industries.



Helen H. Lou received the B.S. degree in chemical engineering from Zhejiang University, China, and the M.S. degree in chemical engineering from Wayne State University, Detroit, MI, in 1993 and 1998, respectively. She is currently working toward the Ph.D. degree in the Department of Chemical Engineering and Materials Science, Wayne State University.

She was a Research Engineer in Luoyang Petrochemical Corporation, China, between 1993 and 1997. Her main research interests include process

modeling, process optimization, and pollution prevention in the metal and polymer coating and chemical processing industries.

Ms. Lou is a recipient of the AESF National Surface Finishing Scholarship and the 1999 Thomas Rumble Graduate Fellowship of Wayne State University.

J. P. Gong received the B.S., M.S., and Ph.D. degrees from Zhejiang University, China, all in chemical engineering.

He was a Research Associate in the Department of Chemical Engineering and Materials Science, Wayne State University, Detroit, MI, between 1995 and 1997. He is currently a Senior Engineer with Aspen Technology, Houston, TX. His current research interests are process modeling, simulation, control, and optimization, with applications to process improvement, pollution prevention, and waste minimization.



Thomas F. Edgar received the Ph.D. degree from Princeton University, Princeton, NJ, in 1971.

He has been a Faculty Member at University of Texas, Austin, since 1971. He is now the Associate Vice President for Academic Computing and Instructional Technology Services, University of Texas. He is also the George T. and Gladys H. Abell Chair in Engineering, University of Texas. He has over 175 publications, including several leading textbooks on these subjects. His main research interests include control system monitoring and diagnosis, microelectronics manufacturing, and modeling and control of reactive distillation columns.

Dr. Edgar was the President of American Automatic Control Council (AACC) in 1990, Chair of the Council for Chemical Research (CCR) in 1992, President of AIChE in 1997. His major honors include the AIChE Colburn Award and Computing in Chemical Engineering Award, the ASEE Westinghouse and Merian-Wiley Awards, the AACC Education Award, and the ISA Eckman Education Award.