Nonlinear Model Reference Adaptive Control with Embedded Linear Models

Richard B. McLain and Michael A. Henson*

Department of Chemical Engineering, Louisiana State University, Baton Rouge, Louisiana 70803-7303

We propose a nonlinear model reference adaptive control strategy in which a linear model (or a set of linear models) is embedded within the nonlinear controller. The technique is applicable to single-input, single-output nonlinear processes with stable zero dynamics and full-state feedback. The nonlinear control law is constructed by embedding linear controller gains obtained from a linear model or multiple linear models. The higher-order controller functions are approximated with locally supported radial basis functions centered in the state space. The number of basis functions is determined a priori, and an on-line pruning algorithm is utilized to ensure functions centered near the current operating point are active. Parameter update laws that guarantee (under certain assumptions) that the plant output asymptotically tracks the output of a linear reference model and the state vector remains bounded are derived via Lyapunov stability analysis. The proposed control strategy is compared to a linear state feedback controller and a linear multimodel adaptive controller using a nonlinear chemical reactor model.

1. Introduction

Most nonlinear control strategies require a nonlinear dynamic model of the process to be controlled.[1] Unfortunately, first-principles modeling is difficult to apply to processes that are poorly understood and/or highly complex. An alternative approach is to develop an empirical model from input/output data via nonlinear system identification.[2] While a variety of techniques have been proposed, there are a number of unresolved theoretical and practical issues that severely limit their applicability.

The modeling step can be eliminated entirely if a satisfactory method for direct construction of the controller is available. Model reference adaptive control (MRAC) provides a framework for synthesizing linear control laws in the absence of explicit linear models.[3] The lack of a systematic methodology for constructing the appropriate controller form and determining stable parameter update laws is the biggest obstacle associated with extending the MRAC approach to nonlinear systems. A number of investigators have proposed MRAC techniques for nonlinear systems.[4−13] However, these methods assume a suitable nonlinear model is available for controller design. This assumption is eliminated in the proposed method where the complications associated with nonlinear model development are replaced by the challenging problem of on-line controller construction.

We recently proposed a nonlinear MRAC strategy based on radial basis function (RBF) networks.[14] The technique is applicable to input/output linearizable (IOL) nonlinear systems with full-state feedback. It is important to emphasize that the IOL approach is restricted to nonlinear systems with well-defined relative degree and stable zero dynamics.[15,16] The major advantage of the proposed method is that controller design can be performed without a detailed nonlinear model. The only structural information needed is the relative degree and the sign of the high-frequency gain. Unknown controller functions are approximated with RBFs that are introduced only in regions of the state space where the closed-loop system actually evolves. Lyapunov stability analysis is used to derive parameter update laws that ensure that the state vector remains bounded and the plant output asymptotically tracks the output of a linear reference model. The proposed method has been successfully applied to a nonlinear biochemical reactor model.

A disadvantage of this approach is that the underlying process dynamics are completely unknown to the nonlinear controller prior to on-line adaptation. Consequently, the closed-loop system can exhibit poor transient performance, and even instability, during training. In this paper, we propose a nonlinear MRAC strategy based on local linear models that addresses this shortcoming. A linear model is used to synthesize a linear controller that provides a satisfactory closed-loop performance near the nominal operating point. The linear controller gains are embedded in the nonlinear controller by adapting RBFs to approximate higher-order terms in the Taylor series expansion of the unknown input/output linearizing controller functions. The tracking problem is addressed by embedding multiple linear models, each of which reflect the process dynamics around a desired operating point. The proposed controllers are compared to a linear state feedback controller and a linear multiple-model adaptive controller (MMAC) using a nonlinear chemical reactor example.

The remainder of the paper is organized as follows. In section 2, the basic nonlinear MRAC strategy is introduced and simulation results for the chemical reactor example are presented. In section 3, the nonlinear MRAC strategy with a single embedded linear model (ELM) is described and compared to a fixed gain state feedback controller. In section 4, a novel MMAC technique is developed by extending the controller design method in section 3 to handle multiple linear models. The nonlinear MMAC controller is compared

* To whom correspondence should be addressed. Phone: 225-388-3690. Fax: 225-388-1476. E-mail: henson@che.lsu.edu.

10.1021/ie990088t CCC: $19.00
Published on Web 07/12/2000

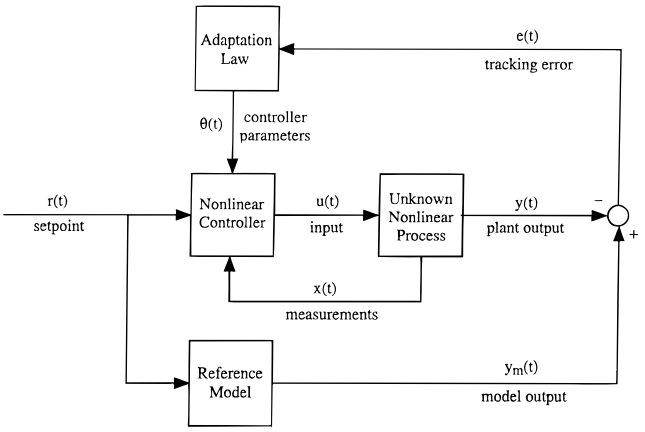

Figure 1. Block diagram of the nonlinear MRAC strategy.

to a linear MMAC controller. Conclusions are presented in section 5.

2. Basic Nonlinear MRAC Strategy

Consider a nonlinear system of the form

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\, u$$
$$y = h(\mathbf{x}) \tag{1}$$

where $\mathbf{x}$ is an n-dimensional vector of measured state variables, u is a scalar-manipulated input, y is a scalar-controlled output, and the functions $f(\mathbf{x})$ and $g(\mathbf{x})$ are unknown. The objective is to make the controlled output (y) track the output of a linear reference model (y_m). Figure 1 shows a simplified block diagram of the nonlinear MRAC scheme. The nonlinear controller uses the setpoint (r) and the state vector ($\mathbf{x}$) to compute the manipulated input (u) introduced to the plant. The plant output is compared to the output of a linear reference model that represents the desired setpoint response of the closed-loop system. The tracking error (e) is used to adapt the controller parameters (θ) such that the desired closed-loop response is obtained asymptotically. The following reference model is appropriate for nonlinear systems of relative degree 1:

$$\dot{y}_m = -\gamma y_m + \gamma y_{sp} \tag{2}$$

where y_{sp} is the setpoint and $\gamma > 0$ is a controller tuning parameter. The input/output linearizing control law that achieves the model-matching objective is[16,17]

$$u = \frac{-L_f h(\mathbf{x}) - \gamma h(\mathbf{x}) + \gamma y_{sp}}{L_g h(\mathbf{x})} \equiv \frac{-\alpha^*(\mathbf{x}) + \gamma y_{sp}}{\beta^*(\mathbf{x})} \tag{3}$$

where $L_f h(\mathbf{x})$ and $L_g h(\mathbf{x})$ are Lie derivatives. This control law cannot be implemented if the functions $f(\mathbf{x})$ and $g(\mathbf{x})$ are unknown. We propose to construct the controller functions $\alpha^*(\mathbf{x})$ and $\boldsymbol{\beta}^*(\mathbf{x})$ directly via on-line adaptation. The development presented below can be extended to higher relative degree systems using the filtered regressor approach to derive the parameter update laws.[11,14]

2.1. Nonlinear Controller Design. The goal is to construct on-line estimates of the controller functions $\alpha^*(\mathbf{x})$ and $\boldsymbol{\beta}^*(\mathbf{x})$ such that input/output linearization is achieved asymptotically given measurements of the state variables $\mathbf{x}(t)$ and the sign of $\boldsymbol{\beta}^*(\mathbf{x})$, which is the nonlinear analogue of the high-frequency gain.[14] The functions are approximated as

$$\alpha^*(\mathbf{x}) \cong \sum_{i=1}^{N} \alpha_i \phi_i(\mathbf{x}) = \boldsymbol{\alpha}^T \boldsymbol{\phi}(\mathbf{x})$$
$$\boldsymbol{\beta}^*(\mathbf{x}) \cong \sum_{i=1}^{N} \beta_i \phi_i(\mathbf{x}) = \boldsymbol{\beta}^T \boldsymbol{\phi}(\mathbf{x}) \tag{4}$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are vectors of time-varying controller parameters, $\boldsymbol{\phi}(\mathbf{x})$ is a vector of basis functions, and N is the number of basis functions employed. The resulting control law has the form

$$u = \frac{-\boldsymbol{\alpha}^T \boldsymbol{\phi}(\mathbf{x}) + \gamma y_{sp}}{\boldsymbol{\beta}^T \boldsymbol{\phi}(\mathbf{x})} \tag{5}$$

We utilize a locally supported RBF of the form[18]

$$\phi(r) = \begin{cases} (1-r)^4(1 + 4r + 3r^2 + 0.75r^3) & r \in [0, 1] \\ 0 & \text{elsewhere} \end{cases} \tag{6}$$

$$r^2 = \sum_{j=1}^{n} \frac{(x_j - c_j)^2}{a_j^2}$$

where n is the number of state variables, c is the basis function center, and a_j are scaling parameters. As compared to RBFs with global support such as the Gaussian,[10] the locally supported basis function (6) offers computational advantages because only a subset of the controller parameters have to be updated at any particular time.

Two assumptions are invoked to facilitate Lyapunov design of the parameter update laws. The first assumption that $\boldsymbol{\beta}^T \boldsymbol{\phi}(\mathbf{x}) \neq 0$ ensures that the nonlinear control law (5) remains well-defined. Because the sign of $\boldsymbol{\beta}^*(\mathbf{x})$ is assumed to be known, this condition often can be satisfied by careful initialization of the $\boldsymbol{\beta}$ controller parameters. The second assumption ensures the existence of "true" controller parameters $\boldsymbol{\alpha}^*$ and $\boldsymbol{\beta}^*$ that achieve model matching:

$$\boldsymbol{\alpha}^{*T} \boldsymbol{\phi}(\mathbf{x}) = L_f h(\mathbf{x}) + \gamma h(\mathbf{x})$$
$$\beta^{*T} \boldsymbol{\phi}(\mathbf{x}) = L_g h(\mathbf{x}) \tag{7}$$

This implies that perfect estimation of the controller functions throughout the entire state space is possible. This assumption does not strictly hold in practice, but results for globally supported RBFs show that the controller functions can be approximated arbitrarily well on a compact set if a sufficient number of basis functions are employed.[19] It is important to note that the condition (7) is not required to successfully apply the adaptive control strategy.

Under the above assumptions, the dynamics of the tracking error $e \equiv y_m - y$ can be written as

$$\dot{e} = -\gamma e + \boldsymbol{\Psi}_1^T \boldsymbol{\phi}(\mathbf{x}) + \boldsymbol{\Psi}_2^T \boldsymbol{\phi}(\mathbf{x})\, u \tag{8}$$

where $\boldsymbol{\Psi}_1 \equiv \boldsymbol{\alpha} - \boldsymbol{\alpha}^*$ and $\boldsymbol{\Psi}_2 \equiv \boldsymbol{\beta} - \boldsymbol{\beta}^*$ are parameter error vectors. The form of the error dynamics suggests the following parameter update laws:[3]

$$\dot{\boldsymbol{\Psi}}_1 = \dot{\boldsymbol{\alpha}} = -\eta_1 e \boldsymbol{\phi}(\mathbf{x})$$
$$\dot{\boldsymbol{\Psi}}_2 = \dot{\boldsymbol{\beta}} = -\eta_2 e \boldsymbol{\phi}(\mathbf{x})\, u \tag{9}$$

where $\eta_i > 0$ are adjustable adaptation gains.

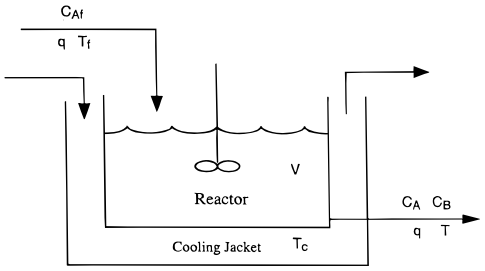

Figure 2. Continuous stirred tank reactor.

2.2. Basis Center Generation. The analysis presented above is based on the assumption that the RBF centers are fixed. This implies that basis functions are placed throughout the entire state space because trajectories of the closed-loop system cannot be predicted a priori. In our previous work,[14,18] we have addressed this problem by placing potential locations for basis function centers on a regular grid in the state space. A particular basis function is activated only if the closed-loop system evolves "near" its center. A disadvantage of this approach is that a very large number of basis functions may be activated if the process is high-dimensional and/or operates in several regions (e.g., a chemical reactor with several different steady-state operating conditions). This results in a large number of adjustable controller parameters, each of which has an associated differential equation of the form (9).

To address this shortcoming, we propose an on-line pruning algorithm. The mesh size is determined from a priori estimates of the smoothness of the unknown controller functions.[10] Then the scaling factors (a_j) are chosen to fix the coverage of a single basis function. This allows the maximum number of basis functions that can be active at any particular time to be determined. The maximum number of active functions is used as the fixed size of the network (N). Only active functions centered near the current operating point are contained in the network. As the operating point changes, new active centers are added and old inactive centers are pruned. The proposed method allows a simple initialization procedure because the centers being added/pruned are far removed from the current operating point and their contribution is small. Although stability and convergence results are derived assuming fixed centers throughout the entire state space, the simulation results in sections 3 and 4 demonstrate that the proposed center placement scheme can yield good closed-loop performance.

2.3. Simulation Results. The chemical reactor shown in Figure 2 is used to evaluate the nonlinear MRAC strategy. The model equations for a single irreversible reaction A → B are[20]

$$\dot{C}_A = \frac{q}{V}(C_{Af} - C_A) - k_0 \exp\left(-\frac{E}{RT}\right)C_A$$

$$\dot{T} = \frac{q}{V}(T_f - T) + \frac{(-\Delta H)}{\rho C_p}k_0 \exp\left(-\frac{E}{RT}\right)C_A + \frac{UA}{V\rho C_p}(T_c - T) \quad (10)$$

where C_A is the reactor concentration of component A, T is the reactor temperature, T_c is the temperature of the fluid in the cooling jacket, and T_f is the temperature of the feed stream. The remaining variables are defined elsewhere.[20] Nominal operating conditions are given in Table 1. The manipulated input and controlled output are the coolant temperature ($u = T_c$) and the reactor temperature ($y = T$), respectively. The resulting nonlinear system has relative degree 1. The input is constrained as follows to maintain feasible operation: 280 K ≤ T_c ≤ 370 K. The tuning parameters are chosen as $\gamma = 2$ min^{-1}, $\eta_1 = 100$, and $\eta_2 = 0.005$. The mesh size for centers is 0.05 g/L and 5 K for C_A and T, respectively. The basis functions are scaled with $a_1 = 0.2$ g/L and $a_2 = 15$ K. For the following simulations, a fixed network consisting of 24 basis functions is used.

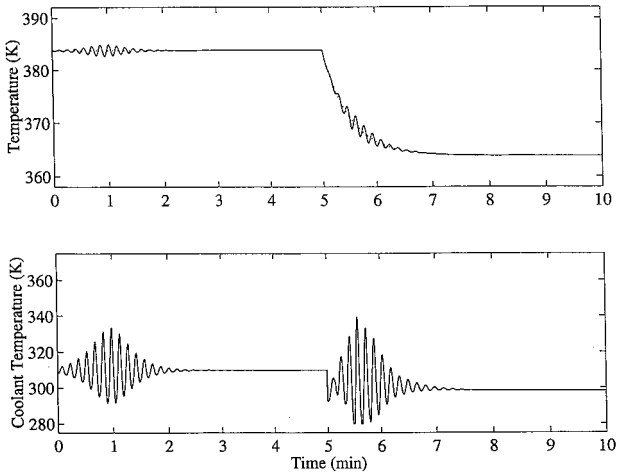

Figure 3. Nonlinear MRAC for setpoint change.

Table 1. Nominal Operating Conditions

variable	value	variable	value
q	100 L/min	E/R	8750 K
C_{Af}	1 mol/L	k_0	7.2×10^{10} min^{-1}
T_f	350 K	UA	5×10^4 J/min·K
V	100 L	ρ	1000 g/L
$(-\Delta H)$	5×10^4 J/mol	C_p	0.239 J/g·K

The performance of the nonlinear MRAC controller for a setpoint change from a stable steady state where $T = 383.8$ K to an unstable steady state where $T = 363.8$ K is shown in Figure 3. The temperature tracks the output of the reference model, but it exhibits bursting behavior before and after the setpoint change. The bursting appears to be caused by initialization of the RBF weights when new centers are introduced to the network. The weights must be carefully initialized such that the control law remains well-behaved. We have not found initial weights or controller tuning parameters that eliminate the bursting behavior. Figure 4 shows the regulatory performance at an unstable steady state for a feed temperature (T_f) disturbance from the nominal value (350 K) to a larger value (380 K). The controller provides excellent disturbance rejection as the reactor temperature is maintained within 1 K of the setpoint. However, the input exhibits bursting behavior during initialization. These results provide the motivation for modifying the nonlinear MRAC strategy to obtain improved transient performance.

3. Nonlinear MRAC with an ELM

The nonlinear MRAC technique is based on the assumption that the underlying process dynamics are completely unknown prior to on-line adaptation. As shown for the CSTR example, this can result in a poor

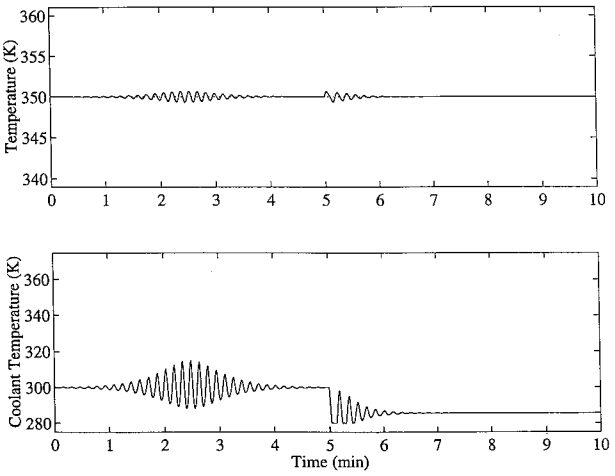

Figure 4. Nonlinear MRAC for feed temperature disturbance.

transient performance, and even instability, during training. In many process applications, some dynamic information is known or can be obtained from plant data. In particular, linear dynamic models can be developed for nominal operating points using well-developed linear system identification techniques.[21] These linear models can be used to synthesize linear controllers that provide approximate model matching near the associated operating points. In this section, we show how the controller gains obtained from a single linear model can be embedded within the nonlinear controller to yield a nonlinear MRAC strategy with a simpler initialization procedure and an improved transient performance. We focus on the relative degree 1 case. As shown in the Appendix, the following development can be extended to higher relative degree systems using the filtered regressor approach.[11,14]

3.1. Taylor Series Expansion. Consider the linear model

$$\dot{x}' = \mathbf{A}x' + \mathbf{b}u'$$
$$y' = \mathbf{c}x' \tag{11}$$

where $x' = x - \bar{x}$, $u' = u - \bar{u}$, and $y' = y - \bar{y}$ are deviation variables and $\bar{x}$, $\bar{u}$, and $\bar{y}$ represent the steady-state operating points of interest. The matrices **A**, **b**, and **c** can be determined using standard linear system identification techniques.[21] The linear controller that provides local model matching with respect to the reference model (2) is

$$u' = \frac{-\mathbf{cA}x' - \gamma\mathbf{c}x' + \gamma y'_{sp}}{\mathbf{cb}} \equiv \frac{-k_1x' + \gamma y'_{sp}}{k_2} \tag{12}$$

where k_1 and k_2 are linear controller gains.

Now consider the nonlinear control law (3), which can be rewritten as

$$\boldsymbol{\alpha}^*(\mathbf{x}) + \boldsymbol{\beta}^*(\mathbf{x})\, u = \gamma y_{sp} \tag{13}$$

The linear approximation of this equation about the steady state ($\bar{x}$, $\bar{u}$, $\bar{y}_{sp}$) is

$$\left[\frac{\partial\boldsymbol{\alpha}^*(\mathbf{x})}{\partial\mathbf{x}} + \bar{u}\,\frac{\partial\boldsymbol{\beta}^*(\mathbf{x})}{\partial\mathbf{x}}\right]_{\bar{x}}(x - \bar{x}) + \beta^*(\bar{x})\,(u - \bar{u}) = \gamma(y_{sp} - \bar{y}_{sp}) \tag{14}$$

The linear approximation can be written as

$$k_1x' + k_2u' = \gamma y'_{sp} \tag{15}$$

where k_1 and k_2 are the linear controller gains that are *embedded* within the nonlinear MRAC controller derived below.

Consider Taylor series expansions of the unknown nonlinear controller functions:

$$\boldsymbol{\alpha}^*(\mathbf{x}) = \alpha^*(\bar{x}) + \left[\frac{\partial\boldsymbol{\alpha}^*(\mathbf{x})}{\partial\mathbf{x}}\right]_{\bar{x}}(x - \bar{x}) + \frac{1}{2!}\left[\frac{\partial^2\boldsymbol{\alpha}^*(\mathbf{x})}{\partial\mathbf{x}^2}\right]_{\bar{x}}(x - \bar{x})^2 + \ldots$$

$$\boldsymbol{\beta}^*(\mathbf{x}) = \beta^*(\bar{x}) + \left[\frac{\partial\boldsymbol{\beta}^*(\mathbf{x})}{\partial\mathbf{x}}\right]_{\bar{x}}(x - \bar{x}) + \frac{1}{2!}\left[\frac{\partial^2\boldsymbol{\beta}^*(\mathbf{x})}{\partial\mathbf{x}^2}\right]_{\bar{x}}(x - \bar{x})^2 + \ldots \tag{16}$$

As shown in the Appendix, substitution of these expansions into (13) yields

$$[k_1x' + \tilde{\alpha}(x')] + [k_2 + \tilde{\beta}(x')]u' = \gamma y'_{sp} \tag{17}$$

where $\tilde{\alpha}(x')$ represents second-order and higher terms in $\alpha^*(x)$ and $\beta^*(x)\,\bar{u}$, while $\tilde{\beta}(x')$ represents first-order and higher terms in $\beta^*(x)$. Thus, the input/output linearizing control law (3) has the following deviation form:

$$u' = \frac{-[k_1x' + \tilde{\alpha}(x')] + \gamma y'_{sp}}{k_2 + \tilde{\beta}(x')} \tag{18}$$

3.2. Parameter Estimation. By embedding the linear controller gains as in (18), the nonlinear controller design problem is reduced to approximating the higher-order functions $\tilde{\alpha}(x')$ and $\tilde{\beta}(x')$. We assume the functions can be represented as

$$\tilde{\alpha}(x') = \sum_{i=1}^{N}\alpha_i^*\phi_i(x') = \alpha^{*\mathrm{T}}\phi(x')$$
$$\tilde{\beta}(x') = \sum_{i=1}^{N}\beta_i^*\phi_i(x') = \beta^{*\mathrm{T}}\phi(x') \tag{19}$$

where $\boldsymbol{\alpha}^*$ and $\boldsymbol{\beta}^*$ are vectors of unknown constant parameters, $\phi(x')$ is a vector of basis functions, and N is the number of basis functions. As discussed previously, in practice the relations (19) will hold only in an approximate sense. The control law (18) can be rewritten as

$$u' = \frac{-[k_1x' + \boldsymbol{\alpha}^{*\mathrm{T}}\phi(x')] + \gamma y'_{sp}}{k_2 + \boldsymbol{\beta}^{*\mathrm{T}}\phi(x')} \tag{20}$$

An implementable control law is obtained by replacing the unknown controller parameters with adjustable parameters $\boldsymbol{\alpha}(t)$ and $\boldsymbol{\beta}(t)$

$$u' = \frac{-[k_1x' + \boldsymbol{\alpha}^{\mathrm{T}}\phi(x')] + \gamma y'_{sp}}{k_2 + \boldsymbol{\beta}^{\mathrm{T}}\phi(x')} \tag{21}$$

where we assume $k_2 + \boldsymbol{\beta}^{\mathrm{T}}\phi(x') \neq 0$ to ensure the control law remains well-defined. Because k_2 and $\boldsymbol{\beta}^*(x')$ have

Table 2. Linear Models for the Chemical Reactor

model number	stable model	steady state			linear state space matrices		
		C_A	T	T_c	**A**	**b**	**c**
1	yes	0.9435	314.6	292.0	$\begin{bmatrix}-1.06 & -0.005\\ 12.5 & -2.05\end{bmatrix}$	$\begin{bmatrix}0\\ 2.09\end{bmatrix}$	[0 1]
2	no	0.5	350	300	$\begin{bmatrix}-2.00 & -0.0357\\ 209.2 & 4.38\end{bmatrix}$	$\begin{bmatrix}0\\ 2.09\end{bmatrix}$	[0 1]
3	yes	0.1	383.8	309.9	$\begin{bmatrix}-10.0 & -0.0536\\ 1889 & 8.13\end{bmatrix}$	$\begin{bmatrix}0\\ 2.09\end{bmatrix}$	[0 1]

the same sign, this assumption usually can be satisfied by initializing the controller with $\boldsymbol{\beta}(0) = 0$ and using an adaptation gain that makes $\boldsymbol{\beta}^T\phi(x')$ an order of magnitude smaller than k_2. This initialization procedure is considerably simpler than that required for the nonlinear MRAC technique without an ELM. The locally supported RBF (6) is used to approximate the unknown controller functions, and the RBF centers are allocated as discussed in section 2.2.

Update laws for the controller parameters $\boldsymbol{\alpha}(t)$ and $\boldsymbol{\beta}(t)$ are derived via Lyapunov stability analysis. The derivative of the output along the system trajectories is

$$\dot{y} = L_f h(\mathbf{x}) + L_g h(\mathbf{x})\, u = \alpha^*(\mathbf{x}) - \gamma h(\mathbf{x}) + \boldsymbol{\beta}^*(\mathbf{x})\, u \tag{22}$$

The following relation is obtained from (22) in the Appendix:

$$\dot{y} + = \gamma y = -\boldsymbol{\Psi}_1^T\phi(x') - \boldsymbol{\Psi}_2^T\phi(x')\, u' + \gamma y_{sp} \tag{23}$$

where $\boldsymbol{\Psi}_1 = \boldsymbol{\alpha} - \boldsymbol{\alpha}^*$ and $\boldsymbol{\Psi}_2 = \boldsymbol{\beta} - \boldsymbol{\beta}^*$. The tracking error $e \equiv y_m - y$ has the dynamics

$$\dot{e} = -\gamma e + \boldsymbol{\Psi}_1^T\phi(x') + \boldsymbol{\Psi}_2^T\phi(x')\, u' \tag{24}$$

This error equation suggests the following gradient update laws:[3]

$$\dot{\boldsymbol{\Psi}}_1 = \dot{\boldsymbol{\alpha}} = -\eta_1 e\phi(x')$$
$$\dot{\boldsymbol{\Psi}}_2 = \dot{\boldsymbol{\beta}} = -\eta_2 e\phi(x')\, u' \tag{25}$$

where $\eta_i > 0$ are adjustable adaptation gains.

Closed-loop stability can be analyzed using the Lyapunov function:

$$V = \frac{e^2}{2} + \frac{1}{2\eta_1}\boldsymbol{\Psi}_1^T\boldsymbol{\Psi}_1 + \frac{1}{2\eta_2}\boldsymbol{\Psi}_2^T\boldsymbol{\Psi}_2$$

The following assumptions are invoked:

1. The nonlinear system (1) has well-defined relative degree and zero dynamics that are exponentially stable and Lipschitz continuous.[11]

2. The matching conditions (19) are satisfied.

3. The control law (21) remains well-defined in the sense that $k_2 + \boldsymbol{\beta}^T\phi(x') \neq 0$.

Under these conditions, the derivative of V along trajectories of the error system is $\dot{V} = -\gamma_1 e^2 \leq 0$. This establishes that e, $\boldsymbol{\Psi}_1$, and $\boldsymbol{\Psi}_2$ are bounded and that e is square integrable.[22] The exponential stability and Lipschitz continuity assumptions imposed on the zero dynamics ensure that x' is bounded and e is uniformly continuous.[11] It follows from Barbalat's Lemma[22] that $\lim_{t\to\infty} e(t) = 0$. This result demonstrates that the proposed method ensures boundedness and asymptotic tracking under rather idealized conditions. In the following section, the performance of the adaptive nonlinear controller under more realistic conditions is evaluated.

3.3. Simulation Results. The nonlinear MRAC strategy with an ELM is evaluated using the chemical reactor model described in section 2.3. The linear controller gains are obtained from one of the linear models in Table 2 (see below). The tuning parameters are chosen as $\gamma = 2\ \text{min}^{-1}$, $\eta_1 = 75$, and $\eta_2 = 0.005$. The mesh size and scaling factors are the same as those in section 2.3, except that the number of active centers is reduced to 20 to decrease the computational burden. The centers of the RBFs are determined on-line using the procedure described in section 2.2.

We compare the nonlinear controller to a linear state feedback controller designed using one of the linear models in Table 2 (see below). To include an explicit integral term, the controller design is based on a second-order reference model rather than the first-order model (2). The resulting control law has the form

$$u' = \frac{-\mathbf{cA}x' + \gamma_1(y'_{sp} - y') + \gamma_0\int_0^t (r' - y')\, d\tau}{\mathbf{cb}} \tag{26}$$

where $\gamma_i > 0$ are controller tuning parameters chosen such that the polynomial $s^2 + \gamma_1 s + \gamma_0$ is Hurwitz. The two controllers have been tuned similarly to ensure a fair comparison. The controller parameters are chosen in terms of a single tuning parameter ϵ for both controllers.[17] For this example $\epsilon = 0.5$ min, which is approximately one-third the open-loop time constant. The nonlinear controller design is based on the first-order reference model (2), yielding the tuning parameter $\gamma = \epsilon^{-1} = 2\ \text{min}^{-1}$. For the second-order reference model associated with the linear controller, the tuning parameters are $\gamma_1 = 2\epsilon^{-1} = 4\ \text{min}^{-1}$ and $\gamma_0 = \epsilon^{-2} = 4\ \text{min}^{-2}$.

Figure 5 shows the performance of the nonlinear MRAC controller with ELM and the linear controller for a setpoint change from a stable operating point to an unstable operating point. Both controllers are designed using the stable model 3 in Table 2. The nonlinear controller provides excellent tracking in the unstable region even though the ELM is stable. While this result is problem specific, it illustrates the robustness of the proposed adaptive controller. The linear controller produces very large input moves that hit the input constraints. As a result, the temperature exhibits large oscillations and the new setpoint is not achieved. This behavior is attributable to the use of a stable model for controller design. The advantage of the nonlinear MRAC controller with ELM over the nonlinear MRAC controller without ELM (Figure 3) also is apparent. The bursting behavior during initialization and transients is almost completely eliminated when the ELM is included.

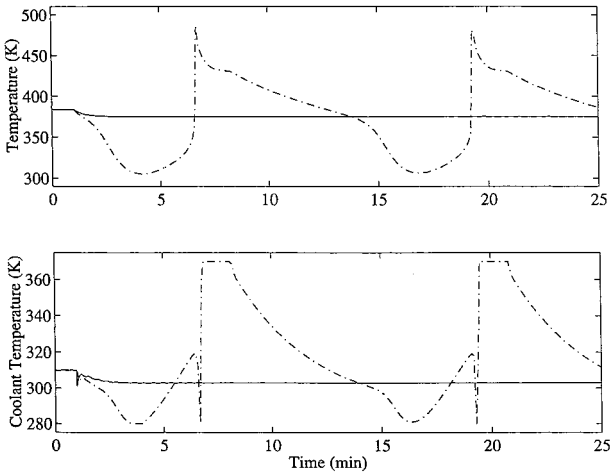

Figure 5. Nonlinear MRAC with stable ELM and linear state feedback for setpoint change: nonlinear MRAC (solid); linear (dash-dotted).

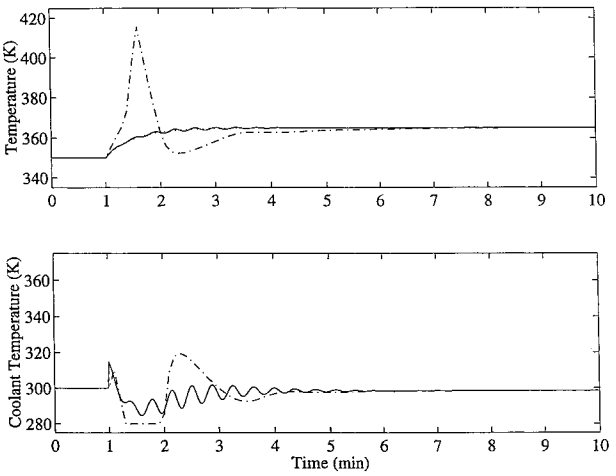

Figure 6. Nonlinear MRAC with unstable ELM and linear state feedback for setpoint change: nonlinear MRAC (solid); linear (dash-dotted).

Figure 6 shows the performance of the nonlinear MRAC controller with ELM and the linear controller for a setpoint change in the unstable operating region. Both controllers are designed using the unstable model 2 in Table 2. The linear controller produces a large overshoot in the temperature as the input saturates at the lower constraint. The nonlinear controller provides excellent tracking, but the input is slightly oscillatory.

Figure 7 shows the performance of the nonlinear MRAC controller with stable model 3 for several setpoint changes. Because a single linear model is used, the tracking deteriorates as the setpoint moves further from the region where the linear model is accurate. This provides motivation for embedding multiple linear models, as discussed in the following section. Figure 8 shows the regulatory performance of the nonlinear MRAC controller with ELM and the linear controller at an unstable steady state for a feed temperature disturbance from the nominal value (350 K) to a larger value (380 K). Both controllers are designed using unstable model 2. The linear controller yields very poor performance. The nonlinear controller with ELM effectively rejects the disturbance, and the input behavior is significantly improved as compared to the nonlinear controller without ELM (Figure 4).

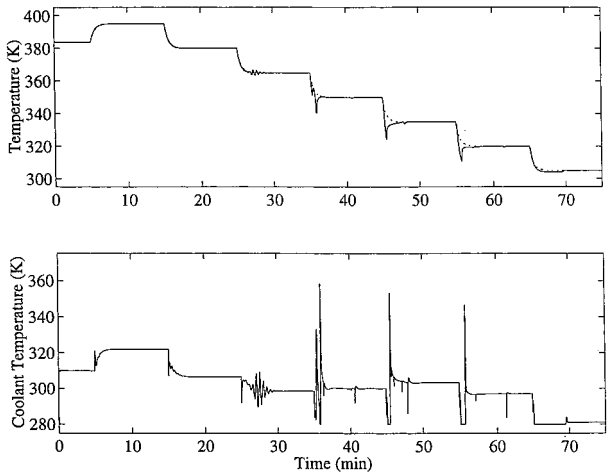

Figure 7. Nonlinear MRAC with stable ELM for multiple setpoint changes.

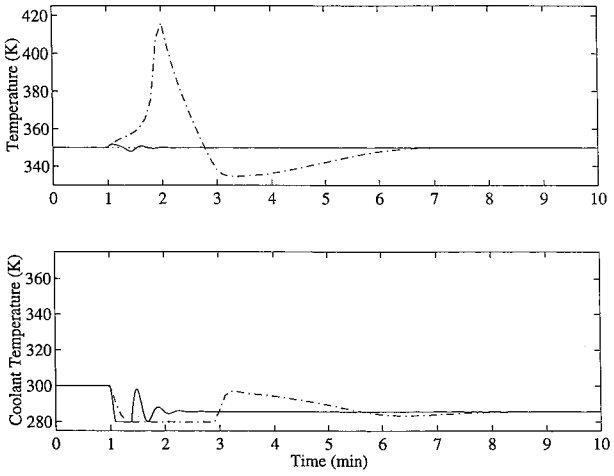

Figure 8. Nonlinear MRAC with unstable ELM and linear state feedback for feed temperature disturbance: nonlinear MRAC (solid); linear (dash-dotted).

4. Extension to Multiple Linear Models

Figures 5 and 6 demonstrate that the nonlinear MRAC strategy can provide excellent tracking performance when the setpoint is in the same region as that used in the development of the ELM. A degradation in performance is observed in Figure 7 when the setpoint moves into a region with stability characteristics different from those of the linear model. This is attributable to a fundamental change in the underlying process dynamics that makes the ELM inappropriate for control. Multiple linear models, each developed for a different operating regime, should provide better setpoint tracking over the entire operating range than a single linear model. This is particularly important for plants that operate in multiple regimes and transition between them (e.g., polymerization reactors producing multiple grades). The proposed method of embedding multiple linear models in the nonlinear MRAC controller can be viewed as a nonlinear extension of the linear MMAC approach.[23−25]

4.1. Controller Design. The objective of MMAC techniques is to use the "best" model or combination of models at the current operating point to calculate the next control move. Typically the models are linear because they can be obtained more readily than nonlinear models and they are more computationally ef-

ficient. Assuming that the linear models are available, the problem is reduced to combining the models such that linear controller gains can be calculated and incorporated into the nonlinear control law (21).

Recently a method for combining multiple linear state space models to generate a "global" state space model was proposed.[23] The basic idea is to generate the state space matrices of the global model by taking weighted sums of the state space matrices of the individual models. Bayesian statistics are used to compute weighting coefficients that reflect the relative validity of each individual model at the current operating point. We propose an analogous method for combining the linear controller gains associated with each linear model to generate a global linear controller. This approach is preferred to the global modeling method because we are interested in constructing the nonlinear controller directly and complications associated with computing the linear controller gains from the combined state space matrices are avoided.

Based on the above discussion, the following combination rules are used:

$$k_1 = \sum_{i=1}^{M} W_i(\mathbf{c}_i\mathbf{A}_i + \gamma\mathbf{c}_i) \qquad k_2 = \sum_{i=1}^{M} W_i(\mathbf{c}_i\mathbf{b}_i)$$

$$\bar{u} = \sum_{i=1}^{M} W_i\bar{u}_i \qquad \bar{x} = \sum_{i=1}^{M} W_i\bar{x}_i \quad (27)$$

where M is the number of linear models, $\mathbf{A}_i$, $\mathbf{b}_i$, and $\mathbf{c}_i$ are the state space matrices for the ith linear model, and W_i is the weight associated with the ith model. Note that the current steady-state operating point ($\bar{x}$, $\bar{u}$) is estimated as the weighted sum of the steady-state operating points of the individual models. Using Bayesian estimation,[23−25] the probability that the ith model at time k represents the plant is

$$p_i(k) = \frac{\exp(-\epsilon_i^{\mathrm{T}}(k)\,\mathbf{K}\epsilon_i(k))p_i(k-1)}{\sum_{j=1}^{M}[\exp(-\epsilon_j^{\mathrm{T}}(k)\,\mathbf{K}\epsilon_j(k))p_j(k-1)]} \quad (28)$$

where $\epsilon_i(k)$ is the normalized residual for the ith model and the diagonal matrix $\mathbf{K}$ is used to adjust the estimator responsiveness. The residual is computed as

$$\epsilon_i(k) = \mathbf{S}[x(k) - x_i(k)] \quad (29)$$

where $x(k)$ is the state measurement at time k, $x_i(k)$ is the state estimate obtained from the ith model at time k, and $\mathbf{S}$ is a diagonal scaling matrix. Because eq 28 is recursive, the probabilities must have a lower bound (δ) to prevent them from remaining at zero. This is achieved by renormalizing the probabilities to determine the actual weights:[25]

$$W_i(k) = \frac{p_i(k)}{\sum_{j=1}^{M} p_j(k)} \qquad p_i(k) > \delta$$

$$W_i(k) = 0 \qquad p_i(k) = \delta \quad (30)$$

4.2. Simulation Results. For the chemical reactor described in section 2.3, three linear models corresponding to different operating points are given in Table 2. Because model 2 is unstable, the Bayesian estimation scheme cannot use simple open-loop observers to generate the state predictions. To ensure bounded state estimates, a closed-loop observer is designed for each model

$$\hat{x}_i' = \mathbf{A}_i\hat{x}_i' + \mathbf{b}_i u_i' + L_i(y' - y_i') \quad (31)$$

where L_i is the observer gain. Typically, the observer poles are chosen such that the observer responds significantly faster than the controller. In this example, the gains L_i are chosen to place the observer poles at -20. The residual can be rewritten as

$$\epsilon_i(k) = \mathbf{S}[x(k) - (\hat{x}_i'(k) + \bar{x}_i)] \quad (32)$$

where

$$\mathbf{S} = \begin{bmatrix} \frac{1}{0.5} & 0 \\ 0 & \frac{1}{350} \end{bmatrix} \quad (33)$$

We compare the proposed nonlinear MMAC controller to a linear MMAC controller, both of which use the state estimation scheme described above. The tuning parameters for the nonlinear MMAC controller are chosen as $\gamma = 2$ min^{-1}, $\eta_1 = 50$, $\eta_2 = 0.005$, $\delta = 0.005$, and

$$\mathbf{K} = \begin{bmatrix} 25 & 0 \\ 0 & 35 \end{bmatrix} \quad (34)$$

The centers of the RBFs are determined on-line using the procedure in section 2.2. The mesh size and scaling factors are the same as those given in section 2.3, but for this example the network consists of only 20 active basis functions. The control law for the linear MMAC scheme is (26), and the tuning parameters are chosen as $\gamma_1 = 2$ min^{-1}, $\gamma_0 = 1$ min^{-2}, $\delta = 0.005$, and

$$\mathbf{K} = \begin{bmatrix} 75 & 0 \\ 0 & 75 \end{bmatrix} \quad (35)$$

In the following simulations, the nonlinear controller is initialized such that the linear model with the nominal operating point closest to the initial operating point is assigned a weighting of $1 - 2\delta$, while the other linear models are both assigned a weighting of δ.

Figure 9 shows the servo performance of the linear MMAC controller for setpoint changes across the operating space. For the second change, the setpoint is not attained because the estimator incorrectly switches to the unstable model. The input is not well behaved during this transition. Increasing the convergence factor causes the controller to switch repeatedly between stable and unstable models, which leads to closed-loop instability. The linear MMAC controller also performs unacceptably for the sixth setpoint change. This behavior is due to the transition from the unstable model to stable model 1. However, the controller performs reasonably well when the setpoint remains in an operating regime where the local stability characteristics are the same as those of the linear model.

Figure 10 shows the servo performance of the proposed nonlinear MMAC controller for the same setpoint sequence as that in Figure 9. The controller performs very well, although the input is somewhat oscillatory. It is interesting to note that the first transition between

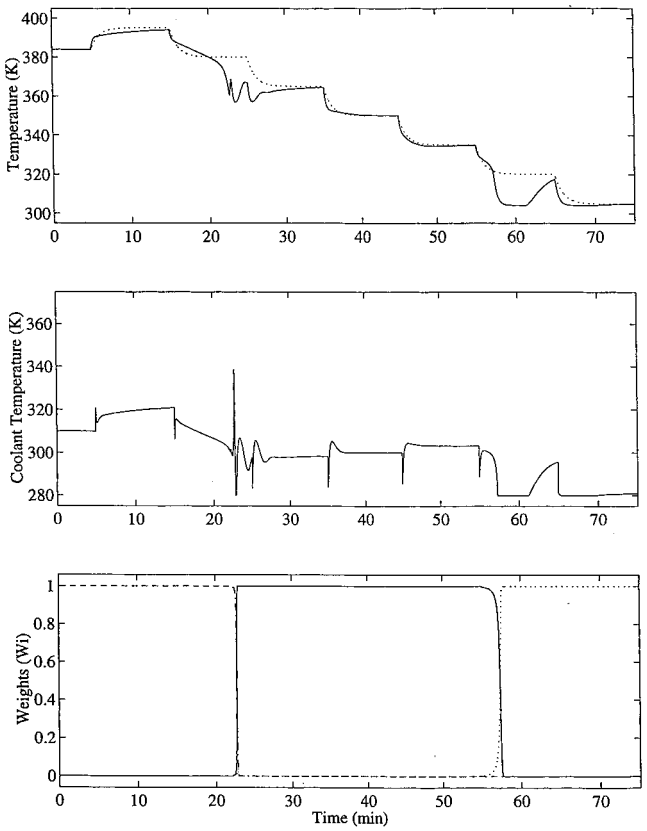

Figure 9. Linear MMAC for multiple setpoint changes: W_1 (dotted); W_2 (solid); W_3 (dashed).

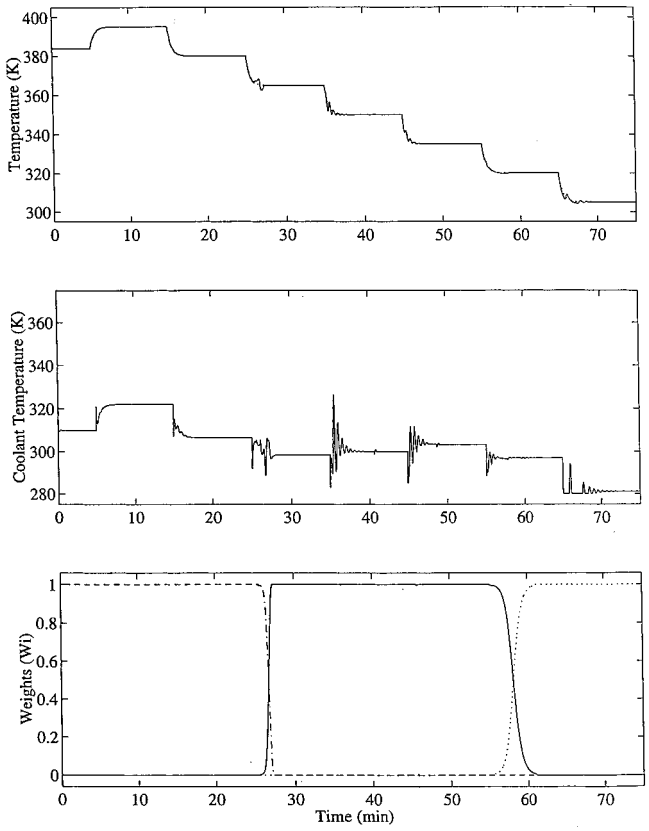

Figure 10. Nonlinear MMAC for multiple setpoint changes: W_1 (dotted); W_2 (solid); W_3 (dashed).

models occurs quite rapidly, whereas the second transition is considerably slower. For all setpoint changes, the temperature is maintained within 2 K of the reference model. The nonlinear MMAC controller clearly outperforms the nonlinear MRAC with a single ELM (Figure 7).

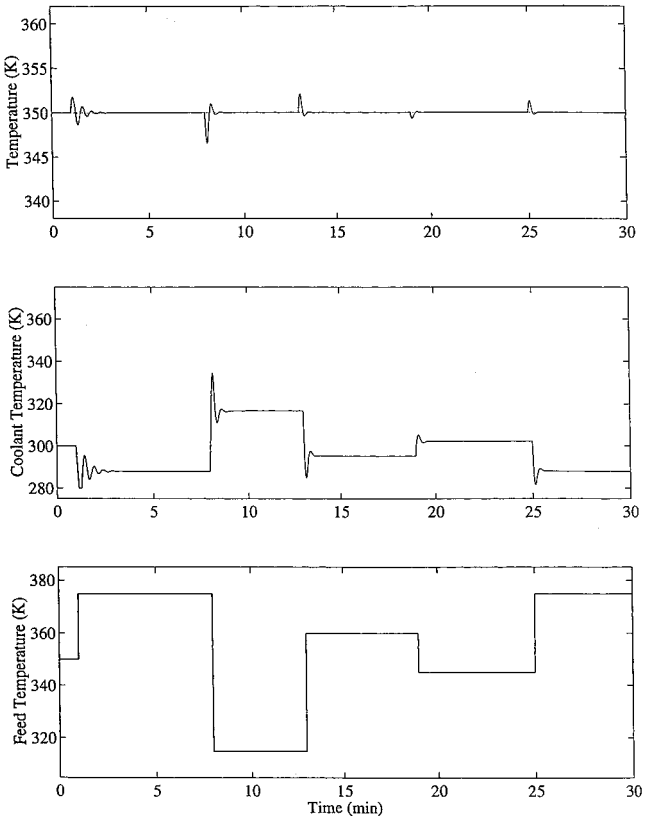

Figure 11. Nonlinear MMAC for random feed temperature disturbances.

Figure 11 shows the regulatory performance of the nonlinear MMAC controller for a series of feed temperature disturbances of random magnitude and duration. The controller provides excellent disturbance rejection as the temperature is maintained within 3 K of the setpoint. The input is well behaved, and the estimator correctly selects the unstable model throughout the test (not shown). Figure 12 shows the servo performance of the nonlinear MMAC controller when the unstable model is not used. For this test, the tuning is slightly modified with $\eta_1 = 100$ and

$$\mathbf{K} = \begin{bmatrix} 20 & 0 \\ 0 & 25 \end{bmatrix} \tag{36}$$

As expected, the controller performs very well in the upper and lower stable operating regions. Despite the lack of an unstable model, the controller is able to track setpoint changes in the unstable region. However, the input is rather oscillatory.

5. Conclusions

We have proposed a nonlinear adaptive control strategy that does not require a detailed dynamic model of the process to be controlled. The technique is applicable to single-input, single-output nonlinear systems with stable zero dynamics and full-state feedback. The proposed technique is based upon embedding a linear model within the nonlinear controller to improve the closed-loop performance during initialization and tran-

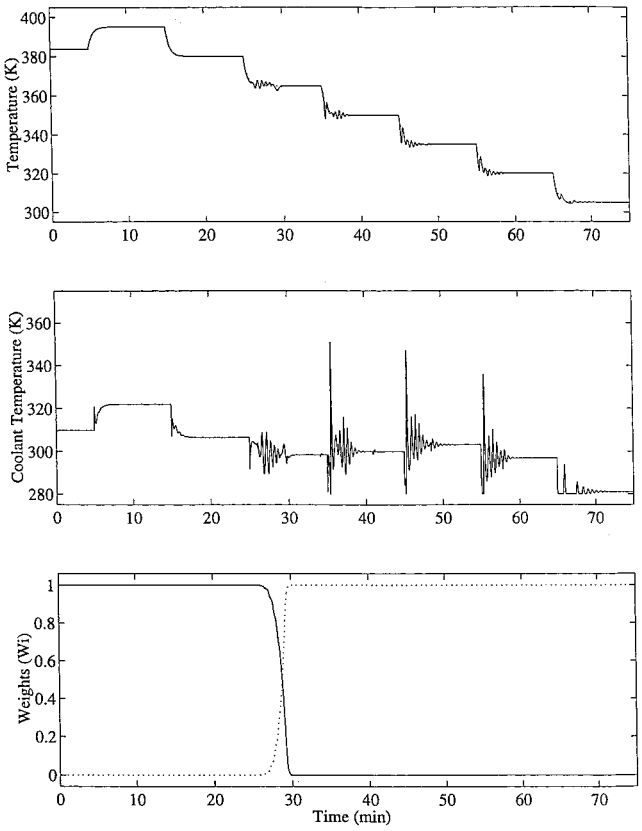

Figure 12. Nonlinear MMAC without unstable linear model for multiple setpoint changes: W_1 (dotted); W_2 (solid).

sients. Higher-order controller functions are approximated with locally supported radial basis functions that are linearly parametrized. The total number of basis functions used is determined a priori, and an on-line pruning algorithm is utilized such that the active functions are centered near the current operating point. Parameter update laws that ensure that the plant output asymptotically tracks the output of a linear reference model and the state vector remains bounded are derived via Lyapunov stability analysis. Bayesian estimation and combination rules are used to embed multiple linear models within the nonlinear controller. This yields a novel nonlinear MMAC scheme with the potential to produce faster transitions between operating points than is possible with linear MMAC techniques. The proposed strategies provide good servo and regulatory performances when applied to a nonlinear chemical reactor model.

Acknowledgment

Financial support from a NSF Career Development Award (CTS-9501368) and DuPont Co. is gratefully acknowledged.

Appendix

Detailed Development of Equation 17. Substitution of the expansions (16) into the controller (13) yields

$$\left(\boldsymbol{\alpha}^*(\bar{x}) + \left[\frac{\partial \boldsymbol{\alpha}^*(\mathbf{x})}{\partial \mathbf{x}}\right]_{\bar{x}}(x-\bar{x}) + \left[\frac{\partial^2 \boldsymbol{\alpha}^*(\mathbf{x})}{\partial \mathbf{x}^2}\right]_{\bar{x}}\frac{(x-\bar{x})^2}{2} + \ldots\right) + \left(\boldsymbol{\beta}^*(\bar{x}) + \left[\frac{\partial \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}}\right]_{\bar{x}}(x-\bar{x}) + \left[\frac{\partial^2 \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}^2}\right]_{\bar{x}}\frac{(x-\bar{x})^2}{2} + \ldots\right)\bar{u} + \left(\boldsymbol{\beta}^*(\bar{x}) + \left[\frac{\partial \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}}\right]_{\bar{x}}(x-\bar{x}) + \left[\frac{\partial^2 \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}^2}\right]_{\bar{x}}\frac{(x-\bar{x})^2}{2} + \ldots\right)u' = \gamma y'_{sp} + \gamma \bar{y}_{sp}$$

This equation can be simplified using the steady-state controller relation $\boldsymbol{\alpha}^*(\bar{x}) + \boldsymbol{\beta}^*(\bar{x})\,\bar{u} = \gamma \bar{y}_{sp}$. The result is

$$[k_1 x' + \tilde{\boldsymbol{\alpha}}(x')] + [k_2 + \tilde{\boldsymbol{\beta}}(x')]u' = \gamma y'_{sp}$$

where

$$\tilde{\boldsymbol{\alpha}}(x') = \left(\left[\frac{\partial^2 \boldsymbol{\alpha}^*(\mathbf{x})}{\partial \mathbf{x}^2}\right]_{\bar{x}}\frac{(x')^2}{2} + \left[\frac{\partial^3 \boldsymbol{\alpha}^*(\mathbf{x})}{\partial \mathbf{x}^3}\right]_{\bar{x}}\frac{(x')^3}{3!} + \ldots\right) + \left(\left[\frac{\partial^2 \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}^2}\right]_{\bar{x}}\frac{(x')^2}{2} + \left[\frac{\partial^3 \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}^3}\right]_{\bar{x}}\frac{(x')^3}{3!} + \ldots\right)\bar{u}$$

$$\tilde{\boldsymbol{\beta}}(x') = \left(\left[\frac{\partial \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}}\right]_{\bar{x}}(x') + \left[\frac{\partial^2 \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}^2}\right]_{\bar{x}}\frac{(x')^2}{2} + \ldots\right)$$

Detailed Development of Equation 23. Substitution of the expansion (16) into (22) yields

$$\dot{y} + \gamma y = \left(\boldsymbol{\alpha}^*(\bar{x}) + \left[\frac{\partial \boldsymbol{\alpha}^*(\mathbf{x})}{\partial \mathbf{x}}\right]_{\bar{x}}(x-\bar{x}) + \ldots\right) + \left(\boldsymbol{\beta}^*(\bar{x}) + \left[\frac{\partial \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}}\right]_{\bar{x}}(x-\bar{x}) + \ldots\right)\bar{u} + \left(\boldsymbol{\beta}^*(\bar{x}) + \left[\frac{\partial \boldsymbol{\beta}^*(\mathbf{x})}{\partial \mathbf{x}}\right]_{\bar{x}}(x-\bar{x}) + \ldots\right)u' + \gamma \bar{y}_{sp} - \gamma \bar{y}_{sp}$$

Substitution of the steady-state controller relation, $\boldsymbol{\alpha}^*(\bar{x}) + \boldsymbol{\beta}^*(\bar{x})\,\bar{u} = \gamma \bar{y}_{sp}$, allows this equation to be simplified as follows:

$$\dot{y} + \gamma y = [k_1 x' + \tilde{\boldsymbol{\alpha}}(x')] + [k_2 + \tilde{\boldsymbol{\beta}}(x')]u' + \gamma \bar{y}_{sp}$$

The approximation relations (19) are utilized to obtain

$$\begin{aligned}\dot{y} + \gamma y &= [k_1 x' + \boldsymbol{\alpha}^{*\mathrm{T}}\phi(x')] + [k_2 + \beta^{*\mathrm{T}}\phi(x')]u' + \gamma \bar{y}_{sp} + [k_2 + \boldsymbol{\beta}^{\mathrm{T}}\phi(x')]u' - [k_2 + \boldsymbol{\beta}^{\mathrm{T}}\phi(x')]u' \\ &= k_1 x' + \boldsymbol{\alpha}^{*\mathrm{T}}\phi(x') - (\boldsymbol{\beta}^{\mathrm{T}} - \boldsymbol{\beta}^{*\mathrm{T}})\phi(x')\,u' + \gamma \bar{y}_{sp} + [k_2 + \beta^{\mathrm{T}}\phi(\boldsymbol{\chi}')]u'\end{aligned}$$

Substitution of the nonlinear control law (21) into the final term yields

$$\begin{aligned}\dot{y} + \gamma y &= k_1 x' + \boldsymbol{\alpha}^{*\mathrm{T}}\phi(x') - (\boldsymbol{\beta}^{\mathrm{T}} - \boldsymbol{\beta}^{*\mathrm{T}})\phi(x')\,u' + \gamma \bar{y}_{sp} - [k_1 x' + \boldsymbol{\alpha}^{\mathrm{T}}\phi(x')] + \gamma y'_{sp} \\ &= -(\boldsymbol{\alpha}^{\mathrm{T}} - \boldsymbol{\alpha}^{*\mathrm{T}})\phi(x') - (\boldsymbol{\beta}^{\mathrm{T}} - \boldsymbol{\beta}^{*\mathrm{T}})\phi(x')\,u' + \gamma y_{sp} \\ &= -\boldsymbol{\Psi}_1^{\mathrm{T}}\phi(x') - \boldsymbol{\Psi}_2^{\mathrm{T}}\phi(x')\,u' + \gamma y_{sp}\end{aligned}$$

Higher Relative Degree Systems. We now extend the MRAC control technique to nonlinear systems of relative degree 2 and higher. The input/output linear-

izing control law is[16,17]

$$u = \frac{-L_f^r h(\mathbf{x}) - \gamma_r L_f^{r-1} h(\mathbf{x}) - ... - \gamma_1 h(\mathbf{x}) + \gamma_1 y_{sp}}{L_g L_f^{r-1} h(\mathbf{x})} \equiv \frac{-\alpha^*(\mathbf{x}) + \gamma_1 y_{sp}}{\beta^*(\mathbf{x})}$$

where y_{sp} is the setpoint, r is the relative degree, γ_i are controller tuning parameters, $L_f^i h(\mathbf{x})$ and $L_g L_f^{r-1} h(\mathbf{x})$ are Lie derivatives, and $\alpha^*(\mathbf{x})$ and $\beta^*(\mathbf{x})$ represent the true controller functions. The appropriate reference model is

$$y_m^r = -\gamma_r y_m^{r-1} - ... - \gamma_1 y_m + \gamma_1 y_{sp}$$

where γ_i are chosen such that $s^r + \gamma_r s^{r-1} + ... + \gamma_1$ is a Hurwitz polynomial. It is straightforward to show that the deviation form of the input/output linearizing control is (18) where $k_1 = -(\mathbf{c}\mathbf{A}^r + \gamma_r \mathbf{c}\mathbf{A}^{r-1} + ... + \gamma_1 \mathbf{c})$ and $k_2 = \mathbf{c}\mathbf{A}^{r-1}\mathbf{b}$. The implementable version of the control law is (21). As in the relative degree 1 case, two assumptions are needed to rigorously derive the parameter update laws. The first assumption is $k_2 + \beta^T \phi(\mathbf{x}) \neq 0$. The second assumption is the existence of "true" controller parameters α^* and β^* that satisfy (19).

Update laws for the controller parameters $\alpha(t)$ and $\beta(t)$ are derived as follows. The rth derivative of the output can be written as

$$y^r = L_f^r h(\mathbf{x}) + L_g L_f^{r-1} h(\mathbf{x})\, u = \alpha^*(\mathbf{x}) - \gamma_r L_f^{r-1} h(\mathbf{x}) - ... - \gamma_2 L_f h(\mathbf{x}) - \gamma_1 h(\mathbf{x}) + \beta^*(\mathbf{x})\, u$$

Following the development of (23), substitution of the expansions (16) into (37) yields

$$y^r + \gamma_r y^{r-1} + ... + \gamma_2 \dot{y} + \gamma_1 y = [k_1 x' + \bar{\alpha}(x')] + [k_2 + \bar{\beta}(x')]u' + \gamma_1 \bar{y}_{sp}$$

Substitution of the control law (21) after addition and subtraction of the term $[k_2 + \beta^T \phi(x')]u'$ yields

$$y^r + \gamma_r y^{r-1} + ... + \gamma_2 \dot{y} + \gamma_1 y = -\Psi_1^T \phi(x') - \Psi_2^T \phi(x')\, u' + \gamma_1 y_{sp}$$

where Ψ_1 and Ψ_2 are parameter error vectors defined previously. The dynamics of the tracking error $e \equiv y_m - y$ are

$$e^r + \gamma_r e^{r-1} + ... + \gamma_2 \dot{e} + \gamma_1 e = \Psi_1^T \phi(x') + \Psi_2^T \phi(x')\, u'$$

For the higher relative degree case, the gradient update laws (25) do not provide Lyapunov stability because the transfer function

$$M(s) \equiv \frac{1}{s^r + \gamma_r s^{r-1} + ... + \gamma_2 s + \gamma_1}$$

associated with the error dynamics is not strictly positive real.[3] This difficulty is overcome using the augmented error approach.[11] Define the parameter error Ψ and the regressor Φ as

$$\Psi \equiv \begin{bmatrix} \Psi_1 \\ \Psi_2 \end{bmatrix}, \quad \Phi \equiv \begin{bmatrix} \phi(x') \\ (\phi x')\, u' \end{bmatrix}$$

Now the error dynamics can be written as $e = M(s)[\Psi^T \Phi)]$, which represents the filtering of the time domain signal $\Psi^T \Phi$ by the stable transfer function $M(s)$. The "true" and estimated values of the controller parameters are defined as

$$\theta^* \equiv \begin{bmatrix} \alpha^* \\ \beta^* \end{bmatrix}, \quad \theta \equiv \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

The augmented error e_1 is defined as[11]

$$e_1 \equiv e + \theta^T M(s)[\Phi] - M(s)[\theta^T \Phi] \tag{37}$$

This relation allows e_1 to be computed from measurable signals. In general, $e_1 \neq e$ because the estimated parameters vary with time. By contrast, the "true" parameters are constant so $\theta^{*T} M(s)[\Phi] - M(s)[\theta^{*T} \Phi] = 0$. When this equation is subtracted from (37), an alternative representation of e_1 that is more convenient for analysis is obtained:

$$e_1 = e + \Psi^T M(s)[\Phi] - M(s)[\Psi^T \Phi] = \Psi^T M(s)[\Phi]$$

The form of this error equation suggests the following normalized gradient update law[3]

$$\dot{\Psi} = \dot{\theta} = \frac{-\eta e_1 \xi}{1 + \xi^T \xi}$$

where $\xi \equiv M(s)[\Phi]$ is the filtered regressor. Stability analysis for higher relative degree systems is considerably more complex than that shown for the relative degree 1 case because of the augmented error scheme. The reader is referred to the paper by Sastry and Isidori[11] for the type of detailed analysis required.

Note that the augmented error scheme can lead to high computational demands because of the introduction of a large number of filters. The total number of differential equations required to implement the parameter estimator is $2r(N + 1) + 2N$, where r is the relative degree and N is the number of basis functions used. Thus, the method is not well suited for nonlinear systems with high relative degree and/or that require a large number of basis functions. Fortunately, many chemical and biochemical systems have low relative degree.

Literature Cited

(1) Henson, M. A.; Seborg, D. E. *Nonlinear Process Control*; Prentice-Hall: Englewood Cliffs, NJ, 1997.

(2) Pearson, R. K.; Ogunnaike, B. A. Nonlinear process identification. In *Nonlinear Process Control*; Henson, M. A., Seborg, D. E., Eds.; Prentice-Hall: Englewood Cliffs, NJ, 1997; Chapter 2, pp 11−110.

(3) Sastry, S.; Bodson, M. *Adaptive Control*: *Stability, Convergence, and Robustness*; Prentice-Hall: New York, 1989.

(4) Chen, F. C.; Khalil, H. K. Adaptive control of nonlinear systems using neural networks. *Proceedings of the IEEE Conference on Decision and Control*, Honolulu, HI, 1990; pp 1707−1712.

(5) Chen, F. C.; Liu, C. C. Adaptively controlling nonlinear continuous-time systems using neural networks. *IEEE Trans. Autom. Control* **1994**, *AC-39*, 1306−1310.

(6) Kanellakopoulos, I.; Kokotovic, P. V.; Marino, R. An extended direct scheme for robust adaptive nonlinear control. *Automatica* **1991**, *27*, 247−255.

(7) Nam, K.; Arapostathis, A. A model reference adaptive control scheme for pure-feedback nonlinear systems. *IEEE Trans. Autom. Control* **1988**, *33*, 803−811.

(8) Polycarpou, M. M. Stable adaptive neural control scheme for nonlinear systems. *IEEE Trans. Autom. Control* **1996**, *AC-41*, 447−451.

(9) Polycarpou, M. M.; Ioannou, P. A. *Identification and control of nonlinear systems using neural network models*: *Design and stability analysis*; Technical Report MC-2563; University of Southern California: Los Angeles, CA, 1991.

(10) Sanner, R. M.; Slotine, J.-J. E. Gaussian networks for direct adaptive control. *IEEE Trans. Neural Networks* **1992**, *3*, 837−863.

(11) Sastry, S.; Isidori, A. Adaptive control of linearizable systems. *IEEE Trans. Autom. Control* **1989**, *AC-34*, 1123−1131.

(12) Sastry, S. S.; Kokotovic, P. V. Feedback linearization in the presence of uncertainties. *Int. J. Adaptive Control Signal Process.* **1988**, *2*, 327−346.

(13) Yesidirek, A.; Lewis, F. L. Feedback linearization using neural networks. *Automatica* **1995**, *31*, 1659−1664.

(14) McLain, R. B.; Henson, M. A.; Pottmann, M. Direct adaptive control of partially known nonlinear systems. *IEEE Trans. Neural Networks* **1999**, *10*, 714−721.

(15) Byrnes, C. I.; Isidori, A. Local stabilization of minimum-phase nonlinear systems. *Syst. Control Lett.* **1988**, *11*, 9−17.

(16) Isidori, A. *Nonlinear Control Systems*; Springer-Verlag: New York, 1989.

(17) Henson, M. A.; Seborg, D. E. Feedback linearizing control. In *Nonlinear Process Control*; Henson, M. A., Seborg, D. E., Eds.; Prentice-Hall: Englewood Cliffs, NJ, 1997; Chapter 4, pp 149−231.

(18) Pottmann, M.; Henson, M. A. Compactly supported radial basis functions for adaptive process control. *J. Process Control* **1997**, *7*, 345−356.

(19) Park, J.; Sandberg, I. W. Universal approximation using radial-basis-function networks. *Neural Comput.* **1991**, *3*, 3017−3034.

(20) Uppal, A.; Ray, W. H.; Poore, A. B. On the dynamic behavior of continuous stirred tank reactors. *Chem. Eng. Sci.* **1974**, *29*, 967.

(21) Ljung, L. *System Identification: Theory for the User*; Prentice-Hall: Engelwood Cliffs, NJ, 1989.

(22) Khalil, H. *Nonlinear Systems*; Macmillan: London, 1992.

(23) Banerjee, A.; Arkun, Y.; Pearson, R. K.; Ogunnaike, B. A. H_∞ control of nonlinear processes using multiple linear models. In *Multiple Model Approaches to Nonlinear Modelling and Control*; Murray-Smith, R., Johansen, T. A., Eds.; Taylor and Frances: London, 1997; Chapter 12, pp 293−305.

(24) He, W.; Kaufman, H.; Roy, R. Multiple model adaptive control procedure for blood pressure control. *IEEE Trans. Biomed. Eng.* **1986**, *BME-33*, 10−19.

(25) Schott, K. D.; Bequette, B. W. Multiple model adaptive control. In *Multiple Model Approaches to Nonlinear Modelling and Control*; Murray-Smith, R., Johansen, T. A., Eds.; Taylor and Frances: London, 1997; Chapter 11, pp 269−291.

Received for review February 8, 1999
Revised manuscript received May 17, 1999
Accepted May 19, 2000

IE990088T