ELSEVIER

J. Proc. Cont. Vol. 7, No. 5, pp. 345–356, 1997

Printed in Great Britain
0959-1524/97 $17.00 + 0.00

PII: S0959-1524(97)00012-7

Compactly supported radial basis functions for adaptive process control

Martin Pottmann* **and Michael A. Henson†**

*E.I. du Pont de Nemours and Company, Wilmington, DE 19880-0101, USA
†Department of Chemical Engineering, Louisiana State University, Baton Rouge, LA 70803-7303, USA

An adaptive nonlinear control strategy based on networks of compactly supported radial basis functions is proposed. The local influence of the basis functions allows efficient on-line adaptation that is performed using a gradient law, and new basis functions are added to the network only when new regions in state space are encountered and the prediction error exceeds a pre-specified tolerance. The approximate model is used to construct an input-output linearizing control law. The adaptive control strategy is applied to a nonlinear chemical reactor model. © 1997 Published by Elsevier Science Ltd

Keywords: adaptive control; artificial neural networks; nonlinear control; nonlinear identification; radial basis functions

Most nonlinear control strategies require an explicit mathematical model of the plant. Since fundamental models of many processes are often not available or extremely difficult to obtain, the identification and/or adaptation of nonlinear dynamic models from process data has received considerable attention in recent years[1]. Both empirical input/output models as well as state-space models have been used for process control purposes and a variety of modeling approaches have been reported in the literature. Here we only focus on techniques that utilize radial basis function (RBF) networks.

Since RBF networks can be applied to input-output and state-space modeling in both discrete-time and continuous-time, we start by introducing RBF networks as a general tool for multivariate function approximation. RBF networks approximate multivariate functions $F(x) : \mathrm{R}^n \rightarrow \mathrm{R}$ as linear combinations of univariate functions $\phi(r)$ in the following manner

$$F(x) \approx \sum_{i=1}^{N} a_i \phi(\| x - c_i \|) \quad x, c_i \in \mathrm{R}^n \tag{1}$$

where N is the number of basis functions, a_i are linear RBF coefficients, c_i are the RBF centers, and $\| \cdot \|$ denotes the Euclidean norm. RBFs with global support, such as the Gaussian[2–4], $\phi(r) = \exp(-\frac{r^2}{\beta})$, and the multiquadric functions,[5] $\phi(r) = (r^2 + \beta)^{\pm 1/2}$, have been used predominantly. There are two main approaches of nonlinear system modeling based on RBF networks.

Discrete-time input-output models

Most of the proposed RBF modeling approaches are based on discrete-time NARMAX (Nonlinear AutoRegressive Moving Average with eXogenous inputs) type model structures. While Artificial Neural Networks (ANN) of various architectures have been used to develop NARMAX process models, RBF modeling stands out in this context as a computationally very efficient approach because the partially linear RBF model parameterization allows standard stepwise regression methods to be applied for network training[2,5].

Referring back to the general RBF model in Equation (1), and considering the problem of modeling a SISO system, the vector x can be defined in terms of past inputs and outputs as follows

$$x = [y(k-1)...y(k-n_y)u(k-1)...u(k-n_u)]^T \tag{2}$$

and $F(x)$ would approximate the current output $y(k)$ as a nonlinear function of past inputs and outputs. While this approach has been used quite extensively for neural network modeling which is typically performed off-line, it has several associated problems. For instance, it is not obvious how to choose appropriate numbers for the lags n_y and n_u, and problems of dimensionality arise if multivariable systems are considered because past versions of *all* the inputs and outputs would have to be included in the vector x.

Continous-time state-space models

RBF networks can also be used in the context of state-space modeling. Consider the following nonlinear control-affine model

$$\dot{x} = f(x) + g(x)u \tag{3}$$

and assume that all state variables are available for measurement. Then the system identification problem reduces to approximating the unknown functions $f(x)$ and $g(x)$ by RBF networks. Instead of developing a single RBF model as in the NARMAX approach, here the individual components of f and g are approximated by RBF networks. Both the number of approximation problems to be solved and the dimensionality of these problems are determined by the number of state variables. The dimensionality of the approximation problems is therefore independent of the number of inputs.

Continuous-time state-space models have important advantages over both discrete-time state-space models and discrete-time input/output (NARMAX) models:

1. *A priori* information, which is often available from simple mass and energy balances, can be easily incorporated into continuous state-space models as structural information. A particular structure of a continuous state-space system which may simplify the identification problem generally does not translate into a similarly simple structure of a corresponding discrete-time state-space model. For instance, the control-affine model structure in Equation (3) usually is not retained by a discrete-time state-space model[6].
2. By using continuous-time state-space models, the system identification task is reduced to several approximation problems of smaller dimension (which is equal to the number of state variables in the control-affine case). By contrast, NARMAX models typically require several past values of inputs and outputs to be included in the model, thus calling for the solution of a single, higher dimensional approximation problem.

In this paper we propose an approach to RBF modeling and control that is based on networks of *compactly supported* basis functions; i.e. the RBF is not identically zero only within a limited distance from its associated center, as opposed to RBFs with global support such as the Gaussian. Networks based on such RBFs can provide advantages attributable to the compact support of the basis functions, such as more efficient on-line adaptation. The continuous-time state-space model is obtained in a recursive fashion by generating the RBF centers c_i on-line, and continuously adjusting the weights a_i corresponding to RBFs that are activated at a particular time. Since both the center generation and weight adaptation affect the model only locally, the modeling scheme is suitable for adaptive control applications. The control strategy is based on input-output linearization of the RBF approximation.

Compactly supported radial basis functions

RBF expansions of the form in Equation (1) can be used to construct interpolating or approximating functions for a given set of data points. If interpolation is considered, N corresponds to the number of data points and the RBF centers c_i coincide with the data points whereas, for approximation problems, expansions consisting of fewer RBFs than data points are used. For RBF techniques to be applicable in practice, the associated interpolation/approximation problems must have unique solutions. Considering for the moment the problem of constructing an interpolating function through a set of distinct data points $\{x_1, ..., x_N\}$, it is easy to verify that this problem is uniquely solvable if the matrix A with

$$A_{jk} = \phi(\| x_j - x_k \|); \quad j, k = 1, ..., N \tag{4}$$

is non-singular. The RBF coefficients a are then given by

$$a = A^{-1} f \tag{5}$$

where $f = [f_1 f_2 ... f_N]^T$ are the functional values at the N data points. Several results exist that characterize classes of basis functions which result in the matrix A being invertible for any given set of data points. To discuss these results we need to introduce the following definitions:

Definition 1. A function $\phi(r)$ is said to be *completely monotonic* on $(0, \infty)$ provided that it is in $C^\infty(0, \infty)$ and $(-1)^l \phi^{(l)}(r) \geq 0, r \in (0, \infty), l = 1, 2, ...$

Definition 2. A radial function $\phi(r)$ is called *positive definite* on R^n (or $\phi \in \mathrm{PD}_n$) if, for any set $X = \{x_1, ..., x_N\}$ of N distinct data points in R^n, the quadratic form

$$\lambda^T A \lambda = \sum_{j=1}^{N} \sum_{k=1}^{N} \lambda_j \lambda_k \phi(\| x_j - x_k \|), \qquad \lambda = [\lambda_1 \lambda_2 ... \lambda_N] \in R^n \tag{6}$$

is positive definite[7].

Definition 3. A scalar function $\phi(r)$ has compact support if it is zero outside a compact subset of R.

The following two conditions are sufficient to provide unique solutions of the interpolation problem.

1. If the first derivative of the function ϕ is completely monotonic but not constant on $(0, \infty)$ then the matrix A can be inverted[7]. Note that this result does not cover functions with compact support.
2. If the radial basis function ϕ is positive definite then the matrix A is positive definite, and hence A is invertible[8]. The class of positive definite functions is closely related to the class of completely monotonic functions, because it has been shown[7] that univariate completely monotonic functions $F(\cdot)$ yield radial functions $\phi(\| r \|) = F(r^2)$ that are

PD_n for all n; i.e. the interpolation/approximation problem is uniquely solvable in all dimensions. The converse has also been shown[9]. These results imply that there are no functions ϕ of compact support that are PD_n for all n, because completely monotonic functions cannot have any zeros.

Note that these conditions are sufficient but not necessary so other functions may exist that can be used as basis functions for RBF interpolation. The above results also apply to approximation problems in which fewer RBFs than data points are used. Basis functions that satisfy one of the above conditions will also guarantee unique solutions of the approximation problem; i.e. the existence of the pseudo-inverse corresponding to Equation (4) is assured if the RBF centers are selected to be a subset of the data points[8].

There exist many radial basis functions that satisfy at least one of the above conditions, including the Gaussian and multiquadric functions introduced above. However, none of these functions have compact support. While the first condition cannot be applied to functions with compact support, the second one can. In fact, there exist RBFs with *compact* support that are positive definite for a *restricted range* of n. Several RBFs within this class have been characterized recently by Wu[9]. For instance, the 'cut-off' polynomial

$$\phi(r) = \begin{cases} (1-r)^4(4+16r+12r^2+3r^3) & \forall r\in[0,1] \\ 0 & \text{elsewhere} \end{cases} \quad (7)$$

is positive definite $\forall n \leq 3$, and the corresponding approximating function $F \in C^2$. More generally, for any given dimension n and prescribed smoothness C^i there exists a locally supported RBF that is a 'cut-off' polynomial of a form similar to Equation (7), which is also in $PD_k \forall k \leq n$. Examples of such functions are given in *Table 1* and compared graphically in *Figure 1*. RBF, Equation (7), appears to be most suitable for the applications in this paper, as it possesses both a sufficient degree of smoothness and order of positive definiteness. In *Figure 2*, this particular basis function is compared to a Gaussian function, where both functions are centered at the origin and scaled appropriately.

Although these functions are indeed very similar, RBFs with compact support may provide certain advantages over RBFs having global support:

1. On-line adaptation is computationally simpler because only a subset of basis functions has to be adjusted continuously.

Table 1 Examples of compactly supported radial basis functions on [0,1]

	Basis function	Smoothness	Positive definiteness
ϕ_1	$(1-r)$	C^0	PD_1
ϕ_2	$(1-r)^2(2+r)$	C^0	PD_3
ϕ_3	$(1-r)^3(8+9r+3r^2)$	C^0	PD_5
ϕ_4	$(1-r)^4(4+16r+12r^2+3r^3)$	C^2	PD_3
ϕ_5	$(1-r)^5(8+40r+48r^2+25r^3+5r^4)$	C^2	PD_5
ϕ_6	$(1-r)^6(6+36r+82r^2+72r^3+30r^4+5r^5)$	C^4	PD_3

2. If the process shifts to a new operating regime, knowledge about the dynamics at previous operating points will not be lost because updating affects the model strictly locally.

An alternative approach to generating these desirable properties is to 'cut off' global RBFs at large distances from the center, where they contribute little to the overall model output. However, the resulting RBF approximation is discontinuous even though the system to be modelled may be smooth. This makes model inversion, which is required for several model-based control approaches, infeasible.

Model identification and adaptation

Consider the problem of generating on-line an RBF approximation of a continuous-time system

$$\dot{x} = f(x) + g(x)u \quad (8)$$

$$y = h(x) \quad (9)$$

Many chemical processes can be described by models of this control-affine form. Moreover, the process output y

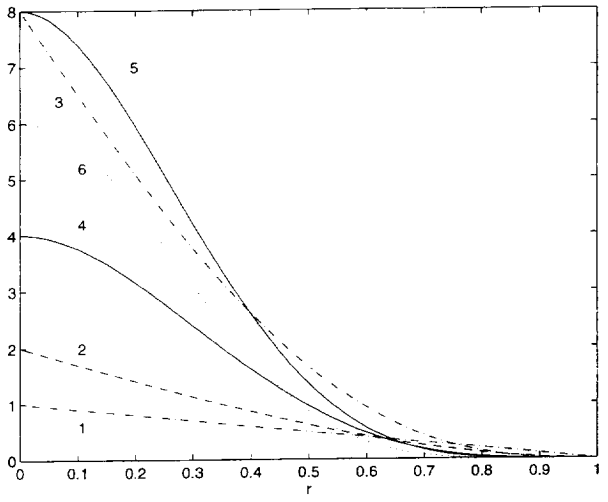

Figure 1 Compactly supported radial functions $\phi_1 - \phi_6$ in *Table 1*

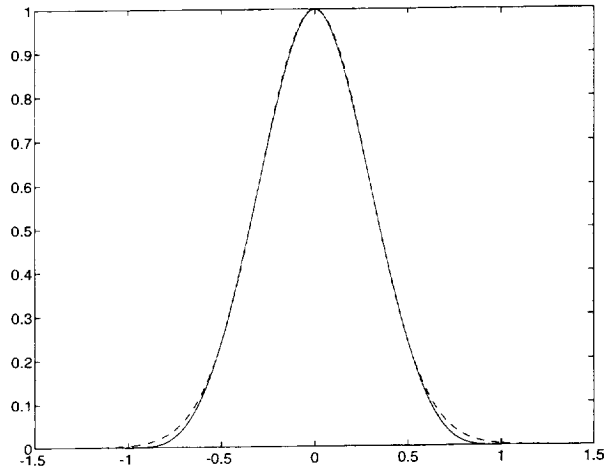

Figure 2 Compactly supported (—) and Gaussian (- - -) basis functions

often coincides with one of the state variables, or it is a known function of the state variables. Thus, we assume that $h(x)$ is known and the system identification problem reduces to approximating the unknown functions $f(x)$ and $g(x)$ by RBF networks.

We consider bounded-input, bounded-state (BIBS) stable systems and assume that the state vector is either completely measured, or that there is enough *a priori* information available to construct suitable approximations of the state equations associated with the unmeasured variables. Introducing a separation of state variables $x^T = [x_m^T\ x_u^T]$, where x_m and x_u are vectors of measured and unmeasured state variables, respectively, Equation (8) can be written as

$$\begin{bmatrix} \dot{x}_m \\ \dot{x}_u \end{bmatrix} = \begin{bmatrix} f_m(x) \\ f_u(x) \end{bmatrix} + \begin{bmatrix} g_m(x) \\ g_u(x) \end{bmatrix} u \tag{10}$$

Next, approximate model equations are specified for the unmeasured state variables; i.e. f_u and g_u are chosen *a priori* to yield

$$\dot{\hat{x}}_u = \hat{f}_u(x_m, \hat{x}_u) + \hat{g}_u(x_m, \hat{x}_u)u \tag{11}$$

The subsystem in Equation (11) is also assumed to be BIBS stable. The state equations associated with the measured state variables are modeled by approximating the functions f_m and g_m with RBF networks

$$\hat{f}_m = W_1\Phi_1(x_m, \hat{x}u); \quad \hat{g}_m = W_2\Phi_2(x_m, \hat{x}_u) \tag{12}$$

where W_1 and W_2 represent weight matrices, and Φ_1 and Φ_2 are vectors of RBFs centered at different locations in state space. Note that the two subsystems are modeled using measured state information x_m and *a priori* estimates of the unmeasured state variables $\hat{x}_u$.

An alternative approach would be to use RBF expansions to model f_u and g_u, and then perform simultaneous state and parameter estimation. Although this approach does not require specifying parts of the model *a priori*, it did not prove to be practically viable and stability of such a scheme would be difficult to guarantee. From this perspective, the assumptions made above about *a priori* modeling represent limitations in terms of the classes of problems that can be addressed by the proposed approach.

Model structure simplification

Despite its limitations, the model form in Equations (9) and (10) may be unnecessarily general. Note that every component of both f and g can depend on the entire set of state variables. This model may be used if there is no additional *a priori* process knowledge available. Fortunately, in most practical cases, two types of process knowledge can be exploited to simplify the model structure:

1. Structural information; i.e. knowledge of which state variables depend on which other state variables. In most cases, this implies that the problem of learning $2n$ multivariate functions that depend on n variables is reduced to a potentially larger number of lower-dimensional learning problems.
2. Certain terms of the state equations may be completely or partially known, as they directly follow from energy and/or mass balances around the processing unit.

In general, the resulting model can contain several RBF approximations of different dimensionality, as well as known terms, potentially with unknown coefficients. However, the entire model has to be linearly parameterized to allow efficient parameter update laws to be applied.

Example: CSTR. Consider a constant volume CSTR in which reactions occur between N independent species. For species i, the mass balance can be written as

$$\dot{c}_i = \frac{Q}{V}(c_{i0} - c_i) + r(c_1, ..., c_N, \theta), \quad i = 1, ..., N \tag{13}$$

where c_i is the concentration of species i, c_{i0} is the inlet concentration of species i, Q is the flowrate, V is the reactor volume, θ is the reactor temperature, and r is the reaction rate. Without making any assumptions about the reaction scheme, the first term in this equation can be assumed known (with c_{i0} as a potentially unknown parameter). Similarly, the energy balance around the reactor can be written as

$$\dot{\theta} = \frac{Q}{V}(\theta_0 - \theta) + h(c_1, ..., c_N, \theta) + (\theta_j - \theta)c \tag{14}$$

where θ_j is the coolant jacket temperature, θ_0 is the feed temperature, c is proportional to the heat transfer coefficient, and the term h comprises all energy contributions due to the reactions. The first term in this equation can be assumed known (with θ_0 as a potentially unknown parameter). Also, the temperature difference $\theta_j - \theta$ is known to be multiplied by a constant.

If θ_j is considered the manipulated input to the system, knowledge of the model structure in Equations (13) and (14) greatly simplifies the construction of g. In fact, the problem of estimating g is reduced to estimating a single constant (c) in one state equation. By contrast, if no process knowledge was available, the general model form in Equation (8) would call for the identification of $(N+1)$ multidimensional functions g_i. If the reaction scheme was known, even more structure could be built into the model and consequently also simplify the problem of estimating f.

Parameter estimation

The parameter adaptation for the RBF coefficients W_1 and W_2 is performed using an *error filtering method*[3], whose block diagrammatic representation is shown in

Figure 3. With the definition of the estimation error $e_m = \hat{x}_m - x_m$, an estimate of the measured state variables is generated by

$$\dot{\hat{x}}_m = -De_m + W_1\Phi_1(x_m, \hat{x}_u) + W_2\Phi_2(x_m, \hat{x}_u)u \quad (15)$$

where D is any positive definite matrix, which for simplicity is chosen to be a diagonal matrix with identical positive elements d.

In order to avoid parameter drift due to modeling errors, a continuous deadzone signal is introduced as[10],

$$e_\Delta \equiv e_m - \alpha\sigma(e_m) \quad (16)$$

where α represents the size of the deadzone and $\sigma(e_m)$ is the vector saturation function with

$$\sigma_i(e_m) = \text{sat}(e_{mi}/\alpha) \quad (17)$$

The following parameter update laws are used

$$\begin{aligned} \dot{W}_1 &= -\gamma_1 e_\Delta \Phi_1(x_m, \hat{x}_u) \\ \dot{W}_2 &= -\gamma_2 e_\Delta \Phi_2(x_m, \hat{x}_u)u \end{aligned} \quad (18)$$

where γ_1 and γ_2 are adaptation gains. A stability proof for this parameter adaptation scheme, as well as conditions for asymptotic convergence of the estimation error, are provided by Sanner and Slotine[4] for the case of complete state measurements. It is straightforward to extend the proof to the partial state measurement case, where the unmeasured state variables are replaced by estimates generated by a stable system. Instead of the deadzone approach, a projection algorithm can be used[3] and stability of the identification algorithm is still ensured. Note that in the present case, the best achievable approximation of the dynamics associated with x_m is limited by the *a priori* choice of state equations that provide estimates of the unmeasured state variables x_u.

RBF center generation

Thus far, it has been assumed that the RBF center positions are known. In practice, however, appropriate locations for the *actual* centers are not known *a priori*. We assume that *potential* centers are located on a regular grid in the state space of Equation (8). A new RBF center c_i (corresponding to the grid point that is closest to the current state) is activated if the estimation error exceeds a threshold, as specified by the size of the deadzone α. Thus, RBF centers are introduced only in regions of the state space that are actually encountered during operation and where significant modeling errors occur. This strategy allows the number of RBFs in the network to remain small. The mesh size of the regular grid can be determined based on *a priori* estimates of the smoothness of the functions $f(x)$ and $g(x)$[4].

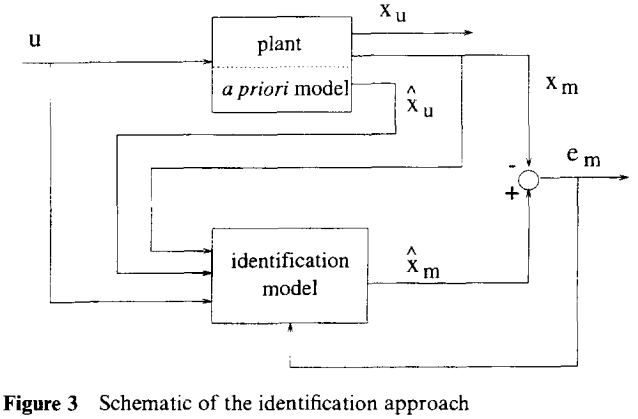

Figure 3 Schematic of the identification approach

It has to be mentioned that a regular grid (with constant mesh size) may not be the best choice for center locations because it can lead to an unnecessarily large number of RBFs. Alternative approaches that determine local grid sizes on-line, along with local RBF shape parameters that correspond to certain RBF 'widths', are currently under investigation[4]. These techniques are based on estimates of the spatial bandwidth of the function to be approximated. Note that, even for a constant width, the arrangement of RBFs on a regular grid may not be optimal in the sense that it may not lead to the best function approximation for a given number of basis functions. Clustering techniques have been proposed[11] that may provide improvements over the regular grid configuration, but they are not readily applicable to the on-line identification and parameter estimation problem considered here. Changing the RBF center locations on-line (via clustering) would certainly complicate the parameter estimation for the RBF weights. For this reason we adopted the approach of placing RBF centers on a regular grid.

Controller design

The proposed controller design strategy is based on the input-output linearization approach[12]. First, the design procedure is presented for the case where the nonlinear functions $f(x)$ and $g(x)$ in Equation (8) are known and the entire state vector x is measured. Then we discuss how the RBF model can be used to extend the design technique to processes in which $f(x)$ and $g(x)$ are unknown and some state variables may be unmeasured.

The input-output linearizing control law is most simply represented using Lie derivative notation[12,13]. The Lie derivative of the scalar function $h(x)$ with respect to the vector field $f(x)$ is defined as

$$L_f h(x) \equiv \frac{\partial h(x)}{\partial x} f(x) \quad (19)$$

Higher-order Lie derivatives are defined recursively

$$L_f^k h(x) \equiv \frac{\partial L_f^{k-1} h(x)}{\partial x} f(x) \quad (20)$$

The nonlinear system, Equations (8) and (9), has relative degree r at the point x_0 if:

1. $L_g L_f^k h(x) = 0$ for all x in a neighborhood of x_0 and all $k < r - 1$
2. $L_g L_f^{r-1} h(x_0) \neq 0$

Note that the relative degree is not well defined at points x_0 where $L_g L_f^{r-1} h(x_0) = 0$. By definition of the relative degree, the first r derivatives of the plant output can be written as

$$y^{(k)} = L_f^k h(x), \quad 0 \leq k \leq r - 1 \tag{21}$$

$$y^{(r)} = L_f^r h(x) + L_g L_f^{r-1} h(x) u \tag{22}$$

where the function $L_g L_f^{r-1} h(x)$ is non-zero.

The objective is to make the plant output track the output of a linear reference model[12]

$$\dot{z}_m = A_m z_m + b_m y_{sp} \tag{23}$$

$$y_m = c_m z_m \tag{24}$$

where y_{sp} is the setpoint, z_m is a q-dimensional state vector, y_m is the model output, and A_m is a stable matrix. The reference model is chosen to have relative degree r and unity steady-state gain. Thus, the first r derivatives of the model output are

$$y_m^{(k)} = c_m A_m^k z_m, \quad 0 \leq k \leq r - 1 \tag{25}$$

$$y_m^{(r)} = c_m A_m^r z_m + c_m A_m^{r-1} b_m y_{sp} \tag{26}$$

The input-output linearizing control law is designed such that the tracking error $e \equiv y_m - y$ has the following dynamics:

$$e^{(r)} + \alpha_r e^{(r-1)} + \ldots + \alpha_2 \dot{e} + \alpha_1 e = 0 \tag{27}$$

where the controller parameters α_k are chosen such that $s^r + \alpha_r s^{r-1} + \ldots + \alpha_2 s + \alpha_1$ is a Hurwitz polynomial. The static state feedback control law that yields the desired error dynamics is[12]

$$u = \frac{-L_f^r h(x) + c_m A_m^r z_m + c_m A_m^{r-1} b_m y_{sp}}{L_g L_f^{r-1} h(x)} + \frac{\sum_{k=1}^{r} \alpha_k [c_m A_m^{k-1} z_m - L_f^{k-1} h(x)]}{L_g L_f^{r-1} h(x)} \tag{28}$$

In the absence of plant/model mismatch, the linearizing control law, Equation (28), has desirable tracking properties. These results follow immediately from Equation (27)

1. *Perfect tracking*: If $y^{(k)}(0) = y_m^{(k)}(0)$ for all $k \in [0, r-1]$, then $y(t) = y_m(t)$ for all $t \geq 0$.
2. *Asymptotic tracking*: If $y^{(k)}(0) \neq y_m^{(k)}(0)$ for some $k \in [0, r-1]$, then $\lim_{t \to \infty} [y(t) - y_m(t)] = 0$.

However, the control law, Equation (28), generally will not provide offset-free tracking if there is modeling error. It is important to note that offset also may occur in the adaptive case because the parameter estimator, Equation (18) is driven by the estimation error ($e_m = \hat{x}_m - x_m$) rather than the tracking error ($e = y_m - y$). An integral controller can be obtained via a simple modification of Equation (28)

$$u = \frac{-L_f^r h(x) + c_m A_m^r z_m + c_m A_m^{r-1} b_m y_{sp}}{L_g L_f^{r-1} h(x)} + \frac{\sum_{k=1}^{r} \alpha_k [c_m A_m^{k-1} z_m - L_f^{k-1} h(x)]}{L_g L_f^{r-1} h(x)} + \frac{\alpha_0 \int_0^t [c_m z_m - h(x)]}{L_g L_f^{r-1} h(x)} \tag{29}$$

In this case, the controller parameters α_k are chosen such that the following polynomial is Hurwitz: $s^{r+1} + \alpha_r s^r + \ldots + \alpha_1 s + \alpha_0$.

The linearizing control law, Equation (28) (and its integral version), ensure asymptotic tracking of the output if there is no plant/model mismatch. As we now illustrate, an additional assumption is required to ensure that the closed-loop system is internally stable if $r < n$. Under the linearizing control law, Equation (28), there exists a nonlinear coordinate transformation $[\xi^T, \eta^T] = \Phi^T(x)$ such that the nonlinear system, Equations (8) and (9) has the following normal form representation[12]

$$\begin{aligned} \dot{\xi} &= A_l \xi + B_l v \\ \dot{\eta} &= q(\xi, \eta) \\ y &= C_l \xi \end{aligned} \tag{30}$$

where

$$v = c_m A_m^r z_m + c_m A_m^{r-1} b_m y_{sp} + \sum_{k=1}^{r} \alpha_k [c_m A_m^{k-1} z_m - \xi_k] \tag{31}$$

and the triplet (A_l, B_l, C_l) is in Brunovsky canonical form. The zero dynamics are defined as dynamics of the $(n - r)$-dimensional nonlinear subsystem when the variables of the r-dimensional linear subsystem $\xi = 0$

$$\dot{\eta} = q(0, \eta) \tag{32}$$

Stability of the zero dynamics is a necessary and sufficient condition for the control law, Equation (28) to yield local closed-loop stability[12]. Therefore, the proposed control strategy is restricted to nonlinear systems with stable zero dynamics.

Now we consider the case where the nonlinear functions $f(x)$ and $g(x)$ are unknown and some state variables may be unmeasured. A natural way to address this

problem is to use estimated functions and state variables from the RBF model in the linearizing control law. That is, the control law is constructed by replacing the unknown functions with their corresponding estimates

$$f(x) \rightarrow \begin{bmatrix} W_1\Phi_1(x_m, \hat{x}_u) \\ \hat{f}_u(x_m, \hat{x}_u) \end{bmatrix} \tag{33}$$

$$g(x) \rightarrow \begin{bmatrix} W_2\Phi_2(x_m, \hat{x}_u) \\ \hat{g}_u(x_m, \hat{x}_u) \end{bmatrix} \tag{34}$$

Thus, the control law Equation (28) can be rewritten as

$$u = \frac{-L_{\hat{f}}^r h(x_m, \hat{x}_u) + c_m A_m^r z_m + c_m A_m^{r-1} b_m y_{sp}}{L_{\hat{g}} L_{\hat{f}}^{r-1} h(x_m, \hat{x}_u)} + \frac{\sum_{k=1}^{r} \alpha_k [c_m A_m^{k-1} z_m - L_{\hat{f}}^{k-1} h(x_m \hat{x}_u)]}{L_{\hat{g}} L_{\hat{f}}^{r-1} h(x_m, \hat{x}_u)} \tag{35}$$

We have assumed that the RBF model has the same relative degree as the nonlinear system. In many applications, this assumption can be satisfied by incorporating *a priori* knowledge into the RBF model. However, it is important to note that the control law, Equation (28) can be constructed and implemented even if the relative degrees are not equal.

The proposed approach can be viewed as an extension of the certainty equivalence principle which is commonly used in conjunction with state[14] and parameter[15] estimation. In the present case, entire functions are replaced by their estimates. This approach also has been pursued by other authors[4,16–18] within the context of nonlinear adaptive control with artificial neural networks. A natural question that arises from the use of the certainty equivalence approach is the possible existence of a separation principle. For linear systems, it is well known that the combination of a stabilizing state feedback controller and a stable state observer yields a stabilizing output feedback controller. By contrast, the separation principle does not necessarily hold when a stable controller and a stable parameter estimator are combined because the resulting control problem is nonlinear. The same problem is encountered in the proposed strategy; closed-loop stability is not implied by stability of the individual elements. It is possible to ensure closed-loop stability by placing restrictive assumptions on the nonlinear system[4,18]. We have chosen not to invoke such assumptions in order to develop a control strategy that is applicable to a much larger class of nonlinear systems. In place of theoretical guarantees, the stability and performance of the proposed technique is evaluated via simulation.

A potential difficulty in the implementation of the adaptive controller is the existence of singular points where $L_{\hat{g}} L_{\hat{f}}^{r-1} h(x) = 0$. At such points the linearizing control law, Equation (35) is not well defined. While singularities are not unique to our technique, the problem is more difficult to handle due to on-line parameter adaptation. As an illustration, assume that the nonlinear system has relative degree $r = 1$. Then the following function appears in the denominator of the control law, Equation (35)

$$L_{\hat{g}} L_{\hat{f}}^{r-1} h(x) = \frac{\partial h(x)}{\partial x} W_2\Phi(x) \tag{36}$$

Thus, the control law is not well defined at points where $W_2\Phi(x) = 0$. Because the weight matrix W_2 is time varying it is very difficult to determine the location of singular points *a priori*. In the relative degree one case, the singularity problem can be avoided by restricting all the weights to have the same sign[3]. This approach seems likely to degrade the approximation capabilities of the network. Moreover, it is difficult to extend this idea to nonlinear systems of higher relative degree. We recommend the addition of a perturbation function[19] to the denominator of the control law to ensure that the input remains bounded if a singular point is encountered.

Another potential problem is that the adaptive controller may yield poor transient performance when the nonlinear system moves into regions of the state space that have not been encountered previously. This is especially likely during the early phase of on-line operation as a result of poor initial estimates of the RBF weights. One way to alleviate this problem is to perform open-loop identification prior to controller implementation. This approach is feasible only if the system is stable or has been stabilized by feedback. Several authors[3,4] have proposed sliding mode control as a means of rapidly forcing the system back into regions where the model is sufficiently accurate for control. While sliding mode controller design is not based on an explicit nonlinear model, it does require bounds on the unknown functions and may lead to large control moves that are unacceptable in chemical process applications. As an alternative, we allow the nonlinear controller to continue operating even when a new region of the state space is encountered. New RBFs are introduced as needed, and the RBF weights are adapted to improve transient performance. The simulation results presented in the next section demonstrate the effectiveness of this approach.

Simulation study

Consider a constant volume CSTR in which a first-order exothermic reaction occurs. The state equations in dimensionless form are given by Uppal *et al.*[20]

$$\dot{x}_{1p} = -x_{1p} + D_a(1 - x_{1p})e^{\frac{x_{2p}}{1+x_{2p}/\varphi}} \tag{37}$$

$$\dot{x}_{2p} = -x_{2p} + BD_a(1 - x_{1p})e^{\frac{x_{2p}}{1+x_{2p}/\varphi}} + C(u - x_{2p}) \tag{38}$$

where x_{1p} and x_{2p} are the conversion and the dimensionless reactor temperature, respectively, and u is the dimensionless cooling jacket temperature. The nominal system parameters are chosen as $D_a = 0.072$, $\varphi = 20.0$, $B = 8.0$, and $C = 0.3$. For this choice of parameters, the CSTR exhibits 'S-shaped' steady-state characteristics between the input u and both state variables; i.e. output multiplicities occur. The initial conditions which correspond to a stable steady-state are

$$u^0 = 0; \quad x_1^0 = 0.144; \quad x_2^0 = 0.089 \tag{39}$$

To facilitate the RBF modeling procedure, the process state variables are scaled to assure that both state variables are of the same order of magnitude. The scaling is chosen as

$$x_1 = x_{1p}, \quad x_2 = 0.1x_{2p} \tag{40}$$

Open-loop simulations

Before the RBF identification technique can be applied, an appropriate model structure has to be chosen. As discussed under Model Structure Simplification, as much *a priori* information as possible should be used in order to simplify the model structure and/or reduce the dimensionality of the resulting RBF approximation problems.

Although the state equations are assumed unknown for identification purposes, the two model equations will correspond to a mass balance and an energy balance. Even without explicit knowledge of the mass balance, it is apparent that the input u will not have a direct effect on reactor conversion and therefore does not appear in the mass balance. Furthermore, we choose a control-affine structure for the energy balance where the input u is multiplied by a constant (which would correspond to a scaled heat transfer coefficient). We assume these facts to be known, as they constitute very general types of *a priori* knowledge and do not require an explicit process model. However, we do not assume any knowledge about the flowrates, reactor volume, inlet concentration, or reaction kinetics, although such information could be used to further simplify the modeling problem (see section on Model Structure Simplification above).

Complete state measurements. First, we consider the case where both x_1 and x_2 are measured and aim at constructing a process model of the form

$$\dot{\hat{x}}_1 = f_1(x_1, x_2) = w_1^T \Phi(x_1, x_2) \tag{41}$$

$$\dot{\hat{x}}_2 = f_2(x_1, x_2) + w_u u = w_2^T \Phi(x_1, x_2) + w_u u \tag{42}$$

The model contains the weight w_u and two 2-dimensional RBF approximations whose centers and weights have to be determined. For the on-line identification scheme discussed in Model Identification and Adaption above, the following parameters are used

$$d = 25; \alpha = 1 \times 10^{-5}; \gamma_1 = 2; \gamma_2 = 6 \tag{43}$$

The basis function, Equation (7), is scaled to have a support of $R = 0.05$, and the regular grid for potential RBF centers has a mesh size of 0.02.

Figure 4 shows the input signal used for identification, as well as the scaled state variables and their estimates. The input signal is constructed by choosing a switching interval and a probability of switching. If a

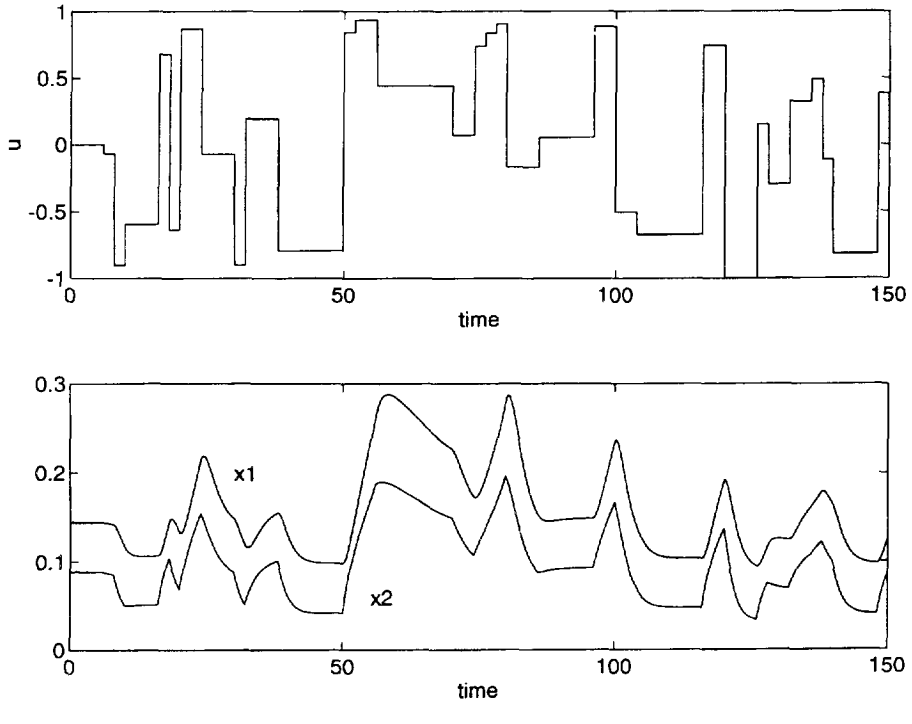

Figure 4 Identification of the RBF model

new input is generated, it is assigned according to a uniform distribution in the interval [−1,1]. Because the state estimates ($\hat{x}$) converge rapidly to the actual values, errors are visible only at the very beginning of the identification process. The RBF center generation mechanism is illustrated in *Figure 5*, which shows the state trajectory and the RBF centers that have been introduced on-line. It is apparent that the data are clustered mainly along a diagonal in the (x_1, x_2) plane. Therefore, only a small fraction of the potential RBF centers on the regular grid are likely to be recruited during normal operation.

To assess whether the model obtained truly captures the plant dynamics, the time courses of the actual and estimated functions $f_1(x)$ and $f_2(x)$ during the initial identification phase are shown in *Figure 6*. Two types of estimates are depicted: (I) instantaneous estimates which are based on the RBF model that is continuously updated during the identification process; and (II) estimates obtained from the final model at $t = 50$. The instantaneous estimates slowly track the true function values, but the estimates obtained from the final model approximate the actual process nonlinearities reasonably well.

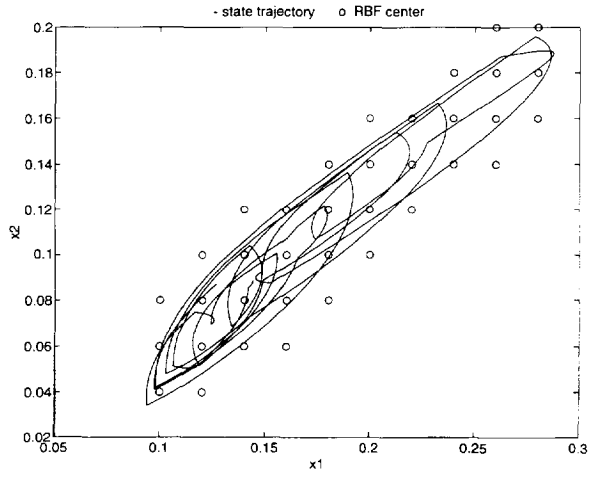

Figure 5 State trajectory and RBF centers

Partial state measurements. Next, we consider the case where only temperature (x_2) measurements are available and an *a priori* approximation of the first state equation will be used to generate estimates of the conversion x_1. The data in *Figure 4* indicate that x_1 approximately is a lagged version of x_2. This suggests a simple first order linear relationship as the *a priori* approximation of the first state equation, resulting in the following model

$$\dot{\hat{x}}_1 = -\hat{x}_1 + x_2 \tag{44}$$

$$\dot{\hat{x}}_2 = f_2(\hat{x}_1, x_2) + w_u u = w_2^T \Phi(\hat{x}_1, x_2) + w_u u \tag{45}$$

Observe that the *a priori* approximation $\hat{x}_1$ is now used in the identification model to predict x_2. As shown in *Figure 7*, the RBF model generated from the same process data as before captures the x_2 dynamics quite accurately despite the crude approximation used for the first state equation. This observation is supported by *Figure 8*, which compares the estimated state variable ($\hat{x}_1$) to the true state variable (x_1). The simple linear model, Equation (44), appears to provide an adequate approximation as the two curves show a very good qualitative agreement. Note that the bias between the curves can be handled as the RBF identification simply

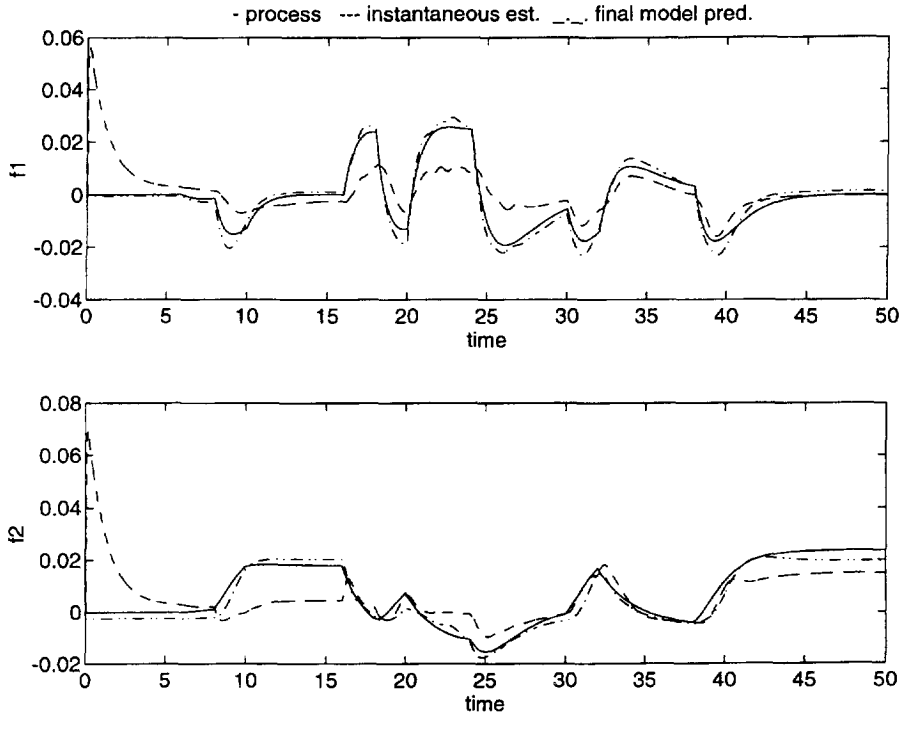

Figure 6 Estimates of the process nonlinearities

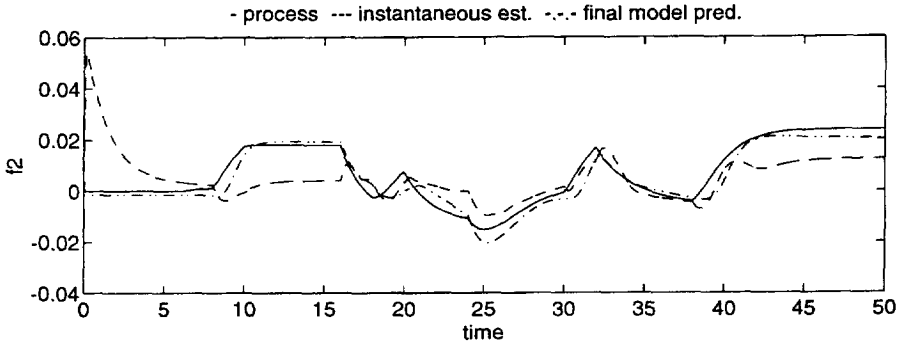

Figure 7 Identification with incomplete state measurements, *a priori* model, Equation (44)

generates a shifted version of the orginal f_2 surface. But clearly, the *a priori* choice of the first state equation has a strong effect on the quality of the resulting RBF model. Severe process/model mismatch can result in poor performance of the identification scheme, as shown next. The model, Equation (44), is modified by increasing the time constant by a factor of 20 and also increasing the gain of the linear approximation such that the resulting state trajectory covers a similar region in state space

$$\dot{\hat{x}}_1 = -0.05\hat{x}_1 + 0.1x_2 \tag{46}$$

For this choice of *a priori* model, the estimated state variable ($\hat{x}_1$) does not follow the trend of the true state variable as closely as before (see *Figure 8*). As a consequence, the corresponding RBF model is not capable of capturing the true process nonlinearities to a similar accuracy (see *Figure 9*). In general, any type of physical process knowledge should be incorporated in the *a priori* choice of state equations associated with unmeasured state variables.

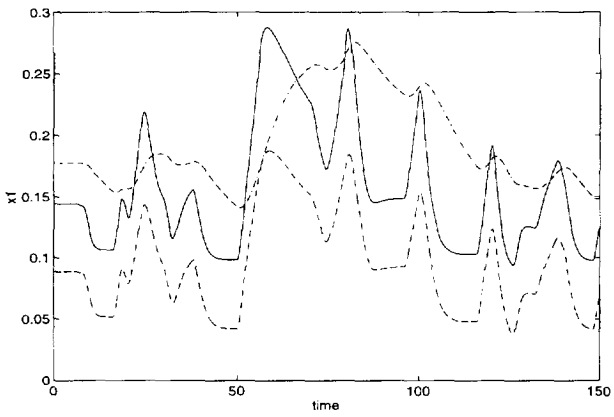

Figure 8 *A priori* approximations of x_1: true (—); Equation (46) (·-·-·); (Equation (44) (- - -)

Closed-loop simulations

The identification scheme based on temperature measurements and the approximate state equation, Equation (44), is combined with the control strategy discussed in Controller Design above. Because it is desired to control temperature, the relative degree is $r = 1$.

Setpoint tracking. The objective for the plant output is to track the output of the following linear reference model

$$\dot{y}_m = -\frac{1}{\tau_r} y_m + \frac{1}{\tau_r} y_{sp} \tag{47}$$

where the time constant is chosen as $\tau_r = 1$. We use the control law, Equation (35), but add explicit integral action as in Equation (29) in order to provide offset-free setpoint tracking. The controller parameters are chosen as follows: $\alpha_0 = 0.04$ and $\alpha_1 = 0.5$. The parameter w_u is initialized as $w_u(t = 0) = 0.1$. We emphasize that there is *no* network training prior to closed-loop operation. *Figure 10* shows the closed-loop performance for repeated setpoint changes between two operating points, one of which corresponds to a stable steady state, while

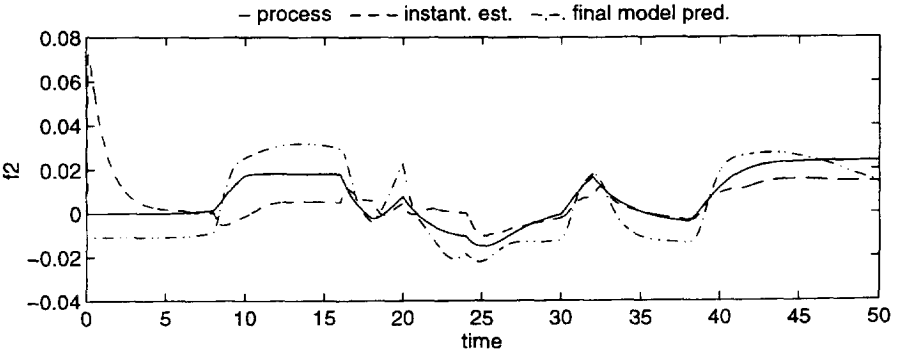

Figure 9 Identification with incomplete state measurements, *a priori* model, Equation (46)

the other one is open-loop unstable. The results in *Figure 10* demonstrate the capability of the RBF model to learn the relevant process dynamics, as the process output tracks the output of the reference model closely after the initial training phase. The initial control move is large, but more conservative changes are produced as the nonlinear dynamics are learned by the RBF model. No singularity problems are encountered during these simulations, partially due to the fact that the initial value of the parameter w_u is chosen to have the correct sign based on physical considerations.

Disturbance rejection. In *Figure 11*, we demonstrate the robustness of the control system to disturbances while operating at an unstable steady state. First, step disturbances in the feed flow rate are implemented by changing D_a from the nominal value of 0.072 to 0.045 at $t = 2$, and then returning it to 0.072 at $t = 12$. The

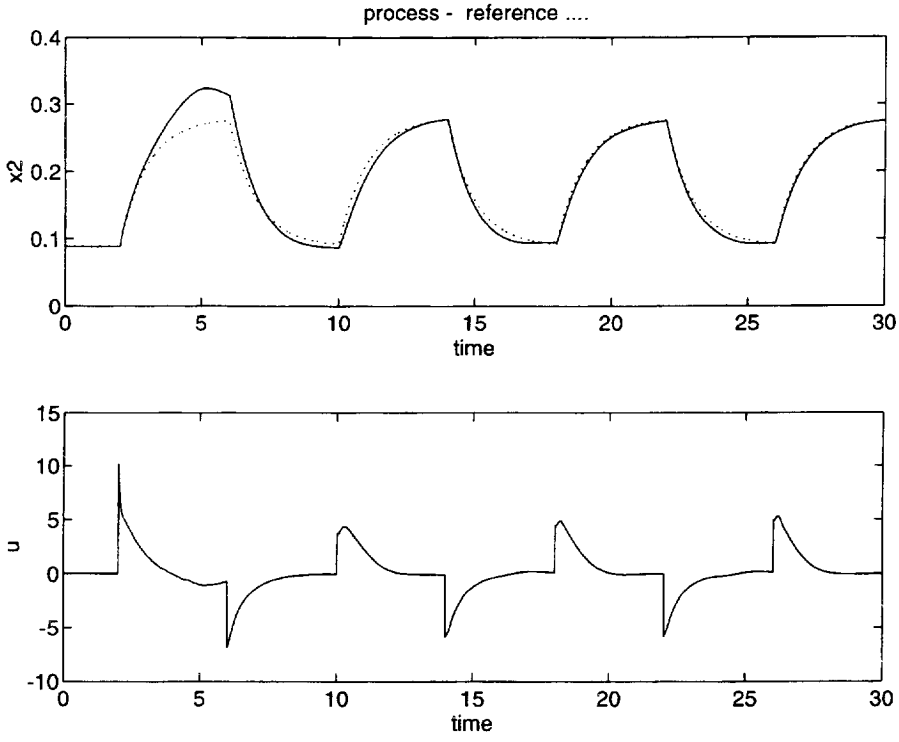

Figure 10 Setpoint tracking

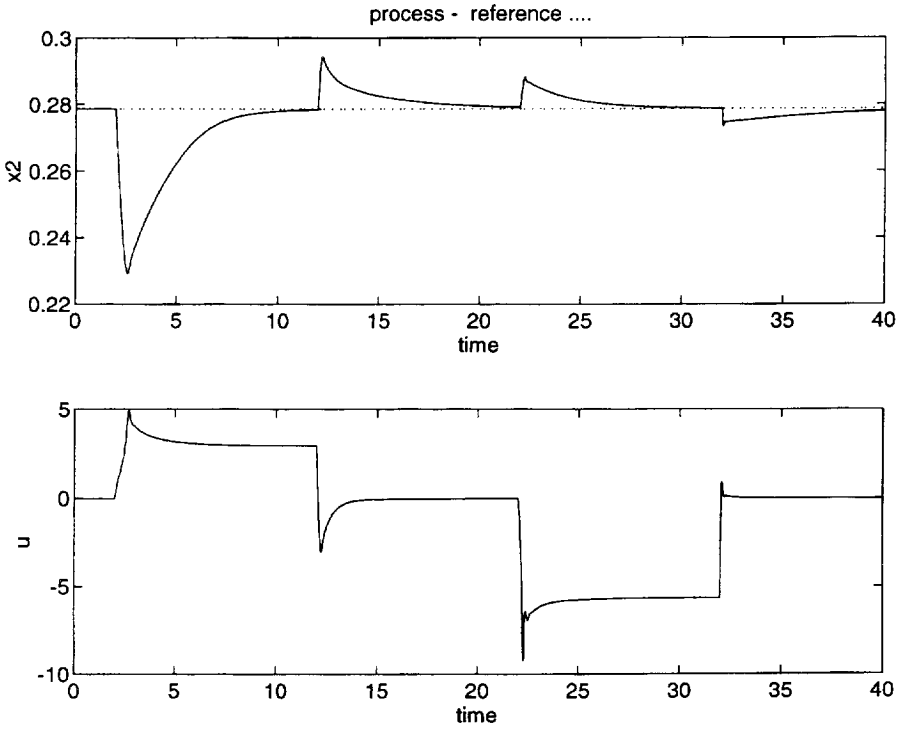

Figure 11 Disturbance rejection

second set of disturbances consists of step changes in the heat transfer coefficient, which are realized by changing C from its nominal value of 0.3 to 0.1 at $t = 22$, and back to 0.3 at $t = 32$. Good performance is obtained, and offset is removed due to the addition of integral action in the control law.

Conclusions

We have presented a nonlinear identification and adaptive control approach based on compactly supported radial basis functions and demonstrated its effectiveness using a second-order chemical reactor model. Due to their compact support, only a small number of basis functions have to be adapted at any time, and new information affects the model only locally. Any unmeasured state variables are estimated by choosing an *a priori* dynamic model for these variables. The proposed state-space modeling approach offers the possibility of incorporating structural information *a priori*, which can simplify the identification problem significantly. Despite these advantages, it should be kept in mind that this approach is restricted to problems of reasonably small dimensionality. In practice, high-dimensional systems may be handled only if they can be reduced to several approximation problems of lower dimension.

Finally, we note that learning techniques using networks of local basis functions can also be applied to a variety of problems where nonlinear relationships of relatively small dimension have to be learned on-line, or where time-varying nonlinear characteristics have to be tracked continuously. Potential applications include block-oriented modeling approaches, which often call for the identification of low-dimensional nonlinearities, and the estimation of time-varying, steady-state nonlinearities, such as titrations curves in pH neutralization systems.

Acknowledgement

MAH would like to acknowledge the National Science Foundation (CTS-9501368) for support of this research.

References

1. Pearson, R. K., Nonlinear input/output modeling, pp. 1–15. *Proc. ADCHEM '94*, Kyoto, Japan, 1994.
2. Chen, S., Billings, S. A., Cowan, C. F. N. and Grant, P. M., Nonlinear systems identification using radial basis functions. *Int. J. Syst. Sci.*, 1991, **12**, 2513–2539.
3. Polycarpou, M. M. and Ioannou, P. A., *Identification and control of nonlinear systems using neural network models: Design and stability analysis*. USC-EE Report, 1991.
4. Sanner, R. M. and Slotine, J.-J. E., Gaussian networks for direct adaptive control. *IEEE Trans. Neural Networks*, 1992, **3**, 837–863.
5. Pottmann, M. and Seborg, D. E., Identification of nonlinear processes using reciprocal multiquadric functions. *Journal of Process Control*, 1992, **2**, 189–203.
6. Grizzle, J. W. and Kokotovic, P. V., Feedback linearization of sampled-data systems. *IEEE Trans. Autom. Control*, 1988, **AC-33**, 857–859.
7. Micchelli, C. A., Interpolation of scattered data: Distance matrices and conditionally positive definite functions. *Constructive Approximation*, 1986, **2**, 11–22.
8. Poggio, T. and Girosi, F., A theory of networks for approximation and learning. *Proc. IEEE*, 1990, Vol. 78, pp. 1481–1497.
9. Wu, Z., Multivariate compactly supported positive definite radial functions. Report of the Institut für Numerische und Angewandte Mathematik, Universität Göttingen, 1994.
10. Sanner, R. M. and Slotine, J.-J. E., Stable recursive identification using radial basis function networks. *Proc. American Control Conf.*: 1992, 1829–1833.
11. Leonard, J. A. and Kramer, M. A. Radial basis function networks for classifying process faults. *IEEE Control Systems Mag.*, April 1991, pp. 31–38.
12. Isidori, A., *Nonlinear control systems*. Springer-Verlag, New York, 1989.
13. Henson, M. A. and Seborg, D. E., Critique of exact linearization strategies for process control. *Journal of Process Control*, 1991, **1**, 122–139.
14. Brogan, W. L., *Modern control theory*. Prentice-Hall, New York, 1991.
15. Sastry, S. and Bodson, M., *Adaptive control: stability, covergence, and robustness*. Prentice-Hall, Englewood Cliffs, NJ, 1989.
16. Chen, F.-C. and Khalil, H. K., Adaptive control of a class of nonlinear discrete-time systems using neural networks. *IEEE Trans. Auto. Control*, 1995, **40**, 791–801.
17. Levin, A. U. and Narendra, K. S., Control of nonlinear dynamic systems using neural networks: controllability and stabilization. *IEEE Trans. Neural Networks*, 1993, **4**, 192–206.
18. Polycarpou, M. M., Stable adaptive neural control scheme for nonlinear control. *IEEE Trans. Auto. Control*, 1996, **41**, 447–451.
19. Castillo, B., Output tracking through singular points for a class of nonlinear SISO systems. *Proc. European Control Conf.*, Grenoble, France, 1991.
20. Uppal, A., Ray, W. H. and Poore, A. B., On the dynamic behavior of continuous stirred tank reactors. *Chem. Eng. Sci.*, 1974, **29**, 967–985.