Copyright © IFAC System Identification, Kitakyushu,
Fukuoka, Japan, 1997

NEURAL NETWORKS FOR SYSTEM IDENTIFICATION

Kumpati S. Narendra * **Snehasis Mukhopadhyay** **

* *Center for Systems Science, Yale University*
New Haven, CT 06520
** *Dept. of Computer & Information Science*
Purdue School of Science at Indianapolis, Indianapolis, IN 46202

Abstract: The paper discusses a theoretical framework within which recent results obtained for the identification and control of single-input single-output (SISO) and multi-input multi-output (MIMO) or multivariable systems can be viewed, and describes some problems encountered in the practical implementation of identification and control methods using neural networks.

Keywords: system identification, neural networks, nonlinear systems, multivariable systems, model approximation

1. INTRODUCTION

In 1990, it was first suggested in (Narendra and Parthasarathy, 1990) that identification and control problems using neural networks can be best addressed within the context of systems theory. In the six years that have elapsed since the publication of (Narendra and Parthasarathy, 1990), a significant body of theoretical results has been generated which provides deeper insights into the nature of both identification and control problems. Extensive simulation studies have also been carried out based on these results and have verified the effectiveness of using neural networks for identifying and controlling complex nonlinear systems. The objective of this paper is twofold: the first is to provide a proper framework within which the recent theoretical advances can be viewed, and the second is to discuss some of the practical considerations that arise while implementing neural network based techniques.

System representation and system identification play an important role in engineering in general and automatic control in particular. Mathematical models of systems can be derived either by a detailed study of the physical laws governing the system, or entirely from input-output measurements, or a combination of the two. Since a model is merely an abstraction of reality, depending upon the simplifying assumptions made by the designer, different models of the plant can be obtained. How modeling of a system is to be carried out to match its intended use, how to determine the best model for a given objective, and how a priori knowledge is to be incorporated in the modeling process are all areas of current research.

It is now common knowledge that neural networks can approximate arbitrarily closely continuous maps over compact sets. If neural networks are to be used as identifiers or controllers which are described by difference (or differential) equations, they have to approximate the nonlinear maps that arise in these equations. Hence, establishing the existence of such maps plays a central role in the use of neural networks both for identification and control. Much of the research that is currently in progress in this area deals with the conditions under which different mappings exist for different problem statements.

In recent years conditions have been derived (Leontaritis and Billings, 1985) (Levin and Narendra, 1996) under which NARMA (nonlinear autoregressive moving-average) model is an exact input-output representation of a nonlinear dynamical system in the neighborhood of an equilibrium state. In this context, it has also come to be realized that the notion of "relative degree" (*i.e.*, the delay of information transmission through the system) is an important feature of dynamical systems for both identificaton and control. In section 3, this concept is discussed and an input-output representation of a system with specified relative degree is presented which is particularly suited for system identification. This sets up the stage for considering identification and control problems in multivariable systems.

At the present time, problems are arising with increasing frequency in a wide spetrum of industries, where poorly defined systems with multiple inputs and multiple outputs have to be identified using stored data and controlled accurately. Needless to say, the effectiveness of neural network based identification and control methods will be judged on how effectively they perform in these different multivariable problems. While the questions that arise in the latter case are similar to those that arise in SISO systems, they are substantially more complex due to the coupling that exists between the inputs and the outputs, as well as the different time delays that can exist between them. The identification of nonlinear multivariable systems consequently plays a central role in this paper.

The great advantage of neural networks in systems identification is that they can model systems using input-output data previously stored, as well as that generated while the systems are in operation. The NARMA representation, which forms the basis for much of neural network based system identificaton using input-output data, is discussed in section 3. The implications of such models for the identification and control of multivariable systems are treated in section 4.

As mentioned earlier, the use to which a model is to be put determines to a large extent the system representation that is chosen. Models for long-term prediction as well as models for on-line control are briefly discussed in section 5. The interest of the authors in constructing models of systems is mainly for their control. The NARMA model which is known to be an exact local representation of the nonlinear system is also known to be very difficult to control. In section 6, approximate models are discussed for SISO and MIMO systems which make the control problem substantially simpler.

Nonlinear control theory and linear adaptive control theory provide the basis for the design of identifiers and controllers. Comments are made throughout the paper regarding the different assumptions that have to be made, the choice of the identifier and controller architectures, and the generation of adaptive laws to design practically viable controllers.

2. STATE AND INPUT-OUTPUT REPRESENTATIONS

In this section, we describe the state space representation of dynamical systems as well as other well-known representations for linear systems. The latter, in turn, forms the basis for similiar representations for nonlinear systems. The state vector representation is particularly suited for theoretical developments and the study of controllability, observability, and stability properties. However, it is rarely that one has access to all the state variables and consequently the estimation of such models from empirical data is a difficult problem. Further, in most practical cases, the behavior of the system is observed by an external observer only through its inputs and outputs. This naturally calls for input-output representations.

2.1 *State Representation*

In all the problems discussed in the following sections we assume that the discrete-time system (the plant) considered has a state representation

$$\Sigma: \qquad \begin{aligned} x(k+1) &= f[x(k), u(k)] \\ y(k) &= h[x(k)] \end{aligned} \tag{1}$$

where $x(k) \in R^n$ denotes the state and $u(k), y(k) \in R^m$ denote the input and output vectors respectively (the dimensions of the input and output vectors are assumed to be the same). $f : R^n \times R^m \to R^n$ and $h : R^n \to R^m$ are analytic functions. When $m = 1$, the system Σ has a single-input and single-output (SISO), and when $m \geq 2, \Sigma$ is a multi-input multi-output (MIMO) or a multi-variable system. The following assumptions are made concerning Σ in the discussion that follow:

(i) the system is of finite dimension n which is known. In practical problems, n is generally not specified, but has to be determined from the available input-output data.
(ii) Σ is BIBO stable. While this assumption may be theoretically restrictive, in many industrial applications, this condition is satisfied.
(iii) $f(0,0) = 0$ and $h(0) = 0$ so that $(0,0)$ is an equilibrium state of the system.

The linearization of Σ around $(0,0)$ is described by Σ_L where

$$\Sigma_L: \quad x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k) \tag{2}$$

where the $(n \times n)$ matrix A and the $(n \times m)$ and $(m \times n)$ matrices B and C are defined by

$$\frac{\partial f(x,u)}{\partial x}|_{0,0} = A$$
$$\frac{\partial f(x,u)}{\partial u}|_{0,0} = B$$
$$\frac{\partial h(x)}{\partial x}|_{0} = C \tag{3}$$

Results from linear control theory are central to those derived in this paper for nonlinear systems. This is due to the fact that if Σ_L is controllable, observable, or asymptotically stable, then the nonlinear system Σ is also locally controllable, observable, or asymptotically stable. In view of this, unless otherwise stated, it is assumed that Σ_L is both controllable and observable (and hence, stabilizable).

2.2 *Representation of Linear Systems*

A large literature currently exists for the representation and identification of linear systems. Three equivalent representations that are commonly used for linear systems are: (i) state representation, (ii) impulse response representation, and (iii) input-output representation (ARMA). These are described by equations (4), (5), and (6) respectively.

$$(i) \quad x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k) \tag{4}$$
$$(ii) \quad y(k+1) = \sum_{i=0}^{\infty} H(i)u(k-i) \tag{5}$$
$$(iii) \quad y(k+1) = \sum_{i=0}^{n-1} A_i y(k-i) + \sum_{j=0}^{n-1} B_j u(k-j) \tag{6}$$

where $H(i), A_i$, and B_i are $m \times m$ matrices. Given the state representation of the system, an equivalent impulse response representation (e.g., $H(i) = CA^{i-1}B$) as well as an equivalent input-output representation can be derived. In the former, the output is expressed in terms of all past values of the input, while in the latter it is expressed in terms of the past n values of input and output vector respectively.

Given a dynamical system Σ, our interest is in representing it in a suitable mathematical form (such as those given for linear systems), and using appropriate identification models to identify it from the available data. In equation (1) if f and h are known, the system Σ is completely specified. In such a case, our interest is in determining representations for the nonlinear system corresponding to linear representations (5) and (6). If such nonlinear representations exist, the problem of identification reduces to one of determining the unknown functions from input-output data. If f and h are unknown, estimating them may be the first step in the identification of Σ. When the state variables are accessible, neural networks can be used to obtain maps $\hat{f}$ and $\hat{h}$ which approximate f and h arbitrarily closely.

As stated in the introduction, the primary interest of the authors in system identification is for control purposes. The control problems, in turn, involve both regulation and tracking. The problems that are currently arising in numerous industries correspond to cases where the functions f and h are unknown, the state variables of the system are not accessible, and identification and control of nonlinear systems have to be effected using only input-output data. As stated earlier, our aim in this paper is to propose methods for achieving the latter for nonlinear multivariable systems.

In section 3 the input-output representation of nonlinear systems is outlined and the approach is extended to multivariable systems in section 4. Identification of multivariable systems based on such a representation are described in section 5.

3. THE NARMA MODEL FOR SINGLE-INPUT SINGLE-OUTPUT SYSTEMS

The first step in the identification problem is the choice of a representation for the dynamical system and a suitable model to identify it.

From equation (1) we have,

$$\begin{aligned} y(k) &= h[x(k)] \\ &\triangleq \Psi_1[x(k)] \\ y(k+1) &= h[f(x(k),u(k))] \\ &\triangleq \Psi_2[x(k),u(k)] \\ &\vdots \\ y(k+n-1) &= h \circ f^{n-1}[\cdot,\cdot] \\ &\triangleq \Psi_n[x(k),u(k),\cdots,u(k+n-2)] \end{aligned} \tag{7}$$

where $f^{n-1}[\cdot,\cdot]$ is an $(n-1)$-times iterated composition of f. Denoting the sequence $y(k), y(k+1), \cdots, y(k+n-1)$ by $Y_n(k)$ and the sequence $u(k), u(k+1), \cdots, u(k+n-2)$ by $U_{n-1}(k)$, equation (7) can be expressed as

$$\Psi[x(k), U_{n-1}(k)] = Y_n(k)$$

Using the implicit function theorem if $\frac{\partial \Psi}{\partial x(k)}|_{(0,0)}$ is nonsingular, $x(k)$ can be expressed locally in terms of the *future* values of $u(\cdot)$ and $y(\cdot)$. However, it is known (by the definition of the state) that $y(k+n) = h[x(k+n)]$ depends upon $x(k)$ and the inputs $u(k), u(k+1), \cdots, u(k+n-1)$. This yields the well-known input-output representation for a *single-input single-output (SISO) system* as

$$y(k+1) = \mathcal{F}[\,y(k), \cdots, y(k-n+1), u(k), \cdots, u(k-n+1)] \quad (8)$$

The derivation of the NARMA model discussed above was based on the implicit function theorem and the assumption that Σ_L is observable. Hence, the region of validity of the NARMA model can be quite limited. Attempts to determine globally valid input-output models have been made in the past by Aeyels (Aeyels, 1981), Sontag (Sontag, 1979), and more recently by Levin and Narendra (Levin and Narendra, 1995). Aeyels considered the dynamical system $\dot{x} = f(x)$, $y = h(x)$ and showed that almost any such system will be observable if $2n+1$ measurements of the output are taken. Sontag studied the existence of global input-output models when f and h in equation (1) are polynomial functions. In (Levin and Narendra, 1995), transversality conditions are used to determine conditions similar to those of Aeyels for nonlinear systems of the form (1), *i.e.*,Σ. From the above discussion, it is seen that while the locally valid NARMA model requires n past values of input and output, the generically globally valid extended NARMA model requires $2n+1$ past values in general.

Definition of Relative Degree (Cabrera and Narendra, 1995, 1996): As shown in equation (8), the input $u(k)$ at time k affects the output at time $k+1$. Quite offen, there is a delay between the time the input is applied and the output is affected. If this is d units, we say qualitatively that the system has a relative degree d. More precisely, let f_e denote the state dynamics $f(\cdot, 0)$ and f_e^i the i-fold iterated composition of f_e. Σ has relative degree d at $(x,u) = (0,0)$ if there exists a neighborhood $\mathcal{X}_d \times \pi_d$ of the equilibrium state $(0,0)$ for which the following conditions are satisfied for all $(x,u) \in \mathcal{X}_d \times \diamond_d$: $\frac{\partial}{\partial u}(h \circ f_e^k \circ f(x,u)) = 0$, $0 \leq k > d-1$ and $\neq 0$ for $k = d-1$.

A detailed discussion of different definitions of relative degree is given in (Cabrera and Narendra, 1995, 1996). According to the above definition, the relative degree of a nonlinear system may not be well defined. However, when it is well defined, it corresponds to the delay through the system as described earlier.

It has been shown (Cabrera and Narendra, 1995, 1996) that, given an SISO dynamical system Σ with relative degree d, an input-output representation exists which has the form

$$y(k+d) = \bar{\mathcal{F}}[\,y(k), \cdots, y(k-n+1), u(k), \cdots, u(k-n+1)] \quad (9)$$

Equation (9), due to its explicit incorporation of the relative degree, is more attractive for the design of controllers than equation (8).

4. THE NARMA MODEL FOR MULTIVARIABLE SYSTEMS

Equation (8) can also be used in the given form for the local representation of nonlinear multivariable systems. However, the convenient form (9) cannot be used directly since there are several inputs and outputs with possibly different relative degrees between them. In the multivariable case, the relative degree d_{ij} has to be defined for each input-output pair (u_j, y_i). If $min_j \; d_{ij} = d_i$, d_i is called the relative degree of the i^{th} output y_i. It corresponds to the minimum time delay between any one of the inputs and the i^{th} output.

For the linear multivariable system Σ_L (equation (2)) the following representation can be easily derived from the state equations:

$$\begin{bmatrix} y_1(k+d_1) \\ y_2(k+d_2) \\ \cdots \\ y_m(k+d_m) \end{bmatrix} = \sum_{i=0}^{n-1} \bar{A}_i y(k-i) + \sum_{j=0}^{n-1} \bar{B}_j u(k-j) \quad (10)$$

where $\bar{A}_i$ and $\bar{B}_j$ are matrices of appropriate dimensions, and $y(k)$ and $u(k) \in \mathbb{R}^m$. Once again, defining the desired value of the vector on the l.h.s. as $r(k)$, it can be shown that the input $u(k)$ can be determined as a linear combination of $r(k), y(k-i), i = 0, 1, \cdots, (n-1), j = 1, 2, \cdots, (n-1)$, if the matrix $\bar{B}_0$ is nonsingular.

In a similar way, using the procedure outlined in section 3, the following local representation can be derived for the nonlinear multivariable system Σ under the same condition of observability of Σ_L:

$$\begin{aligned} y_1(k+d_1) &= f_1[\,y(k), y(k-1), \cdots, y(k-n+1), \\ &\qquad u(k), u(k-1), \cdots, u(k-n+1)] \\ y_2(k+d_2) &= f_2[\,y(k), y(k-1), \cdots, y(k-n+1), \\ &\qquad u(k), u(k-1), \cdots, u(k-n+1)] \\ \vdots \quad &= \quad \vdots \\ y_m(k+d_m) &= f_m[\,y(k), y(k-1), \cdots, y(k-n+1), \\ &\qquad u(k), u(k-1), \cdots, u(k-n+1)] \end{aligned} \quad (11)$$

In the representation given by equation (11), the values of the inputs and outputs at times $k-n+1,\cdots,k$ determine the values of $y_1, y_2, \cdots, y_m$ at times $k+d_1, k+d_2, \cdots, k+d_m$ respectively. This representation forms the basis of the identification and control of nonlinear multivariable systems using neural networks.

5. IDENTIFICATION

The problem of identification consists of setting up a suitable identification model and adjusting the parameters of the model to optimize a preformance function based on the output(s) of the plant and identification model output(s).

Series-Parallel and Parallel Identification Models: Assuming that a NARMA representation of the form (11) is known to exist, neural networks can be used in two different ways to generate the identification models. In the first, called the series-parallel model, the measured inputs as well as outputs of the plant are used to generate a predicted value of the outputs at appropriate future instants of time:

$$\begin{aligned}
\hat{y}_1(k+d_1) &= N_1[\ y(k), y(k-1), \ldots, y(k-n+1), \\
&\qquad u(k), u(k-1), \ldots, u(k-n+1)] \\
\hat{y}_2(k+d_2) &= N_2[\ y(k), y(k-1), \ldots, y(k-n+1), \\
&\qquad u(k), u(k-1), \ldots, u(k-n+1)] \\
\vdots \quad &= \quad \vdots \\
\hat{y}_m(k+d_m) &= N_m[\ y(k), y(k-1), \ldots, y(k-n+1), \\
&\qquad u(k), u(k-1), \ldots, u(k-n+1)]
\end{aligned} \tag{12}$$

where $N_i,\ i = 1 \cdots m$ are suitably defined neural networks. The advantage of using such a model is that, if the performance function to be minimized is defined as $\Sigma_{i=1}^{m}[\hat{y}_i(k+d_i) - y_i(k+d_i)]^2$, the parameters of N_i can be adjusted using standard static back-propagation methods. It is worth pointing out that equation (12) is merely a predictor of the output values at discrete instant of future time, and hence cannot be used effectively as a model (*i.e.*,for predicting future outputs based on different input sequences). However, it is adequate for adaptive control purposes. If a model is required for long-term prediction purposes, the following parallel model may be preferable:

$$\begin{aligned}
\hat{y}_1(k+d_1) &= N_1[\ \hat{y}(k), \hat{y}(k-1), \ldots, \hat{y}(k-n+1), \\
&\qquad u(k), u(k-1), \ldots, u(k-n+1)] \\
\hat{y}_2(k+d_2) &= N_2[\ \hat{y}(k), \hat{y}(k-1), \ldots, \hat{y}(k-n+1), \\
&\qquad u(k), u(k-1), \ldots, u(k-n+1)] \\
\vdots \quad &= \quad \vdots \\
\hat{y}_m(k+d_m) &= N_m[\ \hat{y}(k), \hat{y}(k-1), \ldots, \hat{y}(k-n+1), \\
&\qquad u(k), u(k-1), \ldots, u(k-n+1)]
\end{aligned} \tag{13}$$

Since equation (13) represents a nonlinear dynamical system, the adjustment of the parameters of the neural networks N_i will require dynamic gradient methods, which are computationally intensive. Further, the stability of the model has to be established. A related question worth investigating is whether a series-parallel model can be used to identify the parameters of the nerual networks, and in turn used to construct a stable parallel model. Intuitively, if the series-parallel identification is sufficiently accurate, the resulting parallel model will be a stable dynamical system.

6. APPROXIMATE SISO AND MULTIVARIABLE MODELS

For identification purposes, depending upon the assumptions made, the NARMA model (either SISO (equation (9)) or multivariable (equation (11))) is the best model that is currently available. However, it is not very convenient for the purposes of control. In the case of SISO systems, it can be shown (Cabrera and Narendra, 1995, 1996) that a control input of the form

$$\begin{aligned}
u(k) = \gamma[\ & y(k), \ldots, y(k-n+1), y^*(k+d), \\
& u(k-1), \ldots, u(k-n+1)]
\end{aligned}$$

exists which results in the system tracking a desired output $y^*(k)$ either exactly after a finite number of steps or asymptotically with time. The practical realization of γ poses serious problems since the controller is in a feedback loop and has to be adjusted using one of several techniques (such as back propagation) to minimize the control error. The problem (as in the case of the parallel identification model) is computationally intensive, since dynamic rather than static gradient methods have to be used. In practice, approximations are invariably made to simplify the problem. These include the use of static gradients in the place of dynamical gradients, and model Jacobians in the place of plant Jacobians.

The same problem also arises for multivariable plants, though in a more complex form. For such plants, it has been shown that a control input of the form

$$\begin{aligned}
u(k) = \Phi_c[\ & y(k), y(k-1), \ldots, y(k-n+1), \\
& y_1^*(k+d_1), \ldots, y_m^*(k+d_m), \\
& u(k-1), \ldots, u(k-n+1)]
\end{aligned} \tag{14}$$

exists to track a desired output $y^*(k)$, if the linearized multivariable system Σ_L can be decoupled (i.e., the matrix $\overline{B}_0$ in equation (10) is nonsingular)(Narendra and Mukhopadhyay, 1994). Assuming that Φ_c is realized by a neural network, the difficulties in computing the dynamic gradients to adjust the latter's weights also becomes apparent in this case.

If approximations are inevitable, the question naturally arises as to when they should be made. Is an exact controller for an approximate model better that an approximate controller for an exact model such as NARMA or its extensions? This question is discussed by Narendra and Mukhopadyay in (Narendra and Mukhopadhyay, 1997) where two nonlinear approximations (NARMA-L1 and NARMA-L2) are suggested for the NARMA model for SISO systems. These are based on two different Taylor series expansions of the RHS of equation (9). Due to space limitations, only the models are presented here. For further results concerning both models, the reader is refered to (Narendra and Mukhopadhyay, 1997).

$\underline{NARMA-L1}$: The first approximate model NARMA-L1 is described by the input-output equation

$$
\begin{aligned}
y(k+d) = f_0[y(k),\ldots,y(k-n+1)] + \\
\sum_{i=0}^{n-1} g_i[y(k),\ldots,y(k-n+1)]u(k-i)
\end{aligned}
$$

where $f_0 = \overline{F}[y(k),\ldots,y(k-n+1),0,\ldots,0]$ and $g_i = \frac{\partial \overline{F}}{\partial u(k-i)}|_{y(k),\ldots,y(k-n+1),0,\ldots,0}$, and is obtained by expanding $\overline{F}$ in equation (9) around $(y(k),\ldots,y(k-n+1),0,\ldots,0)$.

$\underline{NARMA-L2}$: The second model is described by the equation

$$
\begin{aligned}
y(k+d) = & F_1[\ y(k),\ldots,y(k-n+1), \\
& u(k-1),\ldots,u(k-n+1)] + \\
& F_2[\ y(k),\ldots,y(k-n+1), \\
& u(k-1),\ldots,u(k-n+1)]u(k)
\end{aligned}
$$

and is obtained by expanding $\overline{F}$ in Taylor series around $[y(k),\ldots,y(k-n+1),0,u(k-1),\ldots,u(k-n+1)]$.

Error bounds for the two models are given in (Narendra and Mukhopadhyay, 1994) in terms of the maximum matrix norm M_1 of the Hessian matrix of $\overline{F}$ with respect to $[u(k),\ldots,u(k-n+1)]^T$ for NARMA-L1 and $\frac{\partial^2 \overline{F}}{\partial u(k)^2}$ for NARMA-L2 when evaluated over a neighborhood of the equilibrium state. The great advantage of the two approximate models for the control problem is that (when they are valid), they make the generation of a control input substantially simpler. In fact, $u(k)$ can be computed algebraically in the two cases so that the computational control problem in completely bypassed and the principal problem becomes one of identification.

For multivariable systems, similar approximate models can also be used. For example, nonlinear multivariable NARMA-L2 model is described by input-output equations of the form

$$
\begin{aligned}
y_i(k+d_i) = & f_i[\ y(k),\ldots,y(k-n+1), \\
& u(k-1),\ldots,u(k-n+1)] \\
& +\sum_{j=1}^{m} g_{ij}[\ y(k),\ldots,y(k-n+1), \\
& u(k-1),\ldots,u(k-n+1)]u_j(k)
\end{aligned}
\tag{15}
$$

where $u(k), y(k) \in R^m$ and $f_i, g_{ij} : R^{(2n-1)m} \rightarrow R$.

When such an approximation model is adequate, the control input $u(k)$ to follow a desired trajectory $y^*(k)$ is given by:

$$
u(k) = \begin{bmatrix} g_{11}[.] & \cdots & g_{1m}[.] \\ \cdots & \cdots & \cdots \\ g_{m1}[.] & \cdots & g_{mm}[.] \end{bmatrix}^{-1} \begin{bmatrix} y_1^*(k+d_1) - f_1[.] \\ \cdots \\ y_m^*(k+d_m) - f_m[.] \end{bmatrix}
\tag{16}
$$

which relies on the nonsingularity of the matrix $[g_{ij}]$ where each g_{ij} is approximated by a suitable neural network.

7. SIMULATION RESULTS

The objective of this simulation experiment is to study how the NARMA model and its approximation NARMA-L2 model compare in identifying a complex nonlinear multivariable plant. The plant is chosen to be the same as that mentioned in (Narendra and Mukhopadhyay, 1994) for purposes of control, and is described by the equations

$$
\begin{aligned}
x_1(k+1) &= 0.9x_1(k)sin[x_2(k)] \\
&\quad +(2+1.5\frac{x_1(k)u_1(k)}{1+x_1^2(k)u_1^2(k)})u_1(k) \\
&\quad +(x_1(k)+\frac{2x_1(k)}{1+x_1^2(k)})u_2(k) \\
x_2(k+1) &= x_3(k)(1+sin[4x_3(k)])+\frac{x_3(k)}{1+x_3^2(k)} \\
x_3(k+1) &= (3+sin[2x_1(k)])u_2(k) \\
y_1(k) &= x_1(k) \\
y_2(k) &= x_2(k)
\end{aligned}
$$

where $x(k) = [x_1(k), x_2(k), x_3(k)]^T$ represents the state, $u(k) = [u_1(k), u(2(k)]^T$ the input and $y(k) = [y_1(k), y_2(k)]^T$ the output, at instant k. The delay from either of the inputs to y_1 is unity, while the delay to y_2 is three from input u_1 and two from input u_2. Hence, $d_1 = 1,\ d_2 = 2$.

When the NARMA model is used, the model equations are described by

$$\hat{y}_1(k+1) = N_1[\ y(k), y(k-1), y(k-2), u(k), u(k-1), u(k-2)]$$
$$\hat{y}_2(k+2) = N_2[\ y(k), y(k-1), y(k-2), u(k), u(k-1), u(k-2)]$$

where both y and u at every instant are two-dimensional vectors. Hence, the neural networks N_1 and N_2 each had 12 inputs. In the case of the NARMA-L2 model, the input-output equations used were of the form

$$\begin{aligned}\hat{y}_1(k+1) = {} & N_3[\ y(k), y(k-1), y(k-2), u(k-1), u(k-2)] \\ & + N_4[\ y(k), y(k-1), y(k-2), u(k-1), u(k-2)]u(k) \\ \hat{y}_2(k+2) = {} & N_5[\ y(k), y(k-1), y(k-2), u(k-1), u(k-2)] \\ & + N_6[\ y(k), y(k-1), y(k-2), u(k-1), u(k-2)]u(k)\end{aligned}$$

Hence, all four neural networks N_3, N_4, N_5 and N_6 had 10 inputs. While N_3 and N_5 had one output each, N_4 and N_6 generated two outputs corresponding to the time-varying coefficients of the two inputs. In all cases, multilayer networks were used in the model instead of radial basis function networks due to the high dimensionality of the input spaces.

For both models (NARMA and NARMA-L2), the plant was excited with a uniformly random input $u_1(k) \in [-0.75, 0.75]$, and the weights of the various neural networks constituting the models were adjusted for 100,000 time steps on the basis of the identification errors $\hat{y}_1(k) - y_1(k)$. Following this, the identification models were fixed and tested for performance on a test input. The test input chosen in both cases was

$$u_1(k) = 0.375\sin(2\pi k/10) + 0.375\sin(2\pi k/25)$$
$$u_2(k) = 0.15\sin(2\pi k/15) + 0.15\sin(2\pi k/40)$$

The test results for identification are shown in Figure 1. It is seen from part (a) that the NARMA model performs quite well in tracking the outputs of the distinctly nonlinear plant, although at certain points, the error in y_1 is slightly larger than that in y_2. Part (b) shows the performance obtained with the approximate NARMA-L2 model. In this particular simulation study, the errors obtained with NARMA-L2 model are found to be very similar to those resulting from the use of the NARMA model. Hence, the NARMA-L2 model, although merely an approxmation to the NARMA model, is adequate to identify the plant. As mentioned earlier, the primary advantages of the former become apparent in the control context where the computation of the control inputs can be carried out algebraically on the basis of the identification model. For simulation results related to control using both NARMA and NARMA-L2 models, the reader is referred to (Narendra and Mukhopadhyay, 1994).

Acknowledgment

The first author was supported by NSF grant # ECS-9521405 during the course of the research reported here. The second author is supported by NSF CAREER grant # ECS-9623961.

8. REFERENCES

Aeyels, D. (1981). Generic observability of differential systems. *SIAM Journal of Control and Optimization* **19**, 595–603.

Cabrera, J.B.D. and K.S. Narendra (1995). On regulation and tracking in nonlinear discrete-time systems: Part I - State variables accessible. *Technical Report No. 9515, Center for Systems Science , Yale University.*

Cabrera, J.B.D. and K.S. Narendra (1996). On regulation and tracking in nonlinear discrete-time systems: Part II - Nonlinear ARMA models. *Technical Report No. 9602, Center for Systems Science, Yale University.*

Leontaritis, I.J. and S.A. Billings (1985). Input-output parametric models for nonlinear systems: Part 1 - deterministic nonlinear systems. *International Journal of Control* **41(2)**, 303–328.

Levin, A.U. and K.S. Narendra (1995). Recursive identification using feedforward neural networks. *International Journal of Control* **61**, 533–547.

Levin, A.U. and K.S. Narendra (1996). Control of nonlinear dynamical systems using neural networks: Part II - Observability, identification and control. *IEEE Transactions on Neural Networks* **7(1)**, 30–42.

Narendra, K.S. and K. Parthasarathy (1990). Identification and control of dynamical systems using neural networks. *IEEE Trans. on Neural Networks* **1**, 4–27.

Narendra, K.S. and S. Mukhopadhyay (1994). Adaptive control of nonlinear multivariable systems using neural networks. *Neural Networks* **7(5)**, 737–752.

Narendra, K.S. and S. Mukhopadhyay (1997). Adaptive control using neural networks and approximate models. *IEEE Transactions on Neural Networks* **8(3)**, 475–485.

Sontag, E.D. (1979). On the observability of polynomial systems. *SIAM Journal of Control and Optimization* **17**, 139–151.

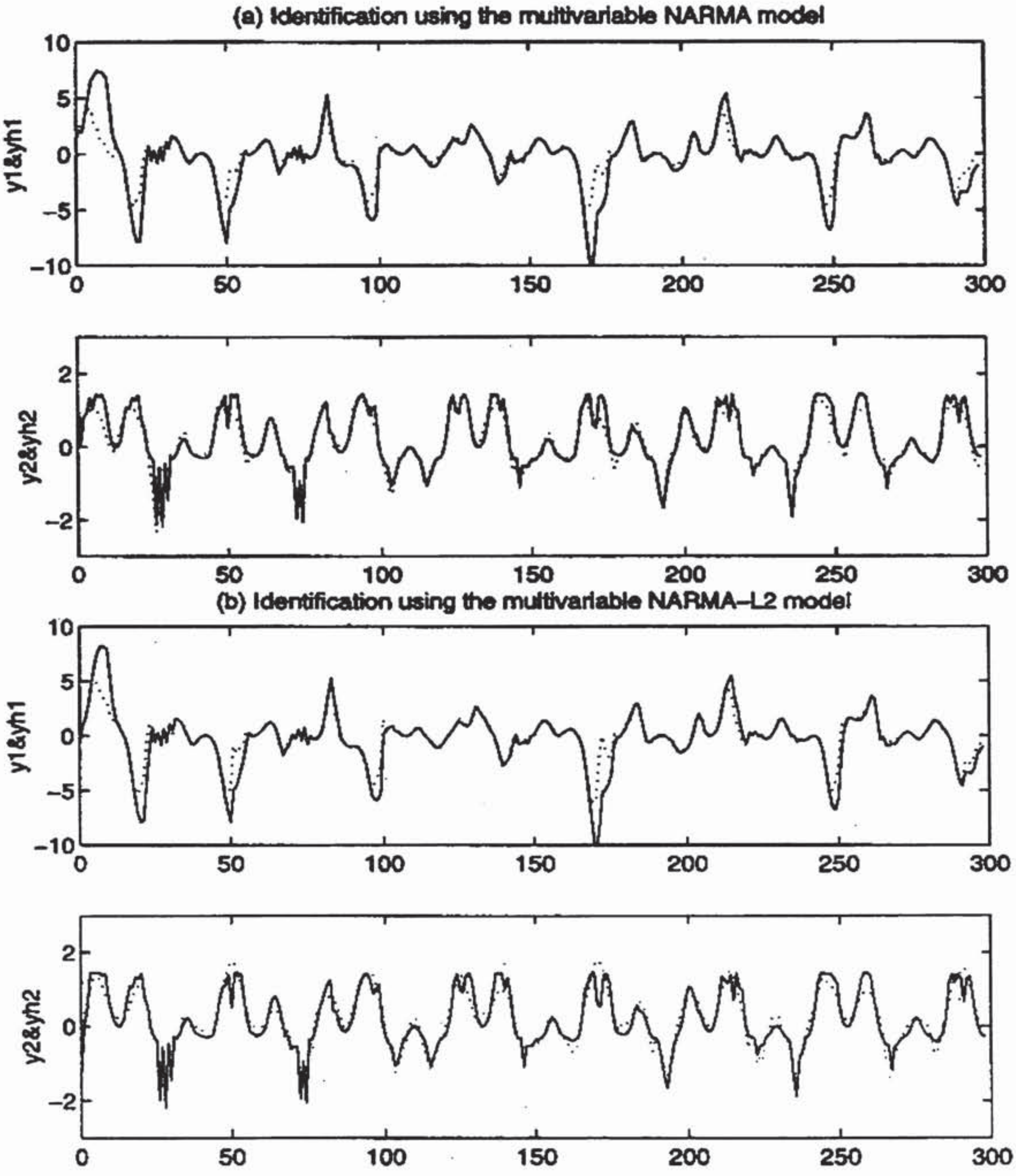

Figure 1: Identification of a nonlinear multivariable plant using NARMA and NARMA-L2 models. In all plots, the solid line represents the plant output, and the dotted line corresponds to the model output.