Available online at www.sciencedirect.com
ScienceDirect
Neurocomputing 70 (2007) 960–974
NEUROCOMPUTING
www.elsevier.com/locate/neucom

# Quasi-sliding mode control strategy based on multiple-linear models

Jeongho Cho[a,*], Jose C. Principe[a], Deniz Erdogmus[b], Mark A. Motter[c]

[a]*Computational NeuroEngineering Laboratory, University of Florida, Gainesville, FL 32611, USA*
[b]*Department of Computer Science and Electrical Engineering, Oregon Health & Science University, Portland, OR 97006, USA*
[c]*Electronics System Branch, NASA Langley Research Center, Hampton, VA 23681, USA*

Received 25 April 2005; received in revised form 25 June 2006; accepted 3 July 2006
Communicated by J. Zhang
Available online 10 October 2006

**Abstract**

In this paper, a multiple discrete quasi-sliding mode (QSM) control scheme is proposed for a general class of nonlinear discrete time systems with unknown dynamical equations, provided that input–output data is available for system identification. The self-organizing map (SOM) is employed to divide the state space into local regions such that it associates the operating region where a local linear model is the winner with a local quasi-sliding mode controller (QSMC). Switching of the controllers is done synchronously with the active local linear model that tracks the different operating conditions. The simulation results show that the proposed controller outperforms tracking the desired trajectory in noisy environments either with a global controller or simpler controllers based on multiple models.


*Keywords:* Multiple models; Sliding mode control; Self-organizing map; Unknown nonlinear system

## 1. Introduction

The identification of unknown nonlinear dynamical systems has received considerable attention in recent years since it is an indispensable step towards controller design for nonlinear systems [27]. Specifically, the concept of multiple models with switching has been an area of interest in control theory in order to simplify both the modeling and the controller design [22,26]. Local modeling derives a model based on neighboring samples in the operating space. If a function $f$ to be modeled is complicated, there is no guarantee that any given global representation will approximate $f$ equally well across all space. In this case, the dependence on representation can be reduced using local approximation where the domain of $f$ is divided into local regions and a separate model is used for each region [8,17,38,42].

In a number of local modeling applications, a self-organizing map (SOM) has been utilized to divide the operating regions into local regions [31,32,42]. The SOM is particularly appropriate for switching, because it converts complex, nonlinear statistical relationships of high-dimensional data into simple geometric relationships that preserve the topology in the feature space [18]. Thus the role of the SOM is to discover patterns in the high-dimensional state space and divide it into a set of regions represented by the weights of each processing element (PE). Under some mild conditions, it has been shown that multiple models can uniformly approximate any system on a closed subset of the state space provided a sufficient number of local models are given [38,42]. Generally, control using multiple models is categorized in two approaches: global model-based control using local models and multiple model-based control with switching.

Global controller design with the aid of multiple models has been extensively reported in the literature [9,14,21,34,36]. Gain scheduling has been perhaps the most common systematic approach to control nonlinear systems in practice due to simple design and tuning [28,33,37,43]. The multiple model adaptive control approach differs from gain scheduling mainly by the use of an estimator-based scheduling algorithm used to weight the local controllers. Murray-Smith and Hunt [21] utilized an extended RBF network where each

*Corresponding author. Tel.: +1 352 273 5409; fax: +1 352 273 5411.
*E-mail address:* jeongho@cnel.ufl.edu (J. Cho).

0925-2312/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2006.07.011

local model is a linear function of the input and they reported great success for control problems. The overall controller is designed based on the local models and a validity function to guarantee smooth interpolation. Similarly, Foss et al. [9] and Gawthrop and Ronco [14] employed model predictive controllers and self-tuning predictive controllers, respectively, using multiple models. Palizban et al. [29] attempted to control nonlinear systems with the linear quadratic optimal control technique using multiple-linear models and provided the stability condition for the closed loop system. Ishigame et al. [16] proposed the sliding mode control scheme based on fuzzy modeling composing a weighted average of linear systems to stabilize an electric power system.

In contrast, Narendra et al. [26] proposed the multiple model approach in the context of adaptive control with switching where local model performance indices have been used to select the local controller. Subsequently, Narendra and Balakrishnan [23] proposed different switching and tuning schemes for adaptive control that combines fixed and adaptive models yielding a fast and accurate response. Principe et al. [32] proposed a SOM-based local linear modeling strategy and predictive multiple model switching controller to control a wind tunnel and showed improved performance with decreased control effort over both the existing controller and an expert human-in-the-loop control. Later Narendra and Xiang [25] proved that the adaptive control using multiple models is globally stable and that the tracking error converges to zero in the deterministic case. Diao and Passino [7] applied multiple model-based adaptive schemes to the fault tolerant engine control problem. A linear robust adaptive controller and multiple nonlinear neural network-based adaptive controllers were exploited by Chen and Narendra [4]. Thampi et al. [40,41] have also shown the applicability of the multiple model approach based on the SOM for flight control.

The control of nonlinear systems considered in this paper has been an important research topic and many approaches have been proposed. While classical control techniques have produced many highly reliable and effective control systems, great attention has been devoted to the design of variable structure control systems (VSCS). Variable structure systems (VSS) are a special class of nonlinear systems characterized by a discontinuous control action, which changes structure upon reaching a set of switching hyperplanes. During the sliding mode, the VSCS has invariance properties, yielding motion that is remarkably good in rejecting certain disturbances and parameter variations [10,20,35,39].

However, sliding mode control systems (SMCS) that were originally conceived for continuous-time systems may not perform well—or may even lead the system to instability—when direct digital implementation is attempted. Thus, many researchers have either addressed the limitations when direct implementation is done or have proposed designs that take the sampling process into account. Milosavljevic [20] was among the first researchers to formally state that the sampling process limits the existence of a true sliding mode. In light of this, definitions of quasi-sliding mode (QSM) have been suggested and the conditions for the existence of such modes have been investigated. Sarpturk et al. [35] specifically addressed the stability issue and gave necessary and sufficient convergence and sliding conditions. Discrete sliding mode tracking controller based on an input–output model in the presence of modeling uncertainty and disturbances have been considered earlier [3,5,11,19,30]. Furuta [11] designed a discrete VSS type self-tuning controller using an adaptive parameter estimator where the control input included a linear feedback term and a switching term with the equivalent control region. Lee and Oh [19] suggested a modified discrete VSS type self-tuning controller and improved the stability of the controller by modifying the sector with separate gains and the equivalent control algorithm. Recently, Chen et al. [5] proposed a discrete robust adaptive QSM tracking controller for the input–output system without knowing the upper and lower bounds of the unknown parameters, which overcome the unpractical assumptions of [11,19] since the bounds of the unknown parameters can hardly ever be known in practice. On the other hand, Gao et al. [12] presented an algorithm that drives the system state to the vicinity of a switching hyperplane in the state space, rather than to a sector of a different shape [11]. They specified desired properties of the controlled systems and proposed a reaching law-based approach for designing the discrete-time sliding mode control law. Later, modified quasi-sliding mode control (QSMC) strategy with a reaching law approach was proposed by Bartoszewicz [1] to guarantee better robustness and improved performance.

Most of the VSCS proposed in the literature have been developed mainly based on the state-space model with the assumption that all state variables are measurable or on the input–output model for a linear system. But in some control problems, we are allowed to access only the input and the output of the nonlinear plant. In this case, an observer could be used to estimate the unmeasurable state variables if the state equations are known. Otherwise, this is not possible. This is where the multiple model-based control framework can be very attractive for nonlinear control problems since it is capable of not only utilizing this robust control technique for nonlinear systems but also applying it for unknown systems. Thus, it is the purpose of this work to provide a new technique to design a sliding mode control law for unknown discrete-time nonlinear systems so that the amount of guesswork[1] is reduced, while attainable performance is increased. In this way, one of the difficulties in designing a SMC (that requires the complete knowledge of the plant to be controlled) can be removed as

[1]In most cases, we need to estimate the unknown parameters, unmodeled dynamics and bounded disturbances. Also, it should be noted that the SMC scheme works best when the plant is completely known.

well as the problems that arise due to the uncertainties of the plant model and measurement noise can be alleviated by incorporating the robustness provided by the sliding mode technique into the multiple modeling approach. In addition, we examined the effect by the modeling error due to the quantization of state space as well as by measurement noise to the proposed multiple model-based sliding mode control performance. It is shown that the switching scheme does not create an issue to be considered in order to guarantee BIBO stability of the overall system.

Simulation results using the proposed strategy for identification and control of nonlinear systems are presented to demonstrate the versatility of the algorithm. Results show that the switching linear models are a promising alternative for system identification when compared with a single global model. The overall system with the controller tracks the desired trajectory very well. Additionally, it offers excellent robustness under noise condition during the control process when compared with a global nonlinear controller and other multiple controller approaches.

## 2. Multiple model-based system identification

The idea of multiple modeling is to approximate a nonlinear system with a set of relatively simple local models valid in certain operating regions, such that the dynamic space is decomposed in the appropriate switching among very simple linear models. Multiple models are very appealing for modeling complex nonlinear systems due to the intrinsic simplicity, since we often cannot derive appropriate models from first principles, and are not capable of deriving accurate and complete equations for input–state–output representations of the systems [8,22,38]. Moreover, when such physical knowledge of the system is not available, models have to determine from a finite number of measurements of the system's inputs and outputs. Hence here we will consider deriving multiple models for the unknown nonlinear system-based solely on input–output data.

Under the observability assumption, a discrete-time nonlinear dynamic system, $f$, can be described by a Nonlinear Auto-Regressive with eXogenous input (NARX) model that is an extension of the linear ARX model, and represents the system by a nonlinear mapping of past inputs and output terms to future outputs, that is

$$y_k = f(y_{k-1}, \cdots, y_{k-m}, u_{k-1}, \cdots, u_{k-n}). \tag{1}$$

Here $y_k \in Y \subset \Re^p$ is the output vector and $u_k \in U \subset \Re^q$ is the input vector. For simplicity, we will set $p = q = 1$. Let the $(m+n)$-dimensional basis vector be

$$\psi_k = [\psi_k^y, \psi_k^u] = [y_{k-1}, \cdots, y_{k-m}, u_{k-1}, \cdots, u_{k-n}], \tag{2}$$

where $\psi_k$ is in the set $\Psi = Y^m \times U^n$. If the nonlinear function $f(\cdot)$ is invertible with respect to the input $u_k$, then a controller may be constructed by training an inverse neural network. Unfortunately, most nonlinear functions are not invertible, so the application of this approach is limited. Also, when the environment of a system changes abruptly, the original model (and hence the controller) is no longer valid. In order to solve these difficulties, it is tempting to use a methodology that decomposes the overall modeling problem into a set of simpler local modeling problems, each for a different operating region. On the other hand, creating a set of models using the embedded input–output vectors in Eq. (2) may cause serious problem in the presence of large noise or outliers since the wrong predictive model due to noise may cause poor control. Hence, the selection of the right model is as important as creating models and designing controllers.

### 2.1. Determining the operating region

For the quantization of state space, a SOM is employed, since it has the characteristic of being a local framework that is able to limit the interference phenomenon and to preserve the topology of the data using neighborhood links between its PEs. It provides a codebook representation of the plant dynamics and organizes the different dynamic regions in topological neighborhoods. Thus we can create a set of models that are local to the data in the Voronoi tessellation created by the SOM. As the number of dependent variables is increased, the process becomes increasingly difficult to model accurately. Therefore models that use only a few of the observed variables may be more accurate than a model that uses all the observed variables. In the proposed scheme, we let the SOM represent only the current output and its past values to decide the winner that represents the operating region, and create the models with the control inputs as shown in Fig. 1 [6].

The SOM is trained to position the local models in the embedded output space. Let $\psi_k^y = [y_{k-1}, \cdots, y_{k-m}]$ denote the input vector for the SOM and $w_{i,k}$ denote the weight vector of PE $i$. With each vector, $\psi_k^y$, presented as the input to the network, the Kohonen learning algorithm [18] adaptively discretizes the continuous input space into a set of $N$ disjoint Voronoi cells. The response of a SOM to input $\psi_k^y$ is determined by the reference vector $w_{i^o}$ of the PE that produces the best match to the input:

$$i^o = \arg\min_i \{ |\psi_k^y - w_{i,k}| \}. \tag{3}$$

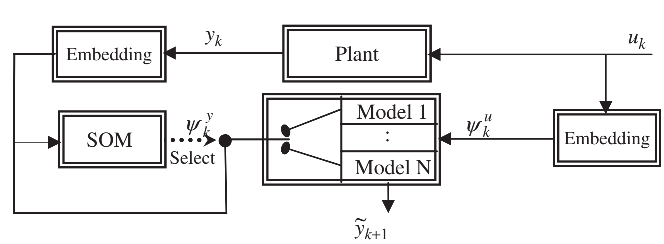



Fig. 1. Block diagram of SOM-based modeling for nonautonomous system.

Then the $k$th adaptation of the weights is done as follows:

$$w_{i,k+1} = w_{i,k} + \eta_k \Lambda_{i,k}(\psi_k^y - w_{i,k}), \quad i = i^o, \tag{4}$$

where $\eta_k$ is the learning rate and $\Lambda_{i,k}$ is a neighborhood function. A typical choice for $\Lambda_{i,k}$ is $\Lambda_{i,k} = \exp(-||r_i - r_{i^o}||^2/2\sigma_k^2)$, where $r_{i^o}$ and $r_i$ designate the position of the winning PE and the PE $i$ on the output lattice space, respectively, and $\sigma_k$ is a time decaying parameter which controls the effective adaptation coverage of $\Lambda_{i,k}$. Each new feature vector presented to the network will trigger a response that is the average for those feature vectors closest to it in the input data space.

### 2.2. Local linear modeling

After the operating regions are divided by the SOM the underlying dynamics $f$ in Eq. (1) is then approximated as

$$f \approx \bigcup_{i=1}^{N} fi, \tag{5}$$

where $N$ is the number of operating regions. Provided that necessary smoothness conditions on $f_i : \Psi \to Y$ are satisfied, a Taylor series expansion can be used around the operating point [38,42]. The first-order approximation about the system's each operating point produces $N$ local predictive ARX models $f_1, \cdots, f_N$ of the plant are described by

$$f_i(\psi_k) \approx \sum_{j=1}^{m} a_{i,j} y_{k-j} + \sum_{j=1}^{n} b_{i,j} u_{k-j}, \quad i = 1, \cdots, N, \tag{6}$$

where $a_{i,j}$ and $b_{i,j}$ are the parameters of the $i$th model. The parameters of each model is then obtained by directly fitting the embedded output samples and corresponding embedded control input samples in a least-square sense.

Each PE has an associated local model $\{\vec{a}_i, \vec{b}_i\}$ in Eq. (6) that represents the approximation of the local dynamics. The local model weights $\{\vec{a}_i, \vec{b}_i\}$ are computed directly from the desired signal samples $d_{i,j}$ and the input–output samples by a least-square fit within a Voronoi region centered at the current winning PE chosen from $\psi_k^y$. The size of the data samples in the region must be at least equal to the $(m+n)$-dimensional basis vector. The design procedure for this local model is as follows:

(1) Apply training data to the SOM and find the winning PE corresponding to the input $\psi_k^y$ such that we have winner–input pairs.
(2) Use the least-square fit to find the local linear model coefficients for the winning PE, $i^o$, where desired output vector $d_{i^o,j} \in \Re^M$ as

$$d_{i^o,j} = \left[\vec{a}_{i^o}^T \vec{b}_{i^o}^T\right] \begin{bmatrix} \psi_{i^o,j}^y \\ \psi_{i^o,j}^u \end{bmatrix} \quad \text{for } \forall j \in M, \tag{7}$$

where $\vec{\theta} = [\vec{a}_{i^o}^T \vec{b}_{i^o}^T]$ is the sought linear model coefficients, $M$ is the size of data involved in the winning PE $i^o$. Specifically, the least-squares problem

$$Y = \vec{\theta} X \tag{8}$$

is solved for $\vec{\theta}$, where $X \in \Re^{(m+n)\times M}$ is defined as a matrix that contains each input vector associated with the winning PE, and $Y \in \Re^M$ is defined as a vector that contains the target outputs.
(3) In testing, once the winning PE is determined the corresponding local model is chosen from the list of associated models. Apply the local model to obtain the estimated output

$$\tilde{y}_k = \vec{a}_{i^o}^T \psi_k^y + \vec{b}_{i^o} \psi_k^u. \tag{9}$$

Our proposed modeling methodology is summarized as follows: first, the delayed version of input–output joint space is decomposed into a set of operating regions that are assumed to cover the full operating space. Next, for each operating region we choose a simple linear ARX model to capture the dynamics of the region. Consequently, a nonlinear nonautonomous system is approximated by a concatenation of local linear models.

We developed a set of local linear models for the plant and switch them according to the measured output history. Thus, once the right local linear model is determined, the corresponding sliding mode controller can be designed easily by choosing an appropriate coefficients vector of the switching surface.

## 3. Discrete-time QSMC for local linear models

Once we identify the plant using multiple models, it is necessary to associate these models with corresponding controllers. In doing so, controllers can be designed a priori corresponding to each of the local models. In addition, if the nonlinear system can be adequately described by a linear model within a sufficiently small neighborhood of an operating point, the corresponding controller is easily designed through the linearized plant [22,26].

### 3.1. Structure of the controllers

As stated before, our principal objective is to determine a control input, $u_k$, which will result in the output, $y_{k+1}$, of the plant to track, with sufficient accuracy, a specified sequence, $d_{k+1}$. The system identification block has $N$ predictive models, denoted by $\{f_i\}_{i=1}^N$, in parallel. Corresponding to each model $f_i$, a controller $C_i$ is designed such that $C_i$ achieves the control objective for $f_i$. Therefore, at every instant one of the models is selected and the corresponding controller is used to control the actual plant. The number of controllers is determined by only the number of models since the control performance is very dependent upon how well the models are built. As the number of models is increased the models become increasingly accurate to represent the dynamics of the system. Very *small* modeling errors by large number of

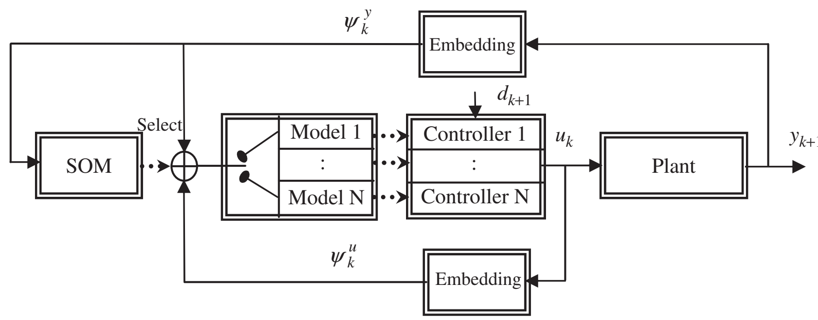



Fig. 2. Proposed SOM-based multiple controller scheme.

models definitely help one to build possibly a perfect set of controllers.

As a result, the set of local linear models simplifies the control design for a nonlinear plant. Instead of a global neuro-controller as in other adaptive control schemes [24], here a group of linear controllers associated with each identified model is sufficient to take care of the system over the whole operating region. In our proposed architecture, once the current operating region is determined by the SOM the associated controller is triggered so that the plant tracks the desired signal shown in Fig. 2. Consequently, the proposed control system can reach the set point fast, and even if the controller corresponding to a neighboring local model is used (due to selection errors), since nearby models are 'similar' there is an extra flexibility to match the set point with small error.

### 3.1. Control law for discrete-time QSMC

We adopted the reaching law approach proposed by Gao et al. [12] to the design of a set of local controllers which are "switched" as the system changes operating conditions. And we extended it such that the control input at each sampling instance is obtained by the local model selected by the SOM while the switching surface is kept the same. Also, their formulation was extended for input–output models as well as for the trajectory tracking problem.

Now we discuss the design of the control law for local linear models using the QSM control framework, where the system states move in a neighborhood around the sliding surface $s_k = 0$. The central advantage of the sliding mode control strategy is that it is an effective robust control strategy for incompletely modeled or uncertain systems. Thus, the feature of the proposed control scheme is that the robustness for disturbances can be obtained by the simple control logic based on the linear model for each region. Another feature of the strategy is that it guarantees convergence of the system output to a vicinity of the predetermined, fixed plane in finite time, specified a priori by the designer.

Consider one of the local single input–output models $f_i$ of the plant $f$ described by (6)

$$y_{k+1} = a_1 y_k + a_2 y_{k-1} + \cdots + a_m y_{k-m+1} + b_1 u_k + b_2 u_{k-1} + \cdots + b_n u_{k-n+1}. \tag{10}$$

Equivalently, the input–output model of the plant in Eq. (10) can be written as the state-space model[2]

$$\vec{x}_{k+1} = \Phi \vec{x}_k + \Lambda_1 u_k + \Lambda_2 u_{k-1} + \cdots + \Lambda_n u_{k-n+1}, \tag{11}$$

where $\vec{x}_k = [y_{k-m+1}, \cdots, y_{k-1}, y_k]^T \subset \Re^m$ is the system state vector which is available for measurement and $\Phi$ and $\Lambda_1, \cdots, \Lambda_n$ have the following forms:

$$\Phi = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 \\ a_m & a_{m-1} & a_{m-2} & a_{m-3} & \cdots & a_1 \end{bmatrix},$$

$$\Lambda_1 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_1 \end{bmatrix}, \Lambda_2 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_2 \end{bmatrix}, \cdots, \Lambda_n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_n \end{bmatrix}.$$

Also defining the tracking error vector as

$$\vec{e}_{k+1} = \vec{r}_{k+1} - \vec{x}_{k+1}, \tag{12}$$

where the desired signal vector is $\vec{r}_{k+1} = [d_{k-m+2}, \cdots, d_k, d_{k+1}]^T$, the switching surface is defined in the space of the tracking error vector given by

$$s_k = \vec{c}^T \vec{e}_k, \tag{13}$$

where $\vec{c} = [c_1, c_2, \cdots, c_m]^T$. Then an equivalent control is designed to satisfy the ideal QSM condition, $s_{k+1} = s_k = 0$, by

$$u_k^{eq} = (\vec{c}^T \Lambda_1)^{-1} \{ \vec{c}^T (\vec{r}_{k+1} - \Phi \vec{x}_k) - \vec{c}^T \Lambda_2 u_{k-1} - \cdots - \vec{c}^T \Lambda_n u_{k-n+1} \} \tag{14}$$

[2]In a formal state-space model, past values of the input should be included in the state vector using delay operators. For simplicity, we include only the system output's past values in the state vector in this notation.

and the closed-loop system response of the ideal QSM substituting (14) into with an equivalent control is given by

$$\vec{x}_{k+1} = \left\{I - \Lambda_1(\vec{c}^T\Lambda_1)^{-1}\vec{c}^T\right\}\Phi\vec{x}_k + \Lambda_1(\vec{c}^T\Lambda_1)^{-1}\vec{c}^T\vec{r}_{k+1}. \tag{15}$$

The system (15) can be viewed as a linear system with the input $\vec{r}_{k+1}$ and the output $\vec{x}_{k+1}$. To get an insight into the tracking capability of the system, (15) can be represented in terms of the tracking error $e_k = d_k - y_k$ by

$$e_{k+1} = -\frac{c_{m-1}}{c_m}e_k - \frac{c_{m-2}}{c_m}e_{k-1} - \cdots - \frac{c_1}{c_m}e_{k-m+2}. \tag{16}$$

Note that by designing the switching surface such that the roots of polynomial $\lambda^{m-1} + (c_{m-1}/c_m)\lambda^{m-2} + \cdots + (c_1/c_m)$ are inside of the unit circle, the error vanishes and thus the condition ensures asymptotic convergence to the desired output. An arbitrary positive scalar $c_m$ also determines the time taken to reach the sliding surface and can be adjusted to get a faster response.

For the SOM-based system identification, one needs to quantify the effect of the modeling error that will occur due to the quantization of state space induced by the SOM, and also by the wrong selection of the winning model. Consider model (10) in the presence of modeling error and measurement noise. The predicted output becomes

$$\begin{aligned}\hat{y}_{k+1} &= \hat{a}_1(y_k + \varepsilon_k) + \hat{a}_2(y_{k-1} + \varepsilon_{k-1}) \\ &\quad + \cdots + \hat{a}_m(y_{k-m+1} + \varepsilon_{k-m+1}) + \hat{b}_1u_k + \hat{b}_2u_{k-1} \\ &\quad + \cdots + \hat{b}_nu_{k-n+1},\end{aligned} \tag{17}$$

where a wrong local model $(\hat{a}_1, \hat{a}_2, \cdots, \hat{a}_m, \hat{b}_1, \hat{b}_2, \cdots, \hat{b}_n)$ is triggered by the SOM due to the noisy output measurement $y_k + \varepsilon_k$. Then, when $s_k = 0$, the overall tracking error response with an equivalent control is given by

$$\begin{aligned}e_{k+1} = &-\frac{c_{m-1}}{c_m}e_k - \frac{c_{m-2}}{c_m}e_{k-1} - \cdots - \frac{c_1}{c_m}e_{k-m+2} \\ &+ (a_1 - \hat{a}_1)(y_k + \varepsilon_k) + \cdots \\ &+ (a_m - \hat{a}_m)(y_{k-m+1} + \varepsilon_{k-m+1}) \\ &+ (b_2 - \hat{b}_2)u_{k-1} + \cdots + (b_n - \hat{b}_n)u_{k-n+1}.\end{aligned} \tag{18}$$

For simplicity, consider the error dynamics (18) when $m = 2$. Defining model parameter error $\vec{\Delta}_k = [a_1 - \hat{a}_1, a_2 - \hat{a}_2, b_1 - \hat{b}_1]^T$ and noise $\vec{n}_k = [\varepsilon_k, \varepsilon_{k-1}, \varepsilon_{k-2}]^T$

$$e_{k+1} = \bar{c}e_k + \vec{\Delta}_k^T(\vec{z}_k + \vec{n}_k), \tag{19}$$

where $\vec{z}_k = [y_k, y_{k-1}, u_{k-1}]^T$, $\bar{c} = c_1/c_2$, and $\bar{c}$ is chosen as $\bar{c} < 1$. We assume that $E[\varepsilon_k^2] = \gamma$ and $||\vec{\Delta}_k||^2 < \delta\|w_k - \hat{w}_k\|^2$ where the norm $\|w_k - \hat{w}_k\|$ is the Euclidean distance between the reference vector of the correct PE and that of the neighboring PE selected by the perturbed output measurements.[3] Also, it is assumed that the noise $\varepsilon_k$ is zero-mean and white. We have the following recursive formula for the tracking mean squared error:

$$\begin{aligned}E[e_{k+1}^2] &= E\Big[\bar{c}^2e_k^2 + 2\bar{c}e_k\vec{\Delta}_k^T(\vec{z}_k + \vec{n}_k) \\ &\qquad + \vec{\Delta}_k^T(\vec{z}_k + \vec{n}_k)(\vec{z}_k + \vec{n}_k)^T\vec{\Delta}_k\Big] \\ &= \bar{c}^2E[e_k^2] + \vec{\Delta}_k^T\vec{z}_k\vec{z}_k^T\vec{\Delta}_k + \vec{\Delta}_k^TE[\vec{n}_k\vec{n}_k^T]\vec{\Delta}_k.\end{aligned} \tag{20}$$

When we take the norm on each side in (20), the norm of the tracking error power is represented by

$$E[e_{k+1}^2] \leqslant \bar{c}^2E[e_k^2] + ||\vec{\Delta}_k||^2||\vec{z}_k||^2 + \gamma||\vec{\Delta}_k||^2, \tag{21}$$

where the Cauchy inequality is used on the second term on the right hand side. As $k \to \infty$, using the earlier assumptions on noise and model error bounds, the following bound on the steady-state tracking error power is obtained:

$$E[e_\infty^2] < \frac{\delta\|w_k - \hat{w}_k\|^2}{(1 - \bar{c}^2)}\left[\|\vec{z}_k\|^2 + \gamma\right]. \tag{22}$$

Note that the difference between the true model (winning PE) and the neighboring model (wrong PE) assigned by noisy input, $\delta\|w_k - \hat{w}_k\|^2$, is typically small, since neighboring SOM PEs represent neighboring regions in the dynamic space. Also, it should be noted that the error can still be very large if we choose $\bar{c}$ as close as 1. In contrast, by choosing $\bar{c}$ as small as possible, the closed-loop system may have very fast transient response, possibly too large unexpected overshoot. Thus we should be careful for determining $\bar{c}$ so as not to have large error. This problem will be discussed later in simulation results. If $\bar{c}$ is set to small enough it then follows that the error by choosing appropriate design parameters mentioned above will be bounded for a given modeling uncertainty and measurement noise bounded by $\gamma$. Moreover, this shows that the switching scheme does not create an issue to be considered in order to guarantee BIBO stability of the overall system.

Gao et al. [12] proposed a reaching law-based approach, which directly specifies the dynamics of the switching surface for designing the discrete-time sliding mode control law. For a discrete-time system described by (11), the reaching law for the discrete-time sliding mode control is

$$s_{k+1} - s_k = -\alpha Ts_k - \beta T\mathrm{sgn}(s_k),\ \alpha > 0,\ \beta > 0,\ 1 - \alpha T > 0, \tag{23}$$

where $T > 0$ is the sampling period. The state reaches the switching surface at a constant rate $-\beta T$ and the term $-\alpha T$ forces the state to approach the switching surfaces faster when $s_k$ is large. The inequality for $T$ guarantees that starting from any initial state, the trajectory will move monotonically towards the switching surface and cross it in finite time. The reaching law (23) always satisfies the reaching condition such that the discrete VSC system designed using the reaching law approach is always stable with a stable ideal QSM [12]. Then the control law is derived by comparing

$$s_{k+1} - s_k = \vec{c}^T\vec{e}_{k+1} - \vec{c}^T\vec{e}_k$$

[3] The first assumption states that measurement noise has finite power. The second assumption means model parameter error is bounded by the distance in the state space.

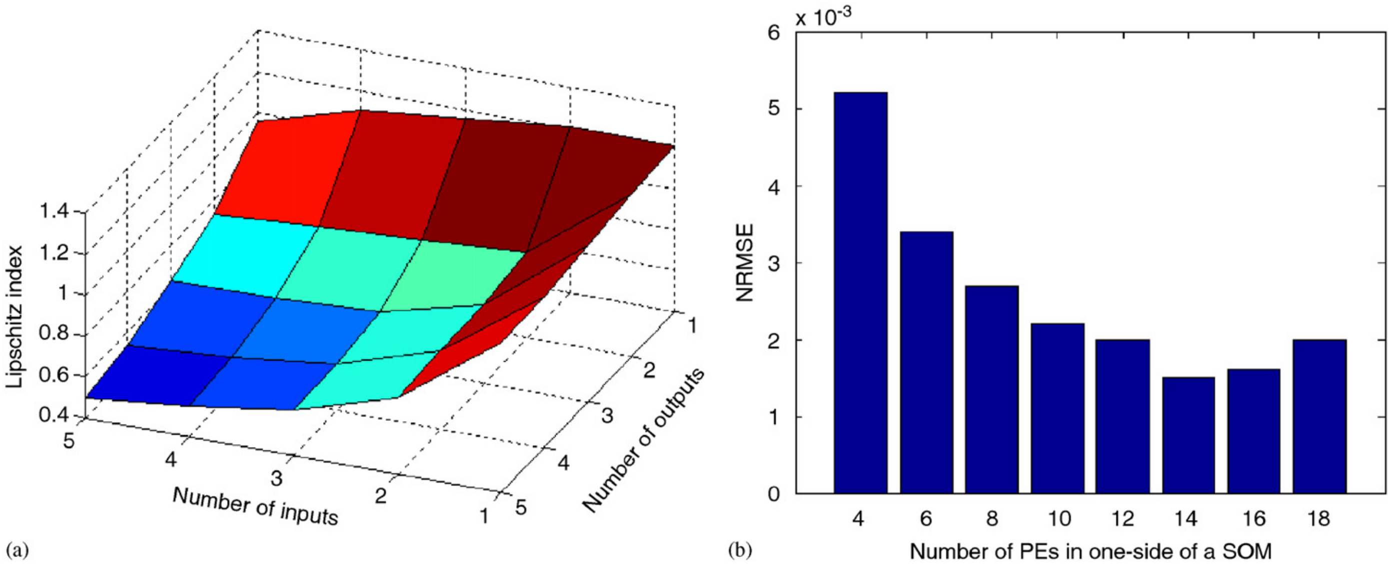



Fig. 3. Parameter selection to design multiple models: (a) Lipschitz index for determining the embedding dimension (b) Identification performance vs. network dimension on independently generated test data for choosing the size of a map.

with the reaching law (23), which yields

$$u_k = \left(\vec{c}^T \Lambda_1\right)^{-1} \left[\vec{c}^T r_{k+1} - \vec{c}^T \Phi \vec{x}_k - \vec{c}^T \Lambda_2 u_k - \cdots - \vec{c}^T \Lambda_n u_{k-n+1} + (\alpha T - 1) s_k + \beta T \operatorname{sgn}(s_k)\right]. \quad (24)$$

One advantage of this scheme is its simplicity and fast convergence to get the desired response. Another advantage is that the dynamic space is decomposed in the appropriate switching among very simple linear models, which leads to accurate modeling and control. A possible disadvantage of the proposed approach is that the overall stability may not be guaranteed due to the switching among models if the models are quite different from each other.

## 4. Simulation results

To examine the effectiveness of the proposed controller design methodology, discrete-time systems have been considered assuming the following: the only state available for measurements is $y_k = x_k^{(1)}$ and the nonlinear function $f$ is completely unknown. By assuming that the function $f$ is unknown, we confront a worst case (least prior knowledge) control design. Our objective is to design multiple sliding mode controller for unknown nonlinear plants that guarantees global stability and forces the output, $y_k$, to asymptotically track the desired signal, i.e., $|y_k - d_k| \to 0$, as $k \to \infty$ without any a priori knowledge of the plant.

**Example 1.** Biological reactor

In the bioreactor model, the microorganisms grow by consuming the substrate. At low concentrations, both microorganisms and substrate are assumed to be present [13]. Denoting by $x^{(1)}$ and $x^{(2)}$ the concentrations of microorganisms and substrate, respectively, we get the following discrete-time model by Euler discretization with sampling time 0.5 s:

$$\begin{aligned} x_{k+1}^{(1)} &= x_k^{(1)} + 0.5 \frac{x_k^{(1)} x_k^{(2)}}{x_k^{(1)} + x_k^{(2)}} - 0.5 u_k x_k^{(1)}, \\ x_{k+1}^{(2)} &= x_k^{(2)} - 0.5 \frac{x_k^{(1)} x_k^{(2)}}{x_k^{(1)} + x_k^{(2)}} - 0.5 u_k x_k^{(2)} + 0.05 u_k, \\ y_k &= x_k^{(1)} + z_k, \end{aligned} \quad (25)$$

where the control $u_k$ is the output flow rate that is uniformly distributed between 0 and 1 and $z_k$ is the measurement noise that is bounded by $|z_k| \leqslant 0.0004$. Suppose that there is no a priori knowledge of the system, but we assume that we have available input–output data samples. We chose the model as

$$\tilde{y}_k = f(y_{k-1}, y_{k-2}, u_{k-1}, u_{k-2}) \quad (26)$$

in accordance with Lipschitz index [15]. The theory states that the best embedding dimension is obtained when the index stops decreasing. However, as shown in Fig. 3(a), the index keeps decreasing, and thus we chose the number of inputs and outputs as the point where the index decreased the most.

First a SOM is trained with $(y_{k-1}, y_{k-2})$ over 6000 samples with the time decaying parameters, $\eta_k = 0.1/(1 + 0.003\,\mathrm{k})$ and $\sigma_k = (\sqrt{N}/2)/(1 + 0.003\,\mathrm{k})$ in Eq. (4), where $N$ is the number of PEs. After training square SOMs with various sizes on the input vector $(y_{k-1}, y_{k-2})$, local linear models are constructed from the embedded output (used for training) and the embedded control input corresponding each PE. The created models are tested by a newly created sequence of 400 samples for various sizes of the SOM. Based on generalization error shown in Fig. 3(b) the network dimension was chosen as $14 \times 14$. The model

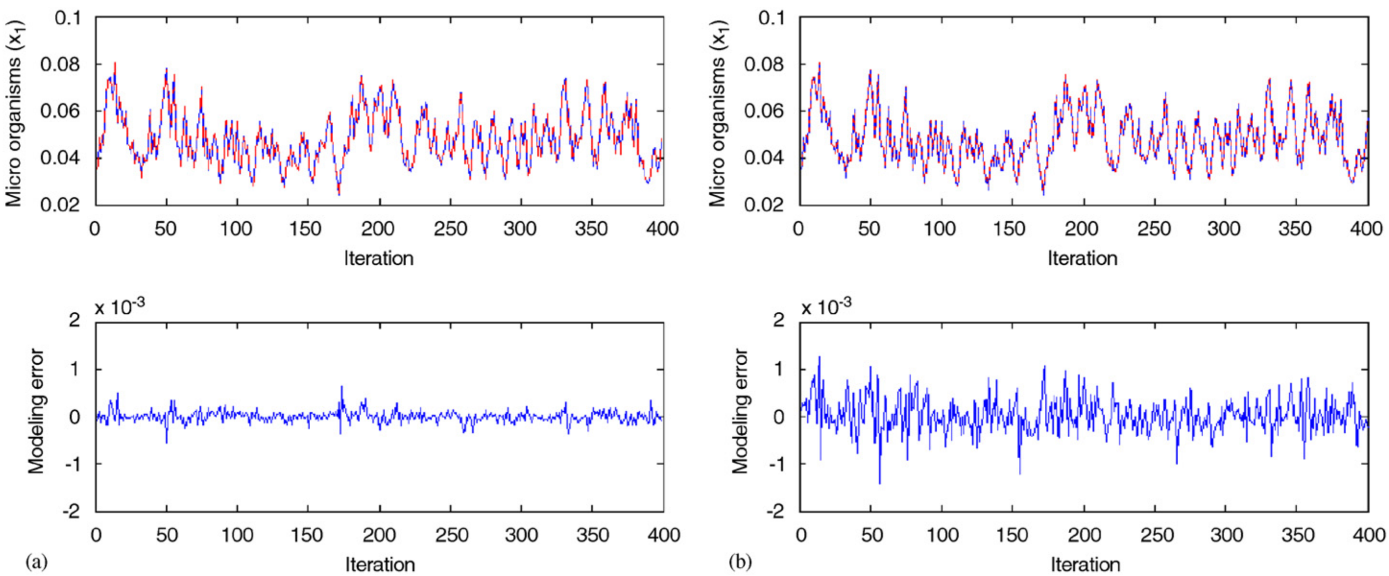



Fig. 4. System identification of nonlinear plant: (a) by multiple models (b) by a global model. The dashed line is the plant output, and the solid line is the output from the plant model.

performance shown in Fig. 3(b) achieves the smallest normalized root mean squared error (*NRMSE*) at this value.[4] Plant modeling performance with 196 multiple-linear models was compared with those by means of a conventional Time Delay Neural Network (TDNN), which was trained by Backpropagation algorithm with the constant learning rate of 0.001 for 3000 iterations on the same number of inputs and outputs as in local linear modeling, adopted as a global nonlinear model in Fig. 4. As we can see, the multiple models are very good approximation of the plant. The best result with the proposed method was a *NRMSE* of 1.4e-3 while with the TDNN[5] obtained a *NRMSE* of 4.4e-3. This result shows that the proposed multiple-linear modeling scheme outperforms the nonlinear global modeling paradigm.

Using the created multiple models, we design the sliding hyperplane in the controller by choosing arbitrary $c_1$ and $c_2$. Fig. 5 illustrates the effect of these parameters on the proposed control scheme under different noise levels. In most cases, as we place the pole ($c_1/c_2$) of (16) closer to the origin inside the unit circle, the controller showed better tracking performance. For instance, from the plot, we can say that the pole should be chosen as less than 0.5 to have the robustness against noise whose level is 25 dB of *SNR* since the error changes very slowly above 0.5. Thus the switching surface was chosen as $s_k = [1, -10][e_{k-1}, e_k]^T$ in order to get small error and short enough transient time as well.

Fig. 6 shows the plant responses of the closed-loop controlled system by the proposed method choosing the

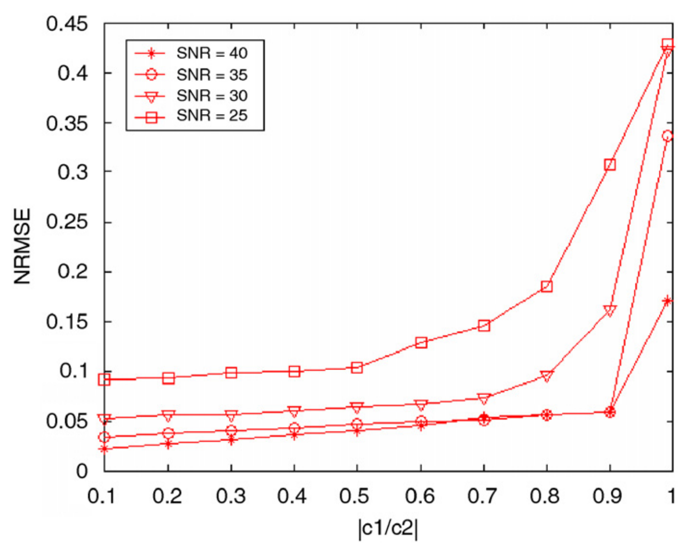



Fig. 5. Set-point tracking performance of the Bio-reactor for different sliding surface.

switching surface with $\vec{c} = [1, -10]^T$. The tracking objective was to make the output $y_k$ follow a desired reference signal generated by the control input $u_k = 0.2$ for $20 \leqslant k \leqslant 50$ and $u_k = 0.5$ otherwise. Also the behavior of the control input, the sliding surface evolution converging to zero, and the unmeasured state $x^{(2)}$ are presented in Fig. 6 where it can be seen that the controller performs impressively. From these results, it can be inferred that the closed-loop signals are bounded for this class of nonlinear systems even though the dynamics are not known a priori to the controller.

Additionally, we compared the performance of the proposed Multiple QSMC (MQSMC) for tracking a desired reference signal with that of the TDNN Controller

[4]Identification performance was evaluated by *normalized root mean squared error* (*NRMSE*).

[5]The number of PEs in the hidden layer of the TDNN is chosen as $7 \times 20$ Monte-Carlo simulations varying the size of the hidden layer.

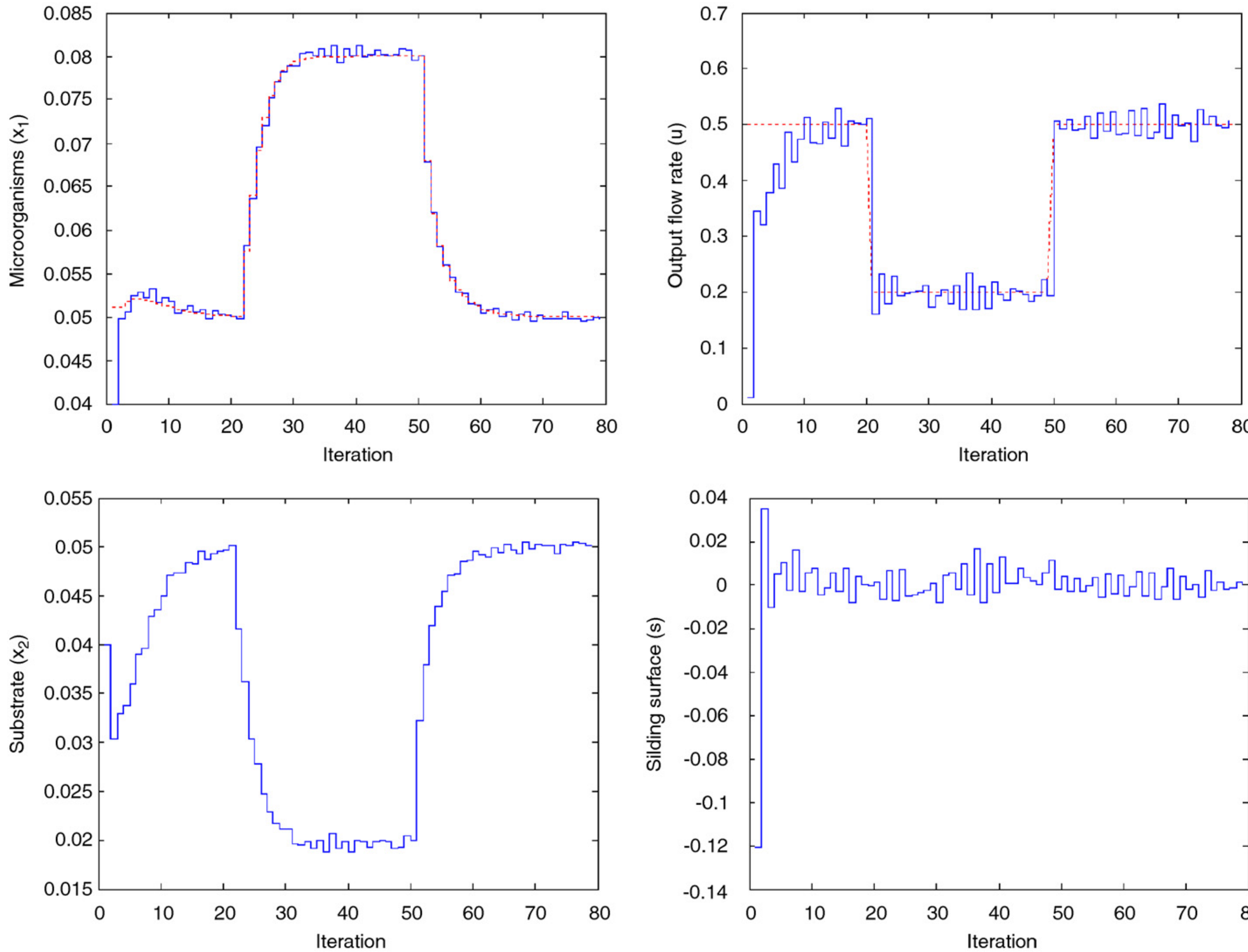



Fig. 6. Reference signal tracking performance by the MQSMC.

(TDNNC) trained through the TDNN model [44] shown previously, since a global nonlinear inverse controller, e.g., TDNNC, have been often utilized to control complex plants. The optimal number of PEs in the hidden layer of TDNNC was chosen as 10. Although it looks the proposed control scheme is too complicated for practical implementations it is very easy to implement since the local controller is selected from the lookup table as the system changes operating conditions and produce the control input by simple additions and multiplications. Also the implementation is much easier than fitting a global parametric model, such as a neural network and since each local piece of the model is linear, practical controller design techniques that have been developed for linear systems can be employed for each piece.

In Table 1 we show that multiple controllers demonstrate superior performance compared to a single global nonlinear controller for an arbitrary trajectory-tracking problem. In addition, we performed experiments for sinusoidal signal tracking by the TDNNC and the MQSMC. Fig. 7 shows the result of controlling the Microorganism population $x^{(1)}$. As can be seen in Fig. 7, the MQSMC converges faster to the desired signal than the TDNNC and exhibits reduced settling time.

Table 1
Comparison of control performance for an arbitrary trajectory tracking

| Controller | NRMSE |
|---|---|
| TDNNC | 9.3e-3 |
| MQSMC | 6.0e-3 |

**Example 2.** Discrete-time nonlinear system

Consider the following nonlinear discrete-time plant [2]:

$$\begin{aligned} x_{k+1}^{(1)} &= x_k^{(2)}, \\ x_{k+1}^{(2)} &= -\frac{3}{16}\left[\frac{x_k^{(1)}}{p+(x_k^{(2)})^2}\right] + x_k^{(2)} + u_k, \\ y_k &= x_k^{(1)} + z_k, \end{aligned} \tag{27}$$

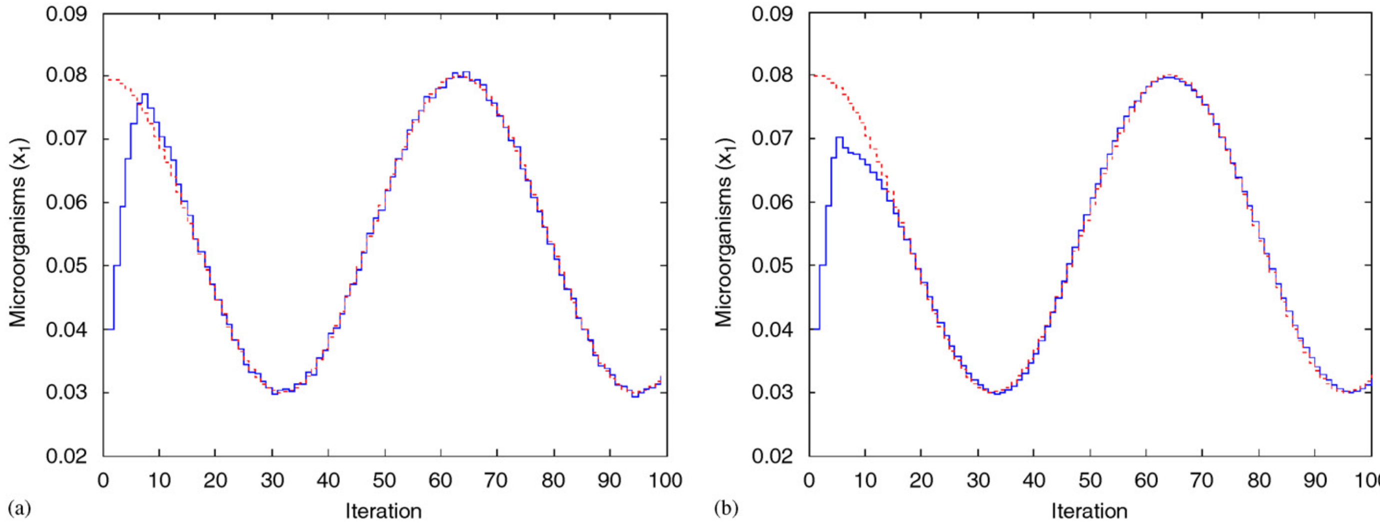



Fig. 7. Square-wave and sinusoidal signal tracking: (a) by the MQSMC (b) by the AIC.

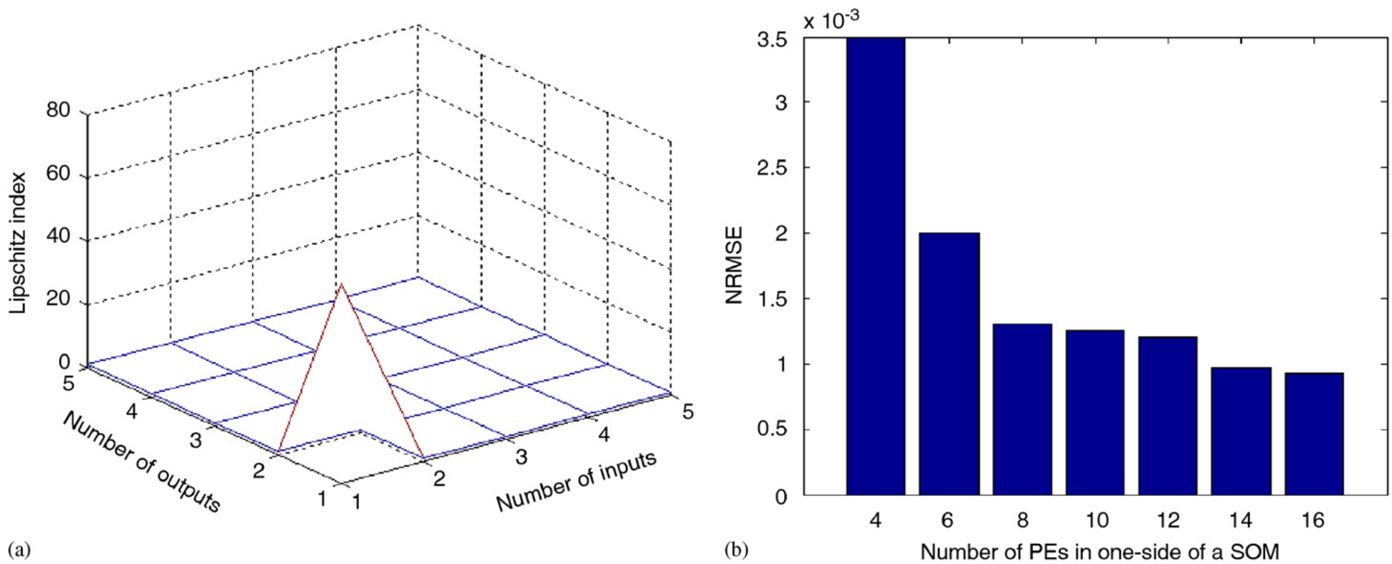



Fig. 8. Parameter selection to design multiple models: (a) Lipschitz index for determining the embedding dimension, (b) identification performance vs. network dimension on independently generated test data for choosing the size of a map.

where $p=1$, $u_k$ is the input and $z_k$ is an external disturbance. In Eq. (27), we consider a SISO model, assuming that only the output is available for measurement. The input is excited with an input signal that is uniformly distributed in $-0.5 \leqslant u \leqslant 0.5$. The model was assumed to be a second order in input and output (26) again based on Fig. 8(a).

After quantization of the embedded output space, a set of models was built with the input–output data samples for each PE. For testing, independently generated 400 data samples were used changing the size of the map. The best size of the map was determined as $8 \times 8$ since the performance did not improve much after 64 PEs (see Fig. 8b). Thus plant identification with 64 multiple models ($8 \times 8$) was tested in the absence of sensor noise as well as in the presence of sensor noise ($SNR = 20\,\mathrm{dB}$) with the plant input signal being uniformly distributed. Results of system identification are shown in Fig. 9. As we can see, the models provide a very good approximation of the plant visually based on the error curves, even when sensor noise exists. Also the proposed multiple modeling scheme was compared with a TDNN again with 5000 samples and a constant learning rate, 0.005. The best modeling result with the proposed method was a $NRMSE$ of around 6.0e-4 while with the TDNN[6] one obtained a $NRMSE$ of about 6.3e-3. When the testing data are perturbed by noise ($SNR = 20\,\mathrm{dB}$), the performances were a $NRMSE$ of

[6]The number of PEs in the hidden layer of the TDNN is chosen as $20 \times 20$ Monte-Carlo simulations varying the size of the hidden layer.

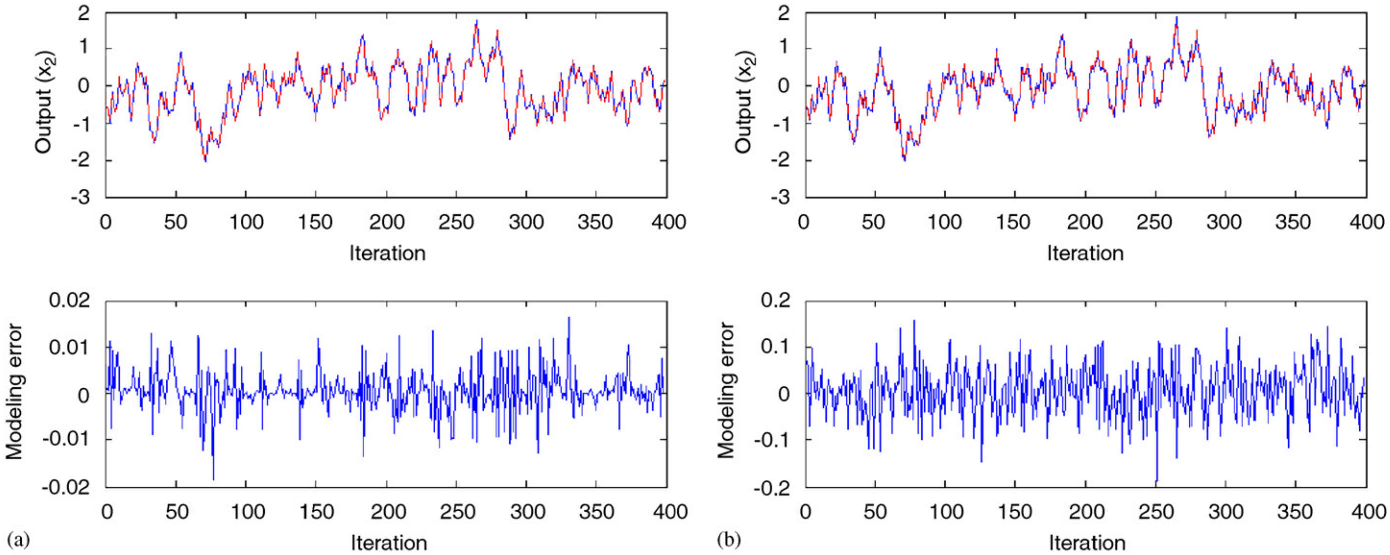



Fig. 9. System identification of nonlinear plant: (a) in the absence of disturbance (b) in the presence of disturbance. The dashed line is the plant output, and the solid line is the output from the plant model.

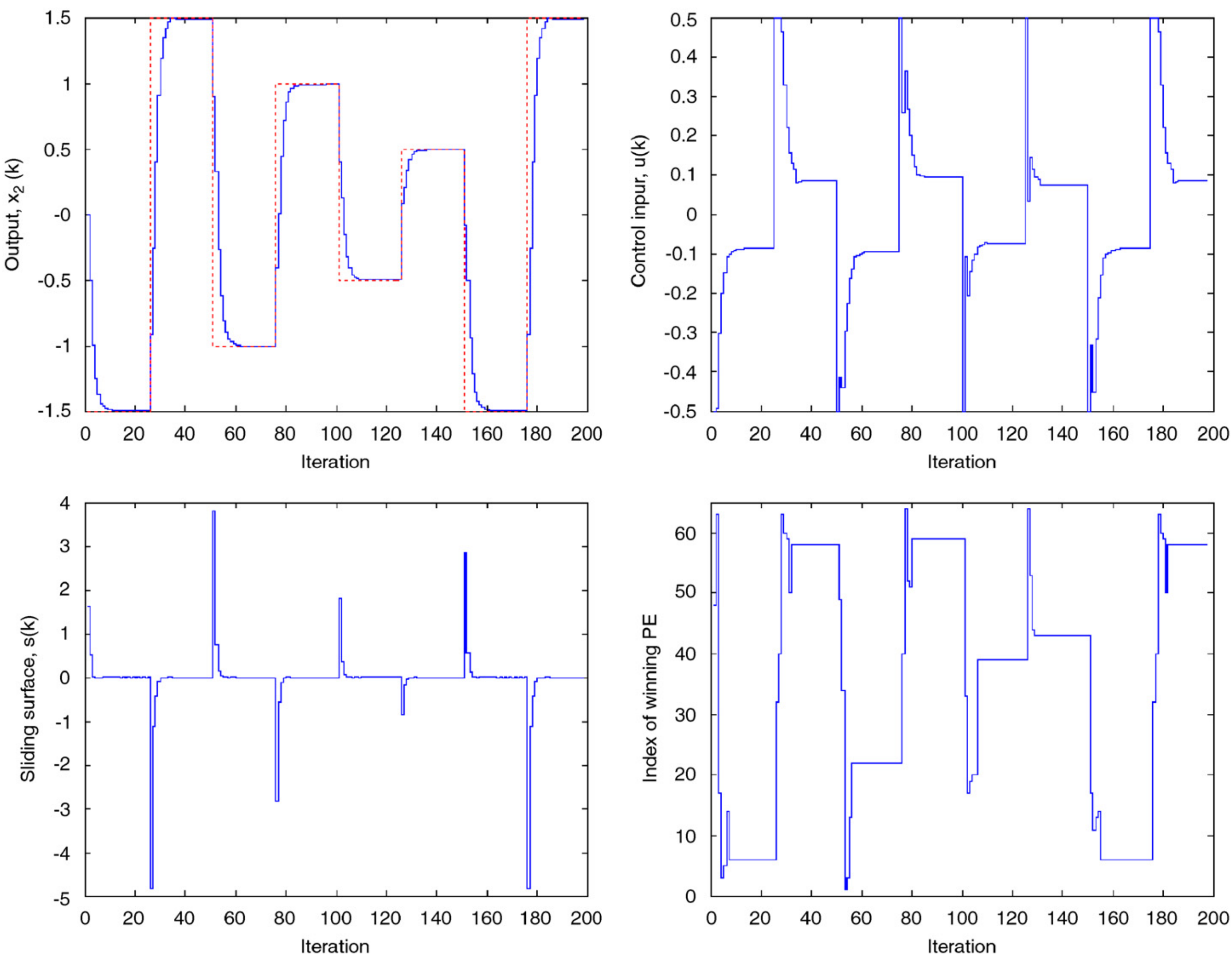



Fig. 10. Performance of square-wave tracking in the absence of disturbance.

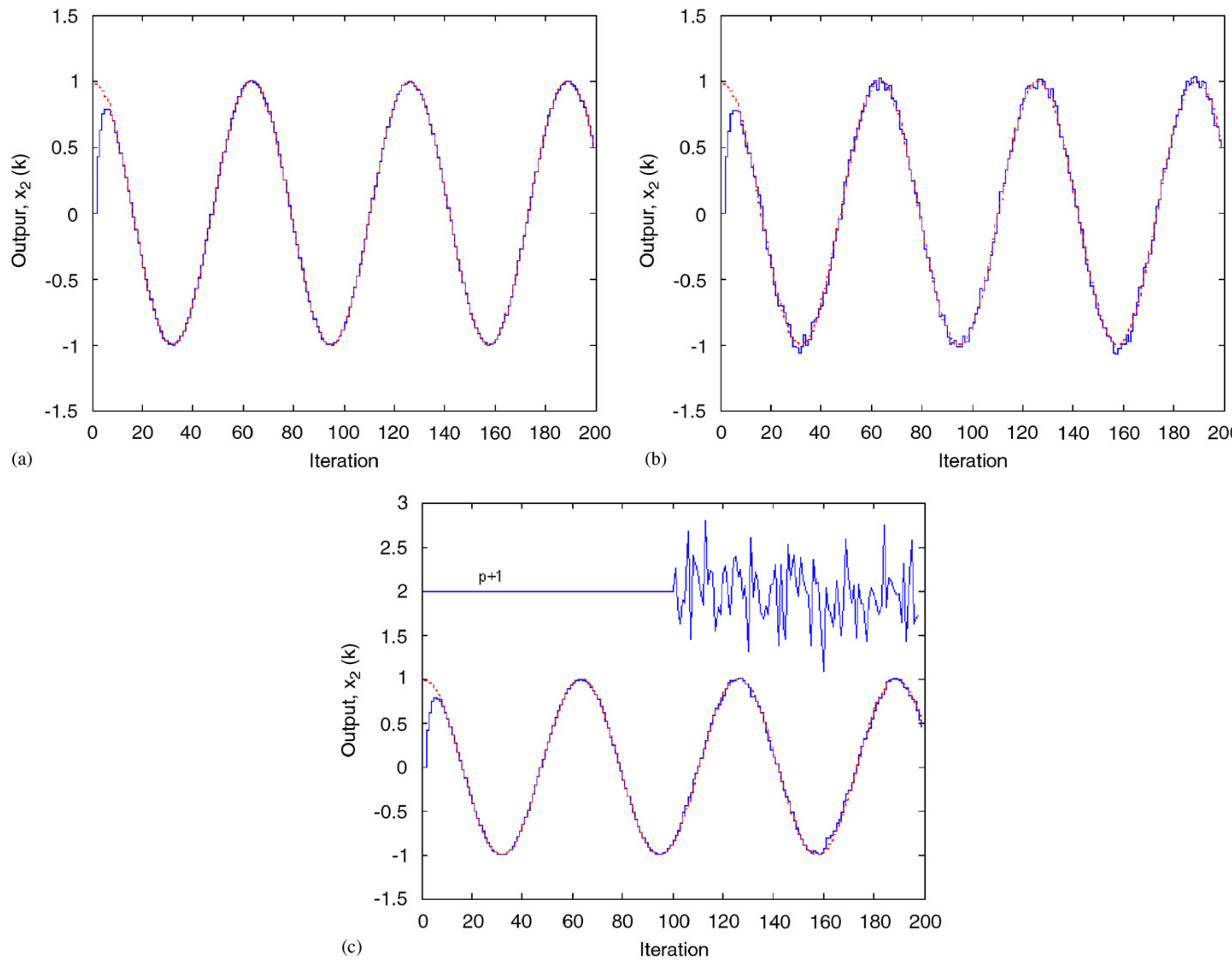



Fig. 11. Sinusoidal trajectory tracking behavior by the MQSMC: (a) in the absence of sensor noise (b) in the absence of sensor noise (c) under parameter variations.

1.4e-2 ± 4.3e-4 and 1.6e-2 ± 4.7e-4, respectively. From this observation, it should be pointed out that the multiple modeling strategy has more noise-immunity feature than global models.

Fig. 10 shows the plant responses of the closed loop control system using the proposed MQSMC choosing the switching surface as $s_k = [1, -2][e_{k-1}, e_k]^T$. The trajectories are seen to converge to the desired values of [−1.5 −1.0 −0.5 0.5 1.0 1.5]. The figure also shows control input, the sliding surface, and the winner activities switched automatically by the SOM. It can be easily seen that the MQSMC guarantees the convergence of the system to the quasi-sliding-mode band around the sliding hyperplane $\vec{c}^T\vec{e}_k = 0$. Furthermore, we tested the closed loop system for tracking a sinusoidal signal under the noisy environment and parameter variations. Once again, the multiple controller networks perfectly track the desired command except for a transient time of few time steps, as shown in Fig. 11, even if not only the measurement is corrupted by zero-mean random noise with 20 dB of *SNR* but the parameter, *p*, varies from 0.5 to 1.5 after 100 iterations.

Next, the robustness of the MQSMC scheme was compared with that of the Multiple Inverse Control (MIC) network proposed in [6], as well as with that of the TDNNC built based on the TDNN model.[7] The standard deviation of the error between the plant output and the desired output versus the standard deviation of the noise is shown in Fig. 12. It is evident that the proposed approach performs best in terms of insensitivity to disturbances. The MIC structure showed the best performance only in the noise-free environment. It should be noted that the MIC began to become less robust than the

[7]The TDNN controller was trained by back-propagating an error through the TDNN model taught by 20 hidden PEs and the number of hidden PEs in the controller was chosen as 40.

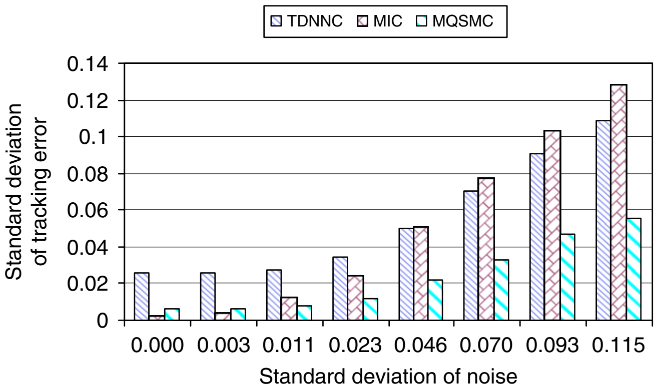



Fig. 12. Comparison of performance against noise among AIC, MIC, and MQSMC.

TDNNC at the point where the standard deviation of the noise is over 0.03. From this examination we can conclude that the MIC can be robust against noise up to certain level. However, wrong selection of the winner, as the amount of noise is increased, can be devastating for the controller that is designed based on the predicted model. Overall, we conclude that the proposed MQSMC approach is the most robust design technique among the three methods considered. This is evident from Fig. 12, where we observe that on average the tracking error of MQSMC increases at a lower rate than that of the MIC and the TDNNC.

## 5. Conclusion

In this paper, the MQSMC strategy has been proposed for a general class of nonlinear unknown discrete-time systems via SOM, which divides the state space into a set of operating regions. Contrary to what is assumed in the field of sliding mode controller design, the plant dynamics under control are assumed to be unknown. This is a challenge in the conventional design framework with the ambiguities introduced by the noise on the measured quantities. Thus, we have taken the concept of self-organization in the embedded output space extended with multiple models, so that a local controller is employed for each region. The problems that arise due to the uncertainties of the plant model and measurement noise are alleviated by incorporating the robustness provided by the sliding mode technique into the multiple modeling approach. The simulation results show that the algorithm proposed is able to compensate deficiencies caused by the imperfect observations of the state variables and complex plant dynamics, driving the tracking error vector to the sliding manifold and keeping it on the manifold. In addition, the proposed method shows better robustness against noise, faster transient response, and better steady-state accuracy of the controlled system by switching local controllers astutely through the SOM, than other neural network-based alternatives.

In addition, it has been shown that the multiple sliding mode control scheme guarantees BIBO stability of the overall system. However, a possible disadvantage of the proposed approach is that the overall stability may not be guaranteed due to the switching among models if the models are quite different each other. Further studies, therefore, will be necessary to investigate the stability conditions for such a switched control systems to achieve better control performance.

## Acknowledgment

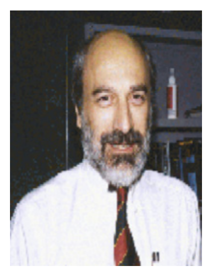
This work was partially supported by NASA grant NAG-1-02068.

## References

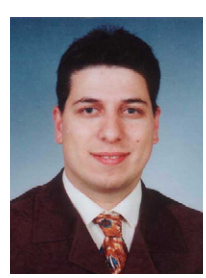
[1] A. Bartoszewicz, Discrete-time quasi-sliding-mode control strategies, IEEE Trans. Ind. Electron. 4 (1998) 633–637.
[2] J. Camps, F.L. Lewis, R. Selmic, Backlash compensation with filtered prediction in discrete time nonlinear systems by dynamic inversion using neural networks, in: Proceedings of the IEEE Conference on Decision and Control, Sydney, Australia, 2000, pp. 3354–3540.
[3] C.Y. Chan, Robust discrete quasi-sliding mode tracking controller, Automatica 10 (1995) 1509–1511.
[4] L. Chen, K.S. Narendra, Intelligent control using neural networks and multiple models, in: Proceedings of the American Control Conference, AK, USA, 2002, pp. 1357–1362.
[5] X. Chen, T. Fukuda, K.D. Young, Adaptive quasi-sliding-mode tracking control for discrete uncertain input–output systems, IEEE Trans. Ind. Electron. 1 (2001) 216–224.
[6] J. Cho, J.C. Principe, D. Erdogmus, M.A. Motter, Modeling and inverse controller design for an Unmanned Aerial Vehicle based on the self-organizing map, IEEE Trans. Neural Networks 16 (2006) 445–460.
[7] Y. Diao, K.M. Passino, Intelligent fault tolerant control using adaptive schemes and multiple models, in: Proceedings of the American Control Conference, VA, USA, 2001, pp. 2854–2859.
[8] J.D. Farmer, J.J. Sidorowich, Predicting chaotic time series, Phys. Rev. Lett. 8 (1987) 845–848.
[9] B.A. Foss, I.A. Johansen, A.V. Sorensen, Nonlinear predictive control using local models—applied to a batch fermentation process, Control Eng. Practice 3 (1995) 389–396.
[10] K. Furuta, Sliding mode control of a discrete time systems, Syst. Control Lett. 14 (1990) 145–152.
[11] K. Furuta, VSS type self-tuning control, IEEE Trans. Ind. Electron. 1 (1993) 37–43.
[12] W. Gao, Y. Wang, A. Homaifa, Discrete-time variable structure control systems, IEEE Trans. Ind. Electron. 2 (1995) 117–122.
[13] J.P. Gauthier, H. Hammouri, S. Othman, A simple observer for nonlinear systems applications to bioreactors, IEEE Trans. Autom. Control 6 (1992) 875–880.
[14] P.J. Gawthrop, E. Ronco, Local model networks and self-tuning predictive control, Technical Report CSC-96001, University of Glasgow, UK, 1996.
[15] X. He, H. Asada, A new method for identifying orders of input–output models for nonlinear dynamic systems, in: Proceedings of the American Control Conference, CA, USA, 1993, pp. 2520–2523.
[16] A. Ishigame, T. Furukawa, S. Kawamoto, T. Taniguchi, Sliding mode controller design based on fuzzy inference for nonlinear systems, IEEE Trans. Ind. Electron. 1 (1993) 64–70.
[17] T.A. Johansen, B.A. Foss, Constructing NARMAX models using ARMAX models, Int. J. Control 5 (1993) 1125–1153.
[18] T. Kohonen, Self-Organizing Maps, Springer, Berlin, 1995.

[19] P.M. Lee, J.H. Oh, Improvements on VSS-type self-tuning control for a tracking controller, IEEE Trans. Ind. Electron. 2 (1998) 319–325.
[20] C. Milosavljevic, General conditions for the existence of a quasi-sliding mode on the switching hyperplane in discrete variable structure systems, Autom. Remote Control 3 (1985) 307–314.
[21] R. Murray-Smith, K.J. Hunt, Local model architectures for nonlinear modelling and control, in: K.J. Hunt, G.R. Irwin, K. Warwick (Eds.), Neural Network Engineering in Dynamic Control Systems, Springer, Berlin, 1995, pp. 61–82.
[22] R. Murray-Smith, T.A. Johansen, Multiple Model Approaches to Modeling and Control, Taylor & Francis Inc, London, 1997.
[23] K.S. Narendra, J. Baakrishnan, Adaptive control using multiple models, IEEE Trans. Autom. Control 2 (1997) 171–187.
[24] K.S. Narendra, K. Parthasarathy, Identification and control of dynamical systems using neural networks, IEEE Trans. Neural Networks 1 (1990) 4–27.
[25] K.S. Narendra, C. Xiang, Adaptive control of discrete-time systems using multiple models, IEEE Trans. Autom. Control 9 (2000) 1669–1686.
[26] K.S. Narendra, J. Balakrishnan, M.K. Ciliz, Adaptation and learning using multiple models, switching, and tuning, IEEE Trans. Control Systems Mag. 3 (1995) 37–51.
[27] O. Nelles, Nonlinear System Identification, Springer, Berlin, 2001.
[28] A. Packard, Gain scheduling via linear fractional transformations, Syst. Control Lett. 1 (1994) 79–92.
[29] H.A. Palizban, A.A. Safavi, J.A. Romagnoli, A nonlinear control design approach based on multi-linear models, in: Proceedings of the American Control Conference, NM, USA, 1997, pp. 3490–3494.
[30] J.K. Pieper, K.R. Goheen, Discrete-time sliding mode control via input–output models, in: Proceedings of the American Control Conference, CA, USA, 1993, pp. 964–965.
[31] J.C. Principe, L. Wang, Nonlinear time series modeling with self-organizing feature maps, in: Proceedings of the IEEE Workshop on Neural Networks for Signal Processing, MA, USA, 1995, pp. 11–20.
[32] J.C. Principe, L. Wang, M.A. Motter, Local dynamic modeling with self-organizing maps and applications to nonlinear system identification and control, Proc. IEEE 11 (1998) 2240–2258.
[33] W.J. Rugh, Analytical framework for gain scheduling, IEEE Trans. Control Syst. Mag. 1 (1991) 79–84.
[34] A.I. Santos, T.A. Johansen, J.M. Cosme, Nonlinear multiple model predictive control in a fed-batch reactor, in: Proceedings of the IFAC Symposium on Artificial Intelligence in real-time control, Budapest, Hungary, 2000.
[35] S.Z. Sarpturk, Y. Istefanopulos, O. Kaynak, On the stability of discrete-time sliding mode control systems, IEEE Trans. Autom. Control 10 (1987) 930–932.
[36] D. Sbarbaro, R. Murray-Smith, Self-tuning control of non-linear systems using Gaussian process prior models, DCS Technical Report TR-2003-143, University of Glasgow, UK, 2003.
[37] J.S. Shamma, M. Athans, Gain scheduling: potential hazards and possible remedies, IEEE Trans. Control Syst. Mag. 3 (1992) 101–107.
[38] A.C. Singer, G.W. Wornell, A.V. Oppenheim, Codebook prediction: a nonlinear signal modeling paradigm, in: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, CA, USA, 1992, pp. 325–328.
[39] H. Sira-Ramirez, Non-linear discrete variable structure systems in quasi-sliding mode, Int. J. Control 5 (1991) 1171–1187.
[40] G. Thampi, J.C. Principe, J. Cho, M.A. Motter, Adaptive inverse control using SOM based multiple models, in: Proceedings of Portuguese Conference on Automatic Control, Aveiro, Portugal, 2002a, pp. 278–282.
[41] G. Thampi, J.C. Principe, M.A. Motter, J. Cho, J. Lan, Multiple model based flight control design, in: Proceedings of the 45th Midwest Symposium on Circuits and Systems Conference, OK, USA, 2002b, pp. 133–136.
[42] J. Walter, H. Ritter, K. Schulten, Nonlinear prediction with self-organizing maps, in: Proceedings of the International Joint Conference on Neural Networks, CA, USA, 1990, pp. 589–594.
[43] J. Wang, W.J. Rugh, Feedback linearization families for nonlinear systems, IEEE Trans. Autom. Control 10 (1987) 935–940.
[44] D.H. Nguyen, B. Widrow, Neural networks for self-learning control systems, IEEE Trans. Control Syst. Mag. 10 (1990) 18–23.

**Jeongho Cho** received the B.S. degree in control and instrumentation engineering from the Soonchunhyang University, Korea, in 1995 and the M.S. degree in electrical engineering from Dongguk University, Korea, in 1997 and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Florida, Gainesville, in 2001 and 2004, respectively. From 2005 to 2006, he worked as a postdoctoral research associate in the department of biomedical engineering at the University of Florida, concentrating on biomedical signal modeling and on EEG signal analysis for the control of epileptic seizure. Since 2006, he has been a senior engineer in Samsung Electronics where he is working on developing multi-functional printers. His research interests include time-series prediction and nonlinear system identification, with applications to navigation and control.



**Jose C. Principe** is currently a Distinguished Professor of electrical and biomedical engineering at the University of Florida, Gainesville, since 2002. He joined the University of Florida in 1987, after an eight-year appointment as Professor at the University of Aveiro, in Portugal. He holds degrees in electrical engineering from the University of Porto (B.S.), Portugal, University of Florida (M.S. and Ph.D.), USA and a Laurea Honoris Causa degree from the Universita Mediterranea in Reggio Calabria, Italy. His interests lie in nonlinear non-gaussian optimal signal processing and modeling in biomedical engineering. He created the Computational NeuroEngineering Laboratory in 1991 to synergistically focus the research in biological information processing models. Dr. Principe is a Fellow of the IEEE, past President of the International Neural Network Society, and Editor in Chief of the Transactions of Biomedical Engineering since 2001, as well as a former member of the Advisory Science Board of the FDA. He holds five patents and has submitted seven more. Dr. Principe was supervisory committee chair of 47 Ph.D. and 61 Master students, and he is author of more than 400 refereed publications (3 books, 4 edited books, 14 book chapters, 116 journal papers and 276 conference proceedings).



**Deniz Erdogmus** received the B.S. in Electrical & Electronics Engineering (EEE), and the B.S. in Mathematics both in 1997, and the M.S. in EEE in 1999 from the Middle East Technical University, Turkey. He received his Ph.D. in Electrical & Computer Engineering from the University of Florida, Gainesville in 2002. He worked as a research engineer at TUBITAK-SAGE, Turkey from 1997 to 1999, focusing on the design of navigation, guidance, and flight control systems. He was also a research assistant and a postdoctoral research associate at UF from 1999 to 2004, concentrating on signal processing, adaptive systems, machine learning, and information theory, specifically with applications in biomedical engineering. Currently, he is holding an Assistant Professor position jointly at the Computer Science and Electrical Engineering Department and the Biomedical Engineering Department of the Oregon Health and Science University. Dr. Erdogmus has over 35 articles in international scientific journals and numerous conference papers and book chapters. He has also served as associate editor and guest editor for various journals. He is a member of Tau Beta Pi, Eta Kappa Nu, and IEEE.

**Mark A. Motter** was born in Columbia, Pennsylvania. He served in the United States Navy from 1973 until 1979, honorably discharged at the rank of Electronics Technician First Class. He then began his formal engineering education at Old Dominion University in Norfolk, Virginia, receiving his BSEE, magna cum laude, and MSEE, in 1983 and 1985, respectively. He received Ph.D. in Electrical and Computer Engineering from the University of Florida in 1998. Since 1985 Dr. Motter has been employed at NASA Langley Research Center. Currently, he is a controls research engineer in the Electronics Systems Branch. His current research project is investigating the implementation of self-organizing and other biologically inspired flight control approaches, using fully autonomous unmanned aerial vehicles. He is a senior member of the IEEE, a registered Professional Engineer, and a member of the Academy of Model Aeronautics.