Sliding-Mode Neuro-Controller for Uncertain Systems

Yildiray Yildiz, *Student Member, IEEE*, Asif Šabanovic, *Senior Member, IEEE*, and Khalid Abidi

***Abstract*—In this paper, a method that allows for the merger of the good features of sliding-mode control and neural network (NN) design is presented. Design is performed by applying an NN to minimize the cost function that is selected to depend on the distance from the sliding-mode manifold, thus providing that the NN controller enforces sliding-mode motion in a closed-loop system. It has been proven that the selected cost function has no local minima in controller parameter space, so under certain conditions, selection of the NN weights guarantees that the global minimum is reached, and then the sliding-mode conditions are satisfied; thus, closed-loop motion is robust against parameter changes and disturbances. For controller design, the system states and the nominal value of the control input matrix are used. The design for both multiple-input–multiple-output and single-input–single-output systems is discussed. Due to the structure of the (M)ADALINE network used in control calculation, the proposed algorithm can also be interpreted as a sliding-mode-based control parameter adaptation scheme. The controller performance is verified by experimental results.**

***Index Terms*—Neural networks (NNs), sliding-mode control (SMC).**

I. Introduction

DUE to its robustness to parameter uncertainties and external disturbances, sliding mode is a well-established control method for application in nonlinear systems [1]–[3]. Merging a well-established control structure like sliding-mode control (SMC) with neural-network (NN)-based algorithms appeared to be a good idea, and many researchers have published various control structures based on this idea. A comprehensive historical investigation and a literature survey can be found in [4]. In the application of SMC methods to NN-based control, a few main ideas seem to be prevailing. The first one attempts to apply NN as an observer in the estimation of equivalent control [5] and in some cases disturbances [6]. Such an application of NN leads to effective linearization of the system and thus allows simpler design of the main controller. The weights of the NN are determined based on the evaluation of the distance from the sliding-mode manifold. Good results in application to nonlinear systems linear with respect to (w.r.t.) control are reported.

In [6], Jezernic *et al.* applied the idea on a 3.D.O.F PUMA-type DD robot system. They used continuous sliding-mode theory to establish a robust control scheme. To avoid the chattering effect, they estimated the equivalent control and used this estimation in the SMC algorithm. The estimation of the equivalent control was done using an online NN estimator. In [7], Rodic *et al.* used a sliding-mode-based learning algorithm for the robust accurate tracking of a single-axis DD robotic manipulator driven with an induction motor. In another work [8], Fang *et al.* proposed a control system on the basis of a discrete Lyapunov function. Part of the equivalent control is estimated by a recurrent NN (RNN), and a real-time iterative learning algorithm is developed and used to train the RNN. They also proved the stability of the system by showing that the learning error converges to zero.

The adoption of a nonlinear dynamic adjustment strategy in an ADALINE-based controller applied to a three degree-of-freedom robot is discussed in [9]. The idea is to impose an SMC while an adaptation is imposed on the controller parameters in such a way that the desired motion is achieved. A proportional–derivative (PD) controller with a bias term is applied so three parameters are to be adjusted. The rate of change of the parameters is discontinuous (sign function), which are approximated by the boundary layer continuous high gain. The sliding-mode manifold is selected to be a difference between desired (unknown) and applied torque; thus, another functional mapping is needed in order to use the available sliding-mode function expressed as a linear combination of position and velocity error. A similar idea is explored in [10]–[12] for a class of nonlinear systems.

Also, in [16], a sliding-mode controller with a modified switching function that produces a low-chattering control is used in parallel with an artificial NN for online identification of the modeling error, which improves the controller performance. In [17], a novel approach that combines SMC and NN control is presented, the weights of which are determined by a fuzzy supervisory controller. The authors call this approach as fuzzy supervisory sliding-mode and NN control. Fuzzy NN sliding-mode controllers were also proposed in [18] and [19].

In this paper, the proposed controller is based on the minimization of a cost function that is obtained by satisfying the Lyapunov stability criteria. The cost function is selected in such a way that its time derivative is an explicit function of control, thus allowing calculation of the control input. This cost function is the same cost function used in [5], but different from

Manuscript received December 11, 2004; revised January 3, 2007. This work was supported in part by Sabanci University, Istanbul, Turkey.

Y. Yildiz is with the Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: yildiray@mit.edu).

A. Šabanovic is with Faculty of Engineering and Natural Sciences (FENS), Sabanci University, Istanbul 34956, Turkey (e-mail: asif@sabanciuniv.edu).

K. Abidi is with the National University of Singapore, Singapore 117576 (e-mail: g0403709@nus.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2007.894719

their approach, the aim is not calculating the equivalent control but computing the whole control signal using the minimization process. Also, the NN used is a one-layer NN that holds the linearity of parameters. Due to the structure of the (M)ADALINE network used in the control calculation, the proposed algorithm can also be interpreted as a sliding-mode-based control parameter adaptation scheme. To verify the performance of the control scheme, two different experimental setups are used. The first experimental setup is a single-axis linear drive that is driven by a dc motor. This setup is used for the implementation of the controller designed for single-input–single-output (SISO) systems. The other setup consists of two piezo stack actuators and used for the controller designed for multiple-input–multiple-output (MIMO) systems. The proposed control schemes performed well in both of the experiments.

II. Problem Statement

In this paper, we will consider dynamical systems consisting of m interconnected subsystems described by $y_i^{n_i-1} = h_i + b_i u_i + g_i$, where $y_i^{l_i}$ stands for the lth time derivative of y_i. By selecting the state vector $x = [y_1, \dot{y}_1, \ldots, y_1^{n_i-1}, \ldots, y_m, \dot{y}_m, \ldots, y_m^{n_i-1}]^T$, these interconnected subsystems can be represented as a class of nonlinear systems linear w.r.t. control as

$$\dot{x} = f(x) + B(x)u + d \tag{1}$$

where $x^T \in \Re^n$ is the state vector $n = \sum_{i=1}^m n_i$; $u \in \Re^m$ is the control vector; $f(x) \in \Re^n$ is an unknown, continuous, and bounded nonlinear function; $B(x) \in \Re^{nxm}$ is a known input matrix whose elements are continuous and bounded; and $rank(B(x))|_{\forall x} = m$, with $d \in \Re^n$ being an unknown bounded external disturbance. Both $f(x) \in \Re^n$ and $d \in \Re^n$ satisfy the matching conditions, and all their components are bounded $\|f_i(x)\|_{\forall x} \leq M$ and $\|d_i(t)\|_{\forall t} \leq N$. Fully actuated mechanical systems belong to the class of systems described by (1). Such systems can be interpreted as m interconnected subsystems $\ddot{q}_i = h_i(q_i, \dot{q}_i) + b_i(q_i, t)u_i + g_i(q_i, q_j)$, where $h_i(q_i, \dot{q}_i)$ in general represents the Coulomb friction term, and $g_i(q_i, q_j)$ represents the interaction term and is regarded as a disturbance.

The aim is to determine the control input $u = [u_1, \ldots, u_m]^T$ such that the outputs of the system $y_1(t), \ldots, y_m(t)$ track the desired trajectories $y_{d_1}(t), \ldots, y_{d_m}(t)$ while the control error satisfies the selected dynamical constraints.

III. Controller Design

The controller will be designed in the SMC framework by first selecting a suitable sliding manifold that will ensure desired systems dynamics and then selecting control such that Lyapunov stability conditions are satisfied. Selecting the Lyapunov function candidate in terms of the sliding function is a natural way of guaranteeing the sliding-mode existence on the selected manifold and thus having desired closed-loop dynamics. Finally, the necessary control input that will fulfill the requirements of the Lyapunov stability criteria should be selected.

A. Sliding Manifold

For system (1), the natural selection of the sliding manifold is in the form

$$\sigma = Ge_t = 0 \tag{2}$$

where the tracking error vector is defined as $e_t = [e_1, \ldots, e_1^{(n_1-1)}, \ldots, e_m, \ldots e_m^{(n_m-1)}]^T \in \Re^n$, $e_i = y_{d_i} - y_i$. $\sigma = [\sigma_i, \ldots, \sigma_m]^T \in \Re^m$ $G \in \Re^{mxn}$. Matrix G is selected such that each component of vector $\sigma(e)$ is selected to be a function of one output control error and its derivatives $\sigma_i(e_i) = 0$ having the form $\sigma_i = \sum_{i=1}^{n_i-1} a_i e_i$; $a_i > 0$, $a_{i1} = 1$ with multiple real root being equal to $-C$.

B. Computing the Necessary Control Input

A Lyapunov function candidate can be selected as

$$V = \frac{1}{2}\sigma^T\sigma \tag{3}$$

where $V \in \Re$. This function can also be stated as $V = (1/2)\|\sigma\|_2^2$, where $\| \bullet \|_2$ indicates the Euclidian norm with $V(0) = 0$. The time derivative of the candidate Lyapunov function $\dot{V}$ should be negative definite. In order to use this condition in the selection of control, we may require that $\dot{V}$ satisfy some preselected form. Equating the time derivative of this function to a negative definite function, we have

$$\dot{V} = -\sigma^T D\sigma - \mu\frac{\sigma^T\sigma}{\|\sigma\|_2} \tag{4}$$

where D is a positive definite symmetric matrix, and $\mu > 0$; thus, the Lyapunov conditions are satisfied. By substituting (3) into (4), the following requirement is found:

$$\sigma^T\left(\dot{\sigma} + D\sigma + \mu\frac{\sigma}{\|\sigma\|_2}\right) = 0. \tag{5}$$

Therefore, for $\sigma \neq 0$, the control law can be calculated by satisfying

$$\left(\dot{\sigma} + D\sigma + \mu\frac{\sigma}{\|\sigma\|_2}\right) = 0 \tag{6}$$

and the sliding-mode conditions are satisfied. The discontinuous term can be selected as small in order to avoid chattering. It had been proven in [13] and [14] that in the discrete-time implementation, the sliding mode is guarantied with continuous control action. We are targeting computer controller systems for which the controller will be implemented in discrete time; so in our application, the discontinuous term will be omitted, and we will be determining the control action that satisfies conditions $(\dot{\sigma} + D\sigma) = 0$, but all further analysis can be easily adopted for application of (6) if the term $(D\sigma)$ is replaced with $(D\sigma + \mu\sigma/\sigma^T\sigma)$.

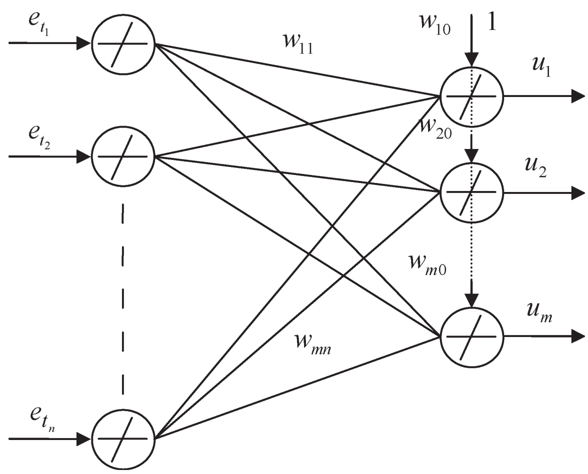


Fig. 1. Structure of the NN.

For system (1) with sliding-mode manifold (2), the control that satisfies $(\dot{\sigma} + D\sigma) = 0$ is determined as

$$u = -(GB)^{-1}\left(G(f + d - \dot{x}_{d_i}) - D\sigma\right) = u_{eq} + (GB)^{-1}D\sigma \tag{7}$$

where $x_d = [y_{d_1}, \ldots, y_{d_1}^{(n_1-1)}, \ldots, y_{d_m}, \ldots, y_{d_m}^{(n_m-1)}]$, and u_{eq} is the so-called equivalent control obtained as a solution of the equation $\dot{\sigma} = 0$. By substituting (7) into (1), the equations of motion of system (1) in manifold (2) are obtained as $\sigma = Ge_t = 0$, and the approach to this solution is governed by (6). This is a result of the specific structure of the plant (1) in which states are selected as the derivatives of measurable outputs, and each subblock is represented in canonical form.

To implement this control input, information about the plant dynamics and external disturbances are needed, which is hard to achieve. Hence, this solution needs the information on the equivalent control and thus may be applied for the plants when u_{eq} is known or can be estimated with sufficient accuracy. The approach in [4]–[6] is based on the application of NN in the estimation of equivalent control.

In this paper, we will take a different approach. Instead of estimating equivalent control and then applying (7), we will apply a least-square minimization using NNs to fulfill $(\dot{\sigma} + D\sigma) = 0$.

C. Structure and Working Principles of NN

The structure of the NN used in this paper is presented in Fig. 1, where e_{t_i} is the ith row of e_t. w_{ij} refers to the weight of the signal that comes from the jth node and goes to the ith node, whereas w_{i0} refers to the bias term of the ith node. Control inputs, which are the outputs of the NN, can be defined as $u = [u_1, \ldots, u_m]$, where

$$u_i = \sum_{j=1}^{n} e_{t_i} w_{ij} + 1w_{i0}, \qquad i = 1, \ldots, m. \tag{8}$$

As seen from (8), in the selected network, the activation functions are linear, and the network is static. In (8), the weights can be treated also as variable coefficients that should be adjusted in order to determine the necessary control. For second-order systems, the structure (8) could be viewed as a PD controller with gain adaptation, and for higher-order systems, it can be viewed as a state feedback controller with adaptation of the gain matrix. As follows from the structure of the network, if the inputs are zero (that means the control error vector is zero—thus, the control objective is reached), the output is equal to the bias vector; thus, the bias weights should compensate the system's disturbance.

In this paper, we will demonstrate the selection of weights in (8) so that for system (1), the requirements $(\dot{\sigma} + D\sigma) = 0$ determined from the Lyapunov stability conditions are satisfied. In order to fulfill the above requirements, we will apply the NN that will minimize the error function

$$E = \frac{1}{2}(\dot{\sigma} + D\sigma)^T(\dot{\sigma} + D\sigma). \tag{9}$$

By selecting the weights such that $E \to 0$ and that $E = 0$ is a stable solution, the condition $(\dot{\sigma} + D\sigma) = 0$ will be satisfied, and the stable sliding-mode motion will be achieved in manifold (2). The selected error function depends on the control input, and this allows to take a partial derivative of the error function w.r.t. control. Due to the fact that, for the selected structure of NN, the control is linearly dependent on weights, the usual weight update is expected to give a simple structure. In addition, the selected error function does not depend on unknown variables, so it can be evaluated online, and it does not need offline training.

1) Weight Updates: Weights are updated according to

$$\dot{w}_{ij} = -\eta \frac{\partial E}{\partial w_{ij}} \tag{10}$$

where $\eta > 0$ is the learning constant. Using the chain rule, (10) can be written as

$$\dot{w}_{ij} = -\eta \frac{\partial E}{\partial u_i} \frac{\partial u_i}{\partial w_{ij}}. \tag{11}$$

Substituting (9) into (11) and taking the derivatives, the following equation is obtained:

$$\dot{w}_{ij} = -\eta(\dot{\sigma} + D\sigma)^T \frac{\partial \dot{\sigma}}{\partial u_i} e_{t_j}. \tag{12}$$

Substituting (2) into (12)

$$\dot{w}_{ij} = -\eta(\dot{\sigma} + D\sigma)^T \frac{\partial(G\dot{x}_d - G\dot{x})}{du_i} e_{t_j} \tag{13}$$

is obtained. Rewriting (1) as

$$\dot{x} = f(x) + \left[B_1(x) \vdots \ldots \vdots B_m(x)\right] \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} + d \tag{14}$$

and substituting (14) into (13) and taking the derivative gives

$$\dot{w}_{ij} = \eta(\dot{\sigma} + D\sigma)^T GB_i(x) e_{t_j} \tag{15}$$

where $B_i(x)$ is the ith column of the matrix $B(x)$. For the bias terms w_{i0}, the weight update can be computed using the same procedure as

$$\dot{w}_{i0} = \eta(\dot{\sigma} + D\sigma)^T GB_i(x). \tag{16}$$

The weight update (15) is simple and depends on the fulfillment of the conditions $(\dot{\sigma} + D\sigma) = 0$. The weights will change as long as condition (6) is not satisfied. Both (15) and (16) depend on the plant gain matrix and the selected sliding-mode manifold (2).

The reason for the introduction of the bias term (16) can be easily seen from the analysis of the error function (9) rewritten as $E = (1/2)[G(\dot{x}_d - f(x) - B\sum_{i=1}^{n} w_i e^{(i-1)} - d) + DGe_t]^2$, where it is obvious that if all control errors are going to zero, the error function will tend to be a finite value $E|_{e=0} = (1/2)[G(\dot{x}_d - f(x) - d)]^2$. By introducing the bias term, the error function becomes $E = (1/2)[G(\dot{x}_d - f(x) - B\sum_{i=1}^{n} w_i e^{(i-1)} \ + B1w_{n+1} - d) + DGe_t]^2$, and terms that satisfy the matching conditions can be rejected so the error function will have a minimum at point $E = 0$. After reaching $E = 0$, the bias term will asymptotically tend to the equivalent control $w_{n+1} \rightarrow -(GB)^{-1}(G(f + d - \dot{x}_d))$, thus allowing the compensation of the matching disturbances. Due to the assumption that the components of vectors f, d, and $\dot{x}_d$ are bounded, the bias term will be also bounded; thus, for bounded initial errors, the control input will also be bounded.

The selection of the linear activation function is not essential to the solution. Single-valued continuous activation functions may be applied. For example, if the model of the NN is described as $n_i = \sum_{j=1}^{n} e_{t_j} w_{ij} + 1w_{i0}$, $i = 1, \ldots, m$, with activation function $u_i = g_i(n_i)$, then in (15), a multiplying term $g_i' = \partial g_i(n_i)/\partial n_i$ will appear so we will have $\dot{w}_{ij} = \eta(\dot{\sigma} + D\sigma)^T g_i' GB_i(x)e_{t_j}$. Proper selection of g_i' as a single-valued positive definite function will preserve the validity of the proof given in the text.

This solution provides the rate of change of the NN weights as a function of the distance from the desired solution $(\dot{\sigma} + D\sigma) = 0$, and at the moment $(\dot{\sigma} + D\sigma) = 0$ is reached, the weights are not further updated while the motion of the system reaches the sliding-mode manifold according to $(\dot{\sigma} + D\sigma) = 0$. As a result, the control determined by (7) at the moment the sliding-mode manifold is reached is equal to the equivalent control, and thus the sliding-mode motion in manifold (2) is enforced.

D. Local Minimum Issue

One of the biggest problems in backpropagation weight update algorithm is that the system may not reach global minimum and may stay in some local minima. Investigating the shape of the error function (9), it can be shown that the local minima do not exist for the selected formulation of the minimization problem in controller parameter space.

1) Shape of the Error Surface: If a function's second derivative w.r.t. a function variable does not change sign, then the function does not have a change in the curvature sign through that variable, which means that the function does not have a local minimum through that variable. Taking the second derivative of the error function (9) w.r.t. the weight w_{ij} gives

$$\frac{\partial^2 E}{\partial w_{ij}^2} = -\eta\left(\frac{\partial(\dot{\sigma} + D\sigma)^T}{\partial w_{ij}}\right) GB_i(x)e_{t_j}. \tag{17}$$

Using the chain rule in derivation

$$\frac{\partial^2 E}{\partial w_{ij}^2} = -\eta\left(\frac{\partial(\dot{\sigma} + D\sigma)^T}{\partial u_i}\frac{\partial u_i}{\partial w_{ij}}\right) GB_i(x)e_{t_j} \tag{18}$$

and substituting (2) and (8) into (18), the following equation is obtained:

$$\frac{\partial^2 E}{\partial w_{ij}^2} = -\eta\left(\frac{\partial(G\dot{x}_d - G\dot{x})^T}{\partial u_i}\right) GB_i(x)e_{t_j}^2. \tag{19}$$

Substituting (1) into (19) gives

$$\frac{\partial^2 E}{\partial w_{ij}^2} = \eta\left(\frac{\partial\left(f(x)^T G^T + u^T B(x)^T G^T + d^T G^T\right)}{\partial u_i}\right) GB_i(x)e_{t_j}^2. \tag{20}$$

Taking the derivative

$$\frac{\partial^2 E}{\partial w_{ij}^2} = \eta B_i(x)^T G^T GB_i(x)e_{t_i}^2 = \eta\left\|GB_i(x)\right\|_2^2 e_{t_j}^2 \tag{21}$$

is obtained. Using the same procedure, the second derivative of the error function (9) w.r.t. the bias weights is computed as

$$\frac{\partial^2 E}{\partial w_{i0}^2} = \eta B_i(x)^T G^T GB_i(x) = \eta\left\|GB_i(x)\right\|_2^2. \tag{22}$$

Equations (21) and (22) show that the sign of the curvature of the error surface (9) is always positive; hence, there are no local minima, which indicates that there is no danger of sticking to local minima in controller parameter space. Also, in the case of reaching the global minimum, since η is a constant, G is a constant matrix, and $B_i(x)$ is bounded, the weight update algorithms (15) and (16) show that weights converge to a finite value. A finite value for the weights in steady-state results in a bounded control input (8). As a result, all the signals in the control system are bounded.

E. Stability

Let the Lyapunov function candidate be the same function that is used for the cost function

$$V = \frac{1}{2}(\dot{\sigma} + D\sigma)^T(\dot{\sigma} + D\sigma). \tag{23}$$

It is easily seen that $V > 0$ for $\dot{\sigma} + D\sigma \neq 0$ and $V = 0$ for $\dot{\sigma} + D\sigma = 0$. Taking the time derivative of V, one obtains

$$\dot{V} = -\sum_{i=1}^{m}\sum_{j=0}^{n}\frac{\partial V}{\partial w_{ij}}\frac{dw_{ij}}{dt} + g(\gamma)\dot{\gamma} \tag{24}$$

where, $g(\gamma)$ represents the derivative of V w.r.t. the variables other than the controller parameters. Substituting (10) into (24) and using the identity $E = V$ gives

$$\dot{V} = -\eta \sum_{i=1}^{m} \sum_{j=0}^{n} \left(\frac{\partial V}{\partial w_{ij}} \right)^2 + g(\gamma)\dot{\gamma}. \tag{25}$$

It is seen that for the system to be stable, the learning gain must be large enough to make the derivative of the Lyapunov function negative definite. This requirement holds for the cases indicated in the simulation and experimental results. However, in real applications, increasing the learning rate too much can be harmful. A more rigorous mathematical analysis clearly indicating the necessary assumptions for the above requirement to hold is under investigation.

IV. Special Case: SISO Systems

Since the MIMO system described in the previous section consists of a cluster of SISO systems, converting the above results to SISO systems is straightforward. In this section, first, the modified problem formulation for the SISO case is given, and then related results are presented directly without derivations.

A. Problem Statement

Consider the class of nonlinear systems described by the differential equation

$$\dot{x} = f(x) + B(x)u + d \tag{26}$$

where $x = [y, \ldots, y^{(n-1)}]^T \in \Re^n$ is the state vector; $y \in \Re$ is the system output; $u \in \Re^1$ is the control vector; $f(x) \in \Re^n$ is an unknown, continuous, and bounded nonlinear function; $B(x) \in \Re^1$ is a known input gain coefficient; and $d \in \Re$ is an unknown bounded external disturbance. Also, $y^{(n)} = d^n y/dt^n$. It is assumed that the system is controllable. The aim is to compute the control action u such that the output of the system y tracks the desired trajectory y_d, while the desired state vector is defined as $x_d = [y_d, \ldots y_d^{(n-1)}]^T$. The tracking error vector is defined as $e_t = [e, \ldots, e^{(n-1)}]^T \in \Re^n$, where, $e = y_d - y$.

B. Structure of the NN and Weight Updates

Again, NN minimizes $\dot{\sigma} + D\sigma$, where $\sigma = Ge_t$ is the sliding function, and $G^T \in \Re^n$. The sliding manifold is defined as $\sigma = (d/dt + C)^{n-1} e = 0$. The NN used is presented in Fig. 2.

This type of NN is called "adaptive linear element" (ADALINE). As seen from the figure, the output of the network [which is the control input u of the system (26)] is the weighted sum of the inputs, which are the individual elements of the tracking error vector, and the bias term. Using the same procedure as in the MIMO case, weight updates can be computed as

$$\dot{w}_i = \eta(\dot{\sigma} + D\sigma)GB(x)e_{t_i}, \qquad i = 1, \ldots, n \tag{27}$$

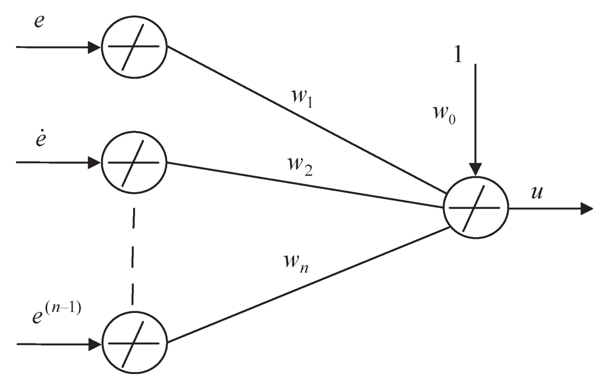


Fig. 2. Structure of the NN for SISO systems.

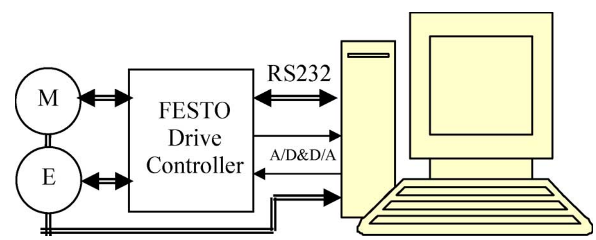


Fig. 3. Simplified structure of the experimental setup.

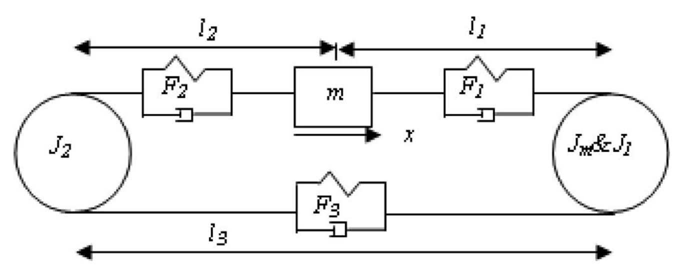


Fig. 4. Physical model of the linear servo drive.

where e_{t_i} is the ith row of the error vector e_t. For the bias term, the weight update takes the form of

$$\dot{w}_0 = \eta(\dot{\sigma} + D\sigma)GB(x)e. \tag{28}$$

V. Experimental Results

A. Experiments With a Single-Axis Linear Servo Drive—A SISO System

The experiments to verify the theory for SISO systems are carried with a single-axis toothed-belt linear servo system, which is now in use at the Mechatronics Laboratory, Sabanci University. The experimental setup scheme is presented in Fig. 3, where "M" and "E" refer to motor and encoder, respectively. This DGEL25-1500-ZR-KF linear drive is equipped with an electrical servo motor MTR-AC-70-3S-AA with a motor driver attached to a dSPACE DS1103 module hosted in the PC with dSPACE software Control Desk v.2.0 and the MATLAB 6.0.0.88.R12. The belt attached to the motor can carry different loads by a carriage. The load carried by the belt, the friction forces between the carriage and the rail, and the friction on the motor bearings affect the motor as disturbance. A physical model of the overall system is presented in Fig. 4.

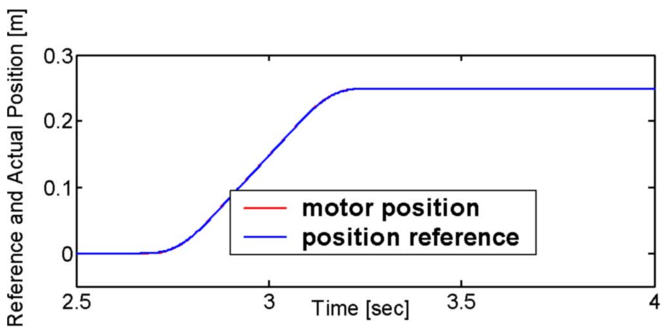


Fig. 5. Position tracking of the motor.

In Fig. 4, J_m, J_1, and J_2 refer to the inertia of the motor, the pulley that is driven by the motor, and the idle pulley, while m refers to the mass of the load that is attached to the belt. The belt is modeled as a spring and damper system so that the overall system is of the fourth order with scalar input—represented by the torque developed by the driving motor. Due to the belt force dependence on the belt stretch, the overall system can be presented as a dual mass system with flexible link. From the point of the motor shaft, the control system can be taken as a second order system with motor current as input, the motor shaft position as output, and the disturbance represented by the motor friction and the belt force reflected to the motor shaft [15]. The aim is to control the motor position without the information of the load or other disturbances. While designing the controller, the load m, the friction at the slider and at the motor bearings, and the belt force are assumed to be unknown. We assume the nominal value of the motor torque constant to be known. To control the position of the motor, the sliding manifold is chosen as $\sigma = \dot{e} + Ce$, where $e = \theta_r - \theta$ refers to the position error of the motor shaft. The control is implemented in a discrete-time form by implementing the calculation of weights as

$$w_1(k+1) = w_1(k) + \eta \frac{K_t r}{J} \left(\dot{\sigma}(k) + D\sigma(k)\right) e(k) \tag{29}$$

$$w_2(k+1) = w_2(k) + \eta \frac{K_t r}{J} \left(\dot{\sigma}(k) + D\sigma(k)\right) \dot{e}(k) \tag{30}$$

$$w_3(k+1) = w_3(k) + \eta \frac{K_t r}{J} \left(\dot{\sigma}(k) + D\sigma(k)\right). \tag{31}$$

The controller parameters are selected as $C = 10$, $D = 200$, and $\eta = 0.00001$, and the sampling rate is 0.0001 s, and the motor parameters K_t, J, and r are assumed to be constant with their nominal values.

Figs. 5–11 show the response of the system to a smooth sigmoid position reference. The selection of such a reference is dictated by the limits on the acceleration that the timing belt system can sustain and by the usual profile of the velocity curves in the point-to-point industrial positioning systems. Fig. 5 shows the position tracking curve of the motor, where the error is hardly noticeable. Fig. 6 shows the error in this tracking. As can be seen, the transient error makes a jump in the very beginning of the motion and then decreases fast during tracking. In the end, the steady-state error reaches its theoretical limit, which is set by the position measurement device. In

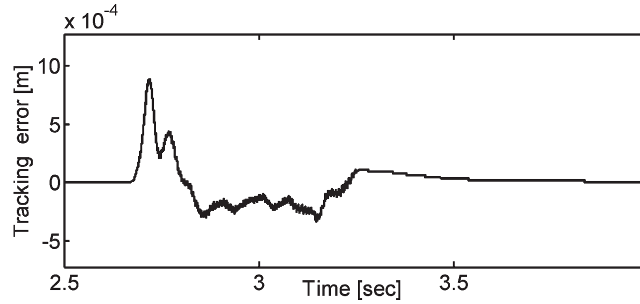


Fig. 6. Position tracking error of the motor.

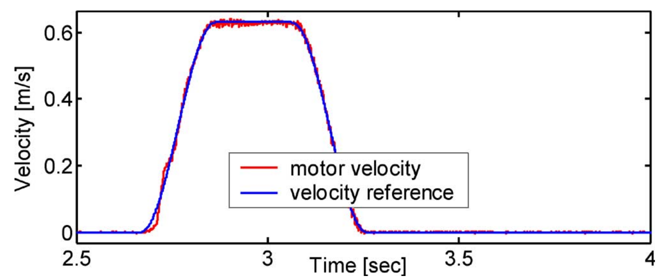


Fig. 7. Velocity tracking of the motor.

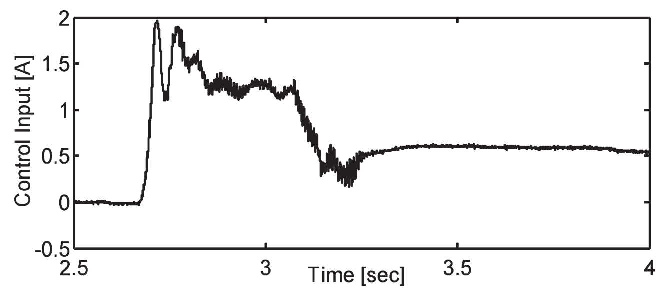


Fig. 8. Control input applied to the motor.

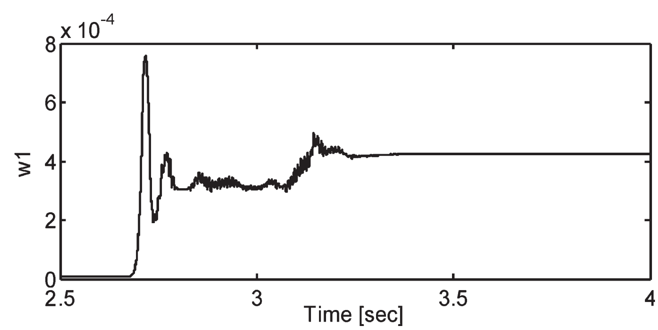


Fig. 9. Time evolution of w1.

Fig. 7, the velocity tracking of the motor is presented. This figure also shows that after a deviation from the reference, the velocity catches its reference and tracks it. The initial deviation can be explained by the weights of the network starting from zero. After they reach certain values in a short time, the system behaves as desired. In Fig. 8, the control signal produced is shown. It is seen that the control signal is sufficiently smooth.

Figs. 9–11 indicate that after having a transient period, the weights are converging to a finite value, matching with the theoretical results.

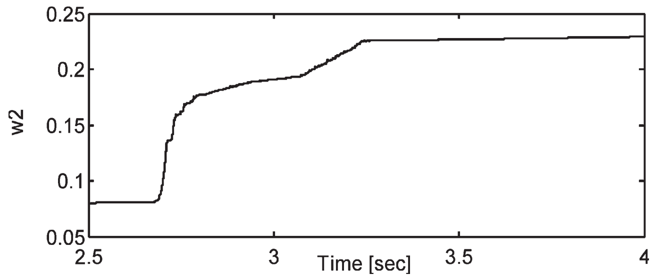


Fig. 10. Time evolution of w2.

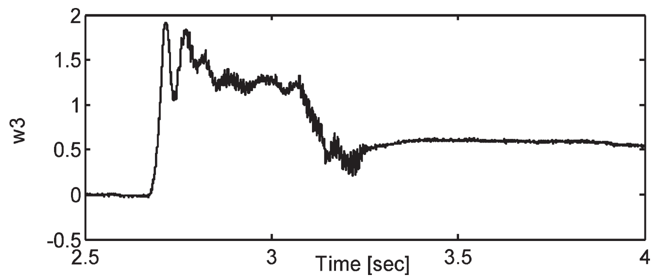


Fig. 11. Time evolution of w3.

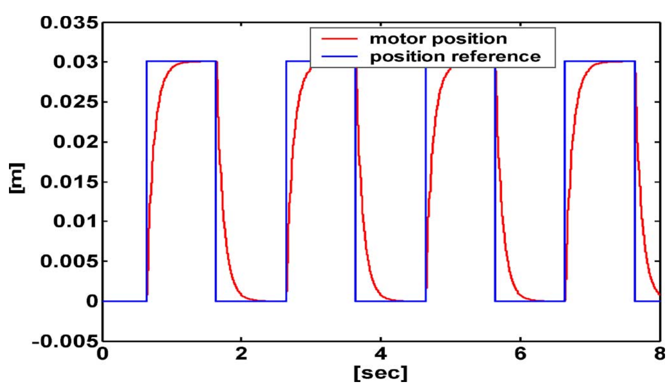


Fig. 12. Transients for step position reference.

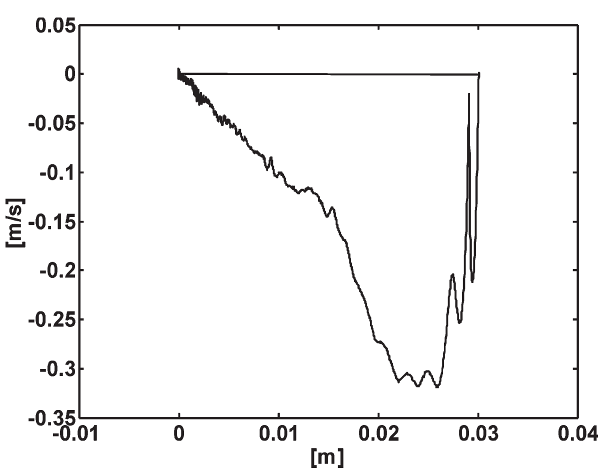


Fig. 13. Phase plane for step position reference.

In Fig. 12, the transients for a small pulse change in the motor position reference are depicted. The smooth transient without overshoot is achieved, and as shown in Fig. 13 on a phase plot diagram, the sliding-mode manifold is reached, and the sliding-mode motion is maintained in the system. This shows the capability of the proposed controller structure to cope with the nonlinear disturbance that depends on the plant state variable (the dependence of the belt force on the motor position and velocity) while taking a nominal value of the plant gain (the motor torque constant).

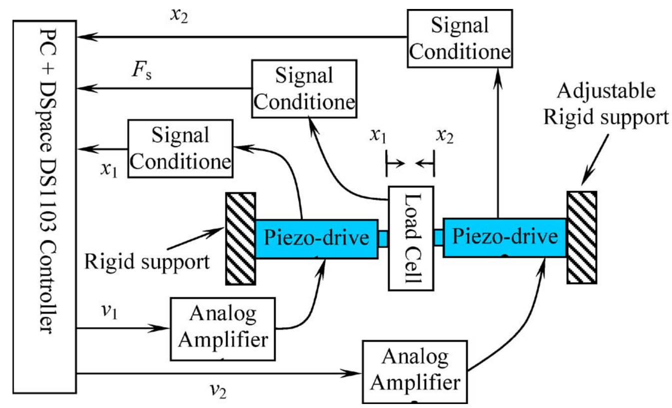


Fig. 14. Simplified experimental setup.

The presented results confirm what has been proven in the previous sections. The sliding-mode motion is achieved by a simple weight update algorithm and a very limited knowledge on the system's parameters.

B. Experiments With Piezoelectric Actuators—A MIMO System

In order to demonstrate applicability to the MIMO case, the piezoelectric transducer (PZT) dual actuator is controlled in such a way that it follows the desired trajectory while enforcing the desired grasping force. The experimental setup consists of Piezomechanik's PSt150/5/60 stack actuators ($x_{\max} = 60$ μm, $F_{\max} = 800$ N, $v_{\max} = 150$ V) connected to SVR150/3 low-voltage low-power amplifiers. The actuators have built-in strain gages for position measurement. The structure of the setup is presented in Fig. 14. In this setup, two PDs are attached to each other via a load cell that is used for force measurement. The aim is to control the position of one actuator while controlling the force that is created due to the reaction of the load cell. Force control is achieved by moving the other actuator. Thus, there are two outputs of the system: the position of one actuator and the force created in the load cell. Also, there are two inputs: the voltage input to the actuator whose position is controlled and the voltage input to the other actuator by the help of which the force is controlled. The overall system is described as two second-order systems in interaction via a load cell that is assumed without static. The conversion from input voltage to force is nonlinear, which has hysteresis characteristics [17] that result in the plant gain being a non–single-valued function of the input voltage and PZT stretch. The presence of hysteresis nonlinearity in the system in addition to the unmodeled dynamics of the load cell makes the design of the controller a challenging task. The sliding-mode manifold is selected to be the intersection of the position tracking sliding-mode function for PD-1 as $\sigma_x = \dot{e} + Ce$, where $e = x_{1r} - x_1$, and the sliding-mode function for PD-2 as $\sigma_F = F_r - F$. The weights are updated in the same way as in (28)–(30) with respective

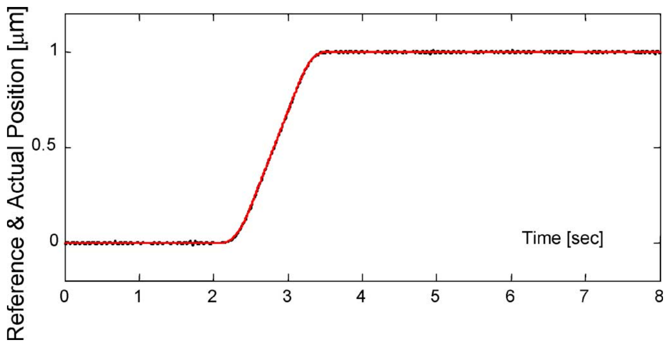


Fig. 15. Position tracking of PD-1.

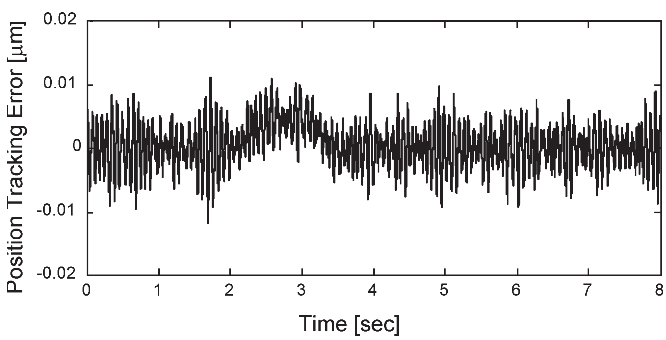


Fig. 16. Tracking error of PD-1.

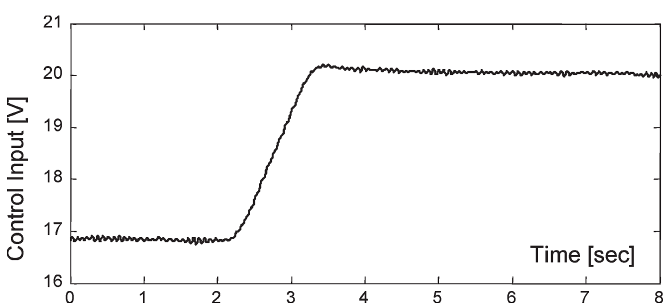


Fig. 17. Control input for PD-1.

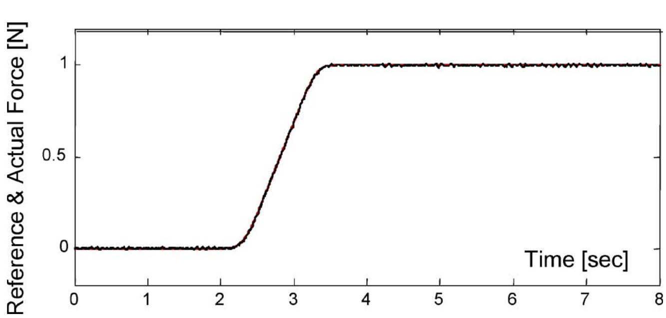


Fig. 18. Force tracking of PD-2.

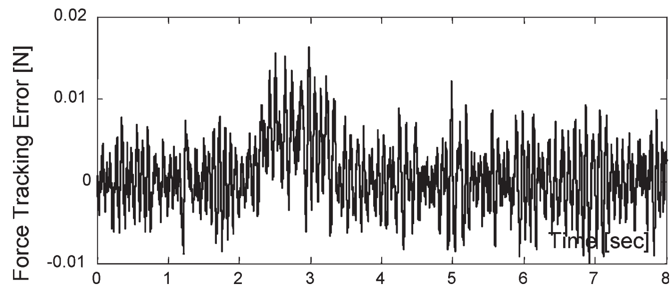


Fig. 19. Tracking error of PD-2.

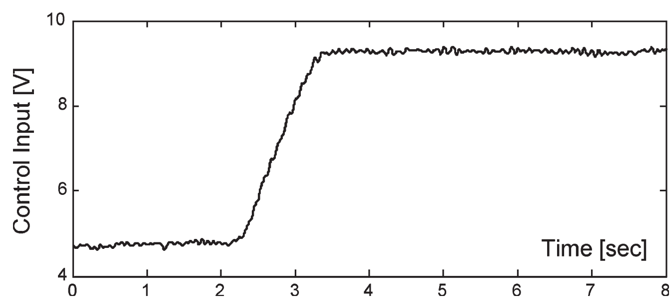


Fig. 20. Control input for PD-2.

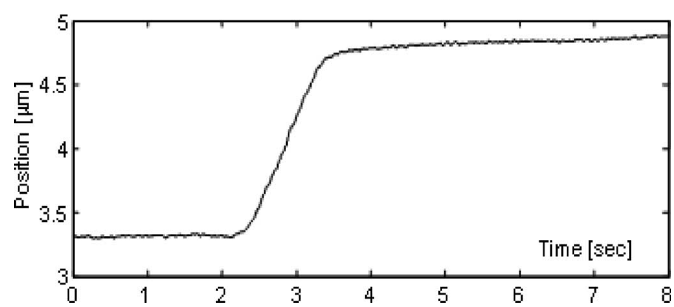


Fig. 21. Position of PD-2.

changes of the switching functions. The nonlinear gain (due to hysteresis) is assumed to have a constant value represented by the symmetry line of the hysteresis, and its variation is treated as a matched disturbance in the system to be compensated by the bias term of the controller.

Figs. 15–21 depict the response of the system for a sigmoid reference for each of the actuators. For all the experiments, the controller parameters are $D = 400$ and $\eta = 1.5$, and the sampling time is selected as 0.0001 s. As shown, the references are applied at the same time, and PD-1 is able to track a sigmoid position reference while PD-2 simultaneously moves in such a way that the force created also tracks a sigmoid reference. Fig. 20 presents the trajectory when PD-2 follows in order to maintain the sigmoid force reference while PD-1 tracks its trajectory reference. The drift visible in Fig. 20 is due to the drift of the force transducer. Also, Figs. 17 and 20 show that both control inputs are bounded and behaving well.

In Figs. 22 and 23, the behavior of the same system for sinusoidal changes in position and force is depicted. It shows the capability of the system to cope with harmonic change in both references.

VI. Conclusion

In this paper, a structure of adapting the weights of a neuro-sliding-mode controller for uncertain systems has been proposed.

Under certain conditions, the proposed controller ensures the overall stability of the closed-loop dynamics of nonlinear

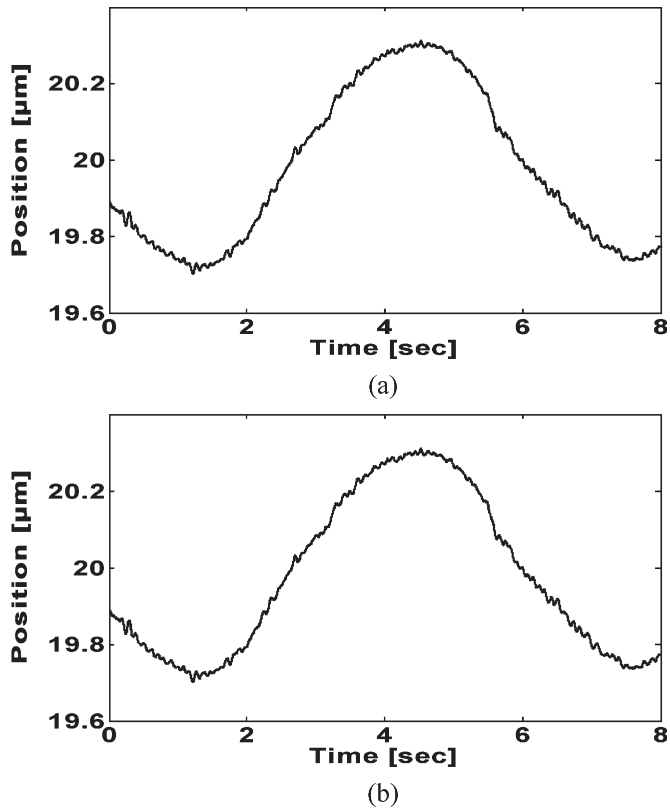


Fig. 22. (a) Position tracking of PD-1 and (b) position of PD-2 producing the force as depicted in Fig. 23.

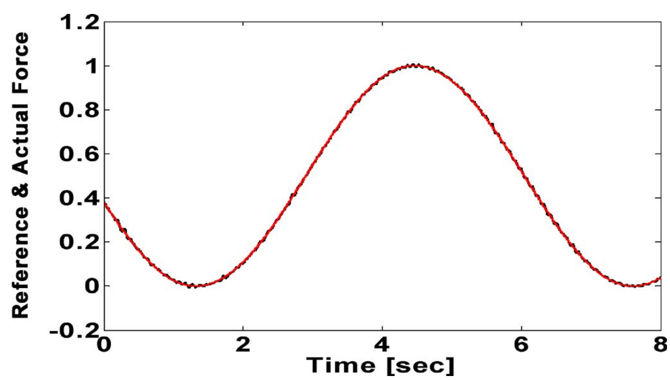


Fig. 23. Force tracking.

systems with bounded disturbances and guarantees asymptotic transient toward sliding-mode manifold, thus showing the robust properties of SMC systems without any offline training. Weight update rules are derived from the stability conditions, and for the implementation, only the sliding-mode function and the nominal value of the plant gain matrix are needed, which makes the algorithm simple enough for real applications. It is shown that under certain conditions, the rejection of disturbances that are satisfying matching condition is possible, and this is demonstrated via experimental results. The applicability of the theoretical results is demonstrated for nonlinear SISO and MIMO systems, and the results are shown to confirm predictions. The proposed structure seems promising for application in sliding-mode controller parameter adaptation.

References

[1] V. I. Utkin, "Variable structure systems with sliding modes," *IEEE Trans. Autom. Control*, vol. 22, no. 2, pp. 212–222, Apr. 1997.

[2] J. Y. Hung, W. B. Gao, and J. C. Hung, "Variable structure control: A survey," *IEEE Trans. Ind. Electron.*, vol. 40, no. 1, pp. 2–22, Feb. 1993.

[3] J. J. E. Slotine and W. Li, *Applied Nonlinear Control*. Upper Saddle River, NJ: Prentice-Hall, 1991.

[4] O. Kaynak, K. Erbatur, and M. Ertugrul, "The fusion of computationally intelligent methodologies and sliding-mode control—A survey," *IEEE Trans. Ind. Electron.*, vol. 48, no. 1, pp. 4–17, Feb. 2001.

[5] H. Morioka, K. Wada, A. Šabanovic, and K. Jezernik, "Neural network based chattering free sliding mode control," in *Proc. 34th SICE Annu. Conf.*, Hokkaido, Japan, Jul. 1995, pp. 1303–1308.

[6] K. Jezernik, M. Rodic, R. Safaric, and B. Curk, "Neural network sliding mode robot control," *Robotica*, vol. 15, no. 1, pp. 23–30, Jan. 1997.

[7] M. Rodic, K. Jezernic, A. Šabanovic, and R. Safaric, "Sliding mode based neural network learning procedure," in *Proc. 5th Int. Workshop Robot. Alpe-Adria-Danube Region*, Jun. 1996, pp. 547–552.

[8] Y. Fang, T. W. S. Chow, and X. D. Li, "Use of a recurrent neural network in discrete sliding-mode control," *Proc. Inst. Electr. Eng.—Control Theory and Appl.*, vol. 146, no. 1, pp. 84–90, Jan. 1999.

[9] M. O. Efe, O. Kaynak, and X. Yu, "Sliding mode control of a three degrees of freedom anthropoid robot by driving the controller parameters to an equivalent regime," *Trans. ASME, J. Dyn. Syst. Meas. Control*, vol. 122, no. 4, pp. 632–640, 2000.

[10] X. Yu, M. O. Efe, and O. Kaynak, "A general backpropagation algorithm for feedforward neural networks learning," *IEEE Trans. Neural Netw.*, vol. 13, no. 1, pp. 251–254, Jan. 2002.

[11] M. O. Efe, C. Unsal, O. Kaynak, and X. Yu, "Sliding mode control of uncertain systems," in *Proc. IEEE Conf. Control Appl.*, Jun. 2003, vol. 1, pp. 78–82.

[12] M. O. Efe, C. Unsal, O. Kaynak, and X. Yu, "Variable structure control of unknown systems," *Automatica*, vol. 40, no. 1, pp. 59–64, 2004.

[13] V. Utkin and S. Drakunov, "On discrete-time sliding mode control," in *Proc. IFAC Symp. NOLCOS*, Capri, Italy, 1987, pp. 484–489.

[14] A. Šabanovic, N. Šabanovic, and K. Jezernik, "Sliding modes in sampled-data systems," in *Automatika*. Zagreb, Croatia: KoREMA-IFAC, 2003, pp. 163–181.

[15] A. Hace, K. Jezernik, and A. Šabanovic, "A new robust position control algorithm for a linear belt drive," in *Proc. IEEE ICM*, Istanbul, Turkey, Jun. 2004, pp. 329–334. 108.

[16] D. Munoz and D. Sbarbaro, "An adaptive sliding-mode controller for discrete nonlinear systems," *IEEE Trans. Ind. Electron.*, vol. 47, no. 3, pp. 574–581, Jun. 2000.

[17] H. Hu and P. Y. Woo, "Fuzzy supervisory sliding-mode and neural-network control for robotic manipulators," *IEEE Trans. Ind. Electron.*, vol. 53, no. 3, pp. 929–940, Jun. 2006.

[18] F. Da and W. Song, "Fuzzy neural networks for direct adaptive control," *IEEE Trans. Ind. Electron.*, vol. 50, no. 3, pp. 507–513, Jun. 2003.

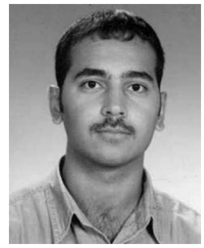
[19] F. J. Lin and P. H. Shen, "Robust fuzzy neural network sliding-mode control for two-axis motion control system," *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp. 1209–1225, Aug. 2006.

Yildiray Yildiz (S'06) received the B.S. degree in mechanical engineering from the Middle East Technical University, Ankara, Turkey, in 2002, and the M.S. degree in electrical engineering and computer science from Sabanci University, Istanbul, Turkey, in 2004. He is currently working toward the Ph.D. degree in mechanical engineering at the Massachusetts Institute of Technology, Cambridge.

He is also currently a Research Assistant at the Massachusetts Institute of Technology. His research interests include system dynamics and control, adaptive control, internal combustion engine control, and time-delay systems.

Asif Šabanovic (M'92–SM'04) received the B.S. degree in electrical engineering in 1970 and the M.S. and Dr. Eng. degrees from the University of Sarajevo, Sarajevo, Bosnia.

From 1970 to 1991, he was with the Energoinvest-Institute for Control and Computer Sciences, Sarajevo. From 1975 to 1976, he was a Visiting Researcher at the Institute of Control Science, Moscow, Russia. He was a Visiting Professor at the California Institute of Technology, Pasadena, from 1984 to 1985, a Visiting Professor at the University of Maribor, Maribor, Slovenia, in 1986, 1988, and 1989, with the Department of Electrical Engineering, University of Sarajevo, in 1991, a Hitachi Chair Professor at Keio University, Yokohama, Japan, from 1991 to 1992, a Full Professor at Yamaguchi University, Yamaguchi, Japan, from 1992 to 1993, the Head of CAD/CAM and Robotics Department at Tubitak-Marmara Research Centre, Istanbul, Turkey, from 1993 to 1995, and the Head of the Engineering Department, B.H. Engineering and Consulting, from 1995 to 1999. He is currently a Full Professor at Sabanci University, Istanbul. His fields of interest include control systems, motion control systems, robotics, mechatronics, and power electronics.



Khalid Abidi received the B.S. degree in mechanical engineering from the Middle East Technical University, Ankara, Turkey, in 2002, and the M.S. degree in electrical engineering and computer science from Sabanci University, Istanbul, Turkey, in 2004. He is currently working toward the Ph.D. degree in electrical and computer engineering at the National University of Singapore, Singapore.

He is also currently a Research Assistant at the National University of Singapore. His research interests include analysis of dynamic systems, adaptive control, iterative learning control, robust control, high-precision motion control, and microsystems.