We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

BOOK CITATION INDEX
CLARIVATE ANALYTICS
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Sliding Mode Control Using Neural Networks

Muhammad Yasser, Marina Arifin and Takashi Yahagi
Research Institute of Signal Processing
Japan

1. Introduction

Variable structure control with sliding mode, which is commonly known as sliding mode control (SMC), is a nonlinear control strategy that is well known for its robust characteristics (Utkin, 1977). The main feature of SMC is that it can switch the control law very fast to drive the system states from any initial state onto a user-specified sliding surface, and to maintain the states on the surface for all subsequent time (Utkin, 1977), (Phuah et al., 2005 a).

The conventional SMC has two disadvantages (Ertugrul & Kaynak, 2000), (Slotine & Sastry, 1983), which are the chattering phenomenon (Slotine & Sastry, 1983), (Young et al., 1999) and the difficulty in calculating the equivalent control law of SMC that requires a thorough knowledge of the parameters and dynamics of the nominal controlled plant (Ertugrul & Kaynak, 2000), (Slotine & Sastry, 1983), (Hussain & Ho, 2004).

Many methods of SMC using neural networks (NN) have been proposed (Phuah et al., 2005 a), (Ertugrul & Kaynak, 2000), (Hussain & Ho, 2004), (Phuah et al., 2005 b), (Yasser et al., 2007), (Topalov et al., 2007).

In this paper, sliding mode controls using NN are proposed to deal with the problem of eliminating the chattering effect and the difficulty in calculating the equivalent control law of SMC that requires a thorough knowledge of the parameters and dynamics of the nominal controlled plant. The first method of this method applies a method using a simplified form of the distance function proposed in (Phuah et al., 2005 a), (Phuah et al., 2005 b). Furthermore, the simplified distance function of our method uses a sliding surface in the space of the output error and its derivations, as proposed in (Yasser et al., 2006 a), (Yasser et al., 2006 c), instead of the space of the states error to construct a corrective control input. Thus, no observer is required in the proposed method. Moreover, we also propose the application of an NN to construct the equivalent control input of SMC. The weights of the NN are adjusted using a backpropagation algorithm as in (Yasser et al., 2006 b). Hence, a thorough knowledge of the parameters and dynamics of the nominal controlled plant is not required for calculating the equivalent control law. Finally, a stability analysis is carried out, and the effectiveness of this first control method is confirmed through computer simulations. This first method has been previously discussed in (Yasser et al., 2007).

The second method of this paper applies an NN to produce the gain of the corrective control of SMC. Furthermore, the output of the switching function the corrective control of SMC is applied for the learning and training of the NN. There is no equivalent control of SMC is used in this second method. As in the first method, this second method applies a method using a sliding surface in the space of the output error and its derivations, as proposed in

(Yasser et al., 2006 a), (Yasser et al., 2006 c). The weights of the NN are adjusted using a sliding mode backpropagation algorithm, that is a backpropagation algorithm using the switching function of SMC for its plant sensitivity. Thus, this second method does not use the equivalent control law of SMC, instead it uses a variable corrective control gain produced by the NN for the SMC. Hence, a thorough knowledge of the parameters and dynamics of the nominal controlled plant is not required for calculating the control law. Finally, a stability analysis is carried out, and the effectiveness of this first control method is confirmed through computer simulations.

2. Sliding mode control

In designing a standard sliding mode controller, first we are required to construct a sliding surface that represents a desired system dynamics, and then to develop a switching control law such that a sliding mode exists on every point of the sliding surface. Any states outside the surface are driven to reach the surface in a finite time.

Let us consider an SISO nonlinear plant with BIBO described as

$$\dot{\mathbf{x}}_p(t) = \mathbf{f}(\mathbf{x}_p(t)) + B_p u_p(t)$$
$$y_p(t) = C_p \mathbf{x}_p(t) + h(\mathbf{x}_p(t)) \tag{1}$$

where $\mathbf{x}_p(t)$ is an n_p th-order plant state vector, $u_p(t)$ is the control input, $y_p(t)$ is a plant output, $\mathbf{f}(\cdot)$ is a nonlinear vector function $\in R^{n_p}$, $h(\cdot)$ is a scalar nonlinear function, and B_p and C_p are matrices with appropriate dimensions. We assume that the system in (1) is controllable and observable.

The control objective is to determine a control law $u_p(t)$ such that the state vector $\mathbf{x}_p(t)$ tracks a given bounded desired state vector $\hat{\mathbf{x}}_p(t) \in R^{n_p}$. Therefore, the states error can be obtained as

$$\mathbf{e}_{x_p}(t) = \hat{\mathbf{x}}_p(t) - \mathbf{x}_p(t)$$
$$= \left[e_{x_p}(t), \dot{e}_{x_p}(t), \cdots, e_{x_p}^{(n_p-1)}(t) \right]^T. \tag{2}$$

Then the sliding surface in the space of the state error can be obtained as

$$S_{x_p}(t) = \mathbf{c}_{x_p}^T \mathbf{e}_{x_p}(t) \tag{3}$$

where $\mathbf{c}_{x_p} = \left[c_{x_p 1}, c_{x_p 2}, \cdots, c_{x_p n_p} \right]^T$ is a slope of sliding surface. Generally $\mathbf{c}_{x_p}$ is chosen to force the state error converge to zero when the state is on the sliding surface.

Meanwhile, the process of SMC can be divided into two phases: the approaching phase with $S_{x_p}(t) \neq 0$ and the sliding phase with $S_{x_p}(t) = 0$. Therefore, two types of control law: an equivalent control and a corrective control can be derived separately corresponding to those two phases.

In the sliding phase, we have $S_{x_p}(t) = 0$ and $\dot{S}_{x_p}(t) = 0$, then the equivalent control term $u_{eq}(t)$ will force the system dynamics to stay on sliding surface. The equivalent control $u_{eq}(t)$ can be obtained as

$$u_{eq}(t) = \left[\mathbf{c}_{x_p}^T B_p \right]^{-1} \left[\mathbf{c}_{x_p}^T \dot{\mathbf{x}}_p(t) - \mathbf{c}_{x_p}^T \mathbf{f}(\mathbf{x}_p(t)) \right]. \tag{4}$$

In the approaching phase, where $S(t) \neq 0$, a corrective control term $u_c(t)$ will force the state error outside the surface to reach the surface. The corrective control term $u_c(t)$ is defined as

$$u_c(t) = k_s sign\left(S_{x_p}(t)\right) \tag{5}$$

where k_s is a positive gain constant, and $sign\left(S_{x_p}(t)\right)$ is a sign function defined as

$$sign\left(S_{x_p}(t)\right) = \begin{cases} +1, \text{ if } S_{x_p}(t) > 0 \\ 0, \text{ if } S_{x_p}(t) = 0 \\ -1, \text{ if } S_{x_p}(t) < 0 \end{cases} \tag{6}$$

Then, the control law of SMC will be expressed as

$$u_p(t) = u_{eq}(t) + u_c(t). \tag{7}$$

3. Sliding mode control using Neural Networks and a simplified distance function

The first method of this method applies a method using a simplified form of the distance function proposed in (Phuah et al., 2005 a), (Phuah et al., 2005 b). An NN is applied to construct the equivalent control input of SMC. The weights of the NN are adjusted using a backpropagation algorithm as in (Yasser et al., 2006 b).

3.1 Chattering elimination using a simplified distance function

Based on the concept of point to hyperplane distance, an alternative control method to calculate the corrective control term $u_c(t)$ has been proposed in (Phuah et al., 2005 a), (Phuah et al., 2005 b) to suppress the chattering phenomenon which is caused by high frequency oscillations exhibited by the corrective control law $u_c(t)$ in (5). This method uses a distance function $h(t)$ to calculate the distance between the trajectory of the state error and the sliding surface to generate the corrective control law. The distance function $h(t)$ is defined as (Phuah et al., 2005 a), (Phuah et al., 2005)

$$h(t) = \left\| \mathbf{c}_{x_p} \right\|^{-1} S_{x_p}(t) \tag{8}$$

where $\|\cdot\|$ is the usual Euclidean norm in R^{n_p}. The corrective control law is defined as (Phuah et al., 2005 a), (Phuah et al., 2005)

$$u_c(t) = kh(t) \tag{9}$$

where k is a positive constant.
To construct the corrective control law, the distance function (8) can be simplified to minimize the calculation process, and modified by applying the sliding surface in the space

of the output error and its derivations, proposed in (Yasser et al., 2006 a), (Yasser et al., 2006 c), instead of the state error. For that, first, we consider a linear reference model to which the plant output required to follow in the form

$$\dot{\mathbf{x}}_{\mathbf{m}}(t) = A_m \mathbf{x}_m(t) + B_m u_m(t)$$
$$y_m(t) = C_m \mathbf{x}_m(t)$$

$$(10)$$

where $\mathbf{x}_m(t)$ is an n_m th-order reference model state vector, $u_m(t)$ is a reference model input, $y_m(t)$ is a reference model output, A_m, C_m are matrices with appropriate dimensions, and B_m is a scalar value. The reference model can be independent of the controlled plant, and it is permissible to assume $n_m \ll n_p$. Then, we define the output error $e_{y_p}(t)$ as

$$e_{y_p}(t) = y_m(t) - y_p(t)$$

$$(11)$$

Thus, the simplified distance function $h_{sim}(t)$ can be described as

$$h_{sim}(t) = k_{sim} S_{y_p}(t)$$

$$(12)$$

where k_{sim} is a positive constant, and $S_{y_p}(t)$ is a sliding surface in the space of the output error and its derivations described as (Yasser et al., 2006 a), (Yasser et al., 2006 c)

$$S_{y_p}(t) = \mathbf{c}_{y_p}^T \mathbf{e}_{y_p}(t)$$
$$= \left[c_{y_p 1}, c_{y_p 2}, \cdots, c_{y_p n_s} \right] \left[e_{y_p}(t), \dot{e}_{y_p}(t), \cdots, e_{y_p}^{(n_s-1)}(t) \right]^T$$

$$(13)$$

where $n_s > 2$.

Then, by replacing $h(t)$ in (9) with $h_{sim}(t)$ from (12), a new corrective control law can be defined as

$$u_c(t) = k h_{sim}(t) = k_{y_p} S_{y_p}(t)$$

$$(14)$$

where $k_{y_p} = k \cdot k_{sim}$ is a positive constant.

3.2 Neural Networks for equivalent control

To avoid the requirement of the thorough knowledge of the parameters and dynamics of the nominal plant (1), we use a feedforward NN, which consists of an input layer, a hidden layer, and an output layer as in (Yasser et al., 2006 b), to construct the equivalent control input $u_{eq}(t)$ of the SMC in (7). The equivalent control input $u_{eq}(t)$ is described as

$$u_{eq}(t) = \alpha u_{NN}(t)$$
$$= \alpha f_{ZOH}\left(u_{NN}(k) \right)$$

$$(15)$$

where α is a positive constant, $u_{NN}(t)$ is a continuous-time output of the NN, $u_{NN}(k)$ is a discrete-time output of the NN, and $f_{ZOH}(\cdot)$ is a zero-order hold function.

As in (Yasser et al., 2006 b), we implement a sampler in front of the NN with an appropriate sampling period to obtain the discrete-time input of the NN, and a zero-order hold is

implemented to transform the discrete-time output $u_{NN}(k)$ of the NN back to the continuous-time output $u_{NN}(t)$ of the NN.

The input $i(k)$ of the NN is given as

$$i(k) = \left[e_{y_p}(k-1), \cdots, e_{y_p}(k-n) \right] \qquad (16)$$

where $e_{y_p}(k)$ is the discrete-time form of $e_{y_p}(t)$ in (11). And the dynamics of the NN are given as (Yasser et al., 2006 b)

$$h_q(k) = \sum_i i_i(k) m_{iq}(k) \qquad (17)$$

$$\begin{aligned} u_{NN}(k) &= o(k) \\ &= \sum_i S_1(h_q(k)) m_{qj}(k) \end{aligned} \qquad (18)$$

where $i_i(k)$ is the input to the i-th neuron in the input layer ($i = 1, \cdots, n_i$), $h_q(k)$ is the input to the q-th neuron in the hidden layer ($q = 1, \cdots, n_q$), $o(k)$ is the input to the single neuron in the output layer, n_i and n_q are the number of neurons in the input layer and the hidden layer, respectively, $m_{iq}(k)$ are the weights between the input layer and the hidden layer, $m_{qj}(k)$ are the weights between the hidden layer and the output layer, and $S_1(\cdot)$ is a sigmoid function. The sigmoid function is chosen as

$$S_1(X) = \frac{2}{1 + \exp(-\mu X)} - 1 \qquad (19)$$

where $\mu > 0$.

The objective of the NN training is to minimize the error function $E_{y_p}(k)$ described as

$$E_{y_p}(k) = \frac{1}{2} e_{y_p}^2(k) = \frac{1}{2} \sum_j^{n_i} \left[y_m(k) - y_p(k) \right] \qquad (20)$$

where $e_{y_p}(k)$ is the discrete-time form of $e_{y_p}(t)$ in (11). The NN training is done by adapting $m_{iq}(k)$ and $m_{qj}(k)$ using the method in (Yasser et al., 2006 b) as follows

$$\begin{aligned} \Delta m_{qj}(k) &= -c \cdot \frac{\partial E(k)}{\partial m_{qj}(k)} \\ &= c \cdot \left[y_m(k) - y_p(k) \right] \cdot J_{plant} \cdot S_1(h_q(k)) \end{aligned} \qquad (21)$$

$$\begin{aligned} \Delta m_{iq}(k) &= -c \cdot \frac{\partial E(k)}{\partial m_{iq}(k)} \\ &= c \cdot \left[y_m(k) - y_p(k) \right] \cdot J_{plant} \cdot m_{qj}(k) \cdot \frac{\mu}{2}(1 - S_1^2(X)) \cdot i_i(k) \end{aligned} \qquad (22)$$

where c is a learning parameter, and J_{plant} represents the plant Jacobian estimated using

$$J_{plant} = sign\left(\frac{\partial y_p(k)}{\partial u_{NN}(k)}\right) \tag{23}$$

as in (Yasser et al., 2006 b).

3.3 Stability

For the stability analysis of our method, we start by defining its Lyapunov function and its derivation as follows

$$V_{SMCNN}(t) = V_{NN}(t) + V_{SMC}(t)$$
$$\dot{V}_{SMCNN}(t) = \dot{V}_{NN}(t) + \dot{V}_{SMC}(t) \tag{24}$$

where $V_{NN}(t)$ is the Lyapunov function of the NN of our method, and $V_{SMC}(t)$ is the Lyapunov function of SMC of our method.

For $\dot{V}_{NN}(t)$, we assume that it can be approximated as

$$\dot{V}_{NN}(t) \cong \frac{\Delta V_{NN}(k)}{\Delta T} \tag{25}$$

where $\Delta V_{NN}(k)$ is the derivation of a discrete-time Lyapunov function, and ΔT is a sampling time. According to (Yasser et al., 2006 b), $\Delta V_{NN}(k)$ can be guaranteed to be negative definite if the learning parameter c satisfies the following conditions

$$0 < c < \frac{2}{n_q} \tag{26}$$

for the weights between the hidden layer and the output layer, $m_{qj}(k)$, and

$$0 < c < \frac{2}{n_q}\Big[\max_k\big\|m_{qj}(k)\big\| \cdot \max_k\big\|i_i(k)\big\|\Big]^{-2} \tag{27}$$

for the weights between the input layer and the hidden layer, $m_{iq}(k)$. Furthermore, if the conditions in (26) and (27) are satisfied, the negativity of $\dot{V}_{NN}(t)$ can also be increased by reducing ΔT in (25).

For $V_{SMC}(t)$, it is defined as

$$V_{SMC}(t) = \frac{S_{y_p}^2(t)}{2}$$
$$\dot{V}_{SMC}(t) = S_{y_p}(t)\dot{S}_{y_p}(t). \tag{28}$$

Then we the following assumption.

Assumption 1: The sliding surface in (13) can approximate the sliding surface in (3) (Yasser et al., 2006 c)

$$S_{y_p}(t) \cong S_{x_p}(t). \tag{29}$$

$\dot{V}_{SMC}(t)$ in (28) can be assured to be negative definite if

$$\dot{S}_{y_p}(t) \cong \dot{S}_{x_p}(t)$$
$$= -k'_{y_p} S_{y_p}(t) \cdot$$
(30)

where k'_{y_p} is a positive constant. Following the stability analysis method in (Phuah et al., 2005 a), we apply $(1)-(3)$, (7), (14), (15), (29) and (30) to (28) and assume that (15) can approximate (4). Thus, $\dot{V}_{SMC}(t)$ can be described as

$$\dot{V}_{SMC}(t) = S_{y_p}(t)\dot{S}_{x_p}(t)$$
$$= S_{y_p}(t)\mathbf{c}_{x_p}^T \dot{\mathbf{e}}_{x_p}(t)$$
$$= S_{y_p}(t)\mathbf{c}_{x_p}^T \left[\dot{\hat{\mathbf{x}}}_p(t) - \dot{\mathbf{x}}_p(t) \right]$$
$$= S_{y_p}(t)\left[\mathbf{c}_{x_p}^T \dot{\hat{\mathbf{x}}}_p(t) - \mathbf{c}_{x_p}^T \mathbf{f}(\mathbf{x}_p(t)) - \mathbf{c}_{x_p}^T B_p u_p(t) \right]$$
(31)
$$= S_{y_p}(t)\left[\mathbf{c}_{x_p}^T \dot{\hat{\mathbf{x}}}_p(t) - \mathbf{c}_{x_p}^T \mathbf{f}(\mathbf{x}_p(t)) - \mathbf{c}_{x_p}^T B_p \left[u_{eq}(t) + u_c(t) \right] \right]$$
$$= S_{y_p}(t)\left[-\bar{k}_{y_p} S_{y_p}(t) \right]$$
$$= -\bar{k}_{y_p} S_{y_p}^2(t)$$

which is negative definite, where $\bar{k}_{y_p} = \mathbf{c}_{x_p}^T B_p k_{y_p}$. The reaching condition (Phuah et al., 2005 a) can be achieved if

$$-\bar{k}_{y_p} S_{y_p}(t) \leq \eta sign(S_{y_p}(t)) .$$
(32)

where η is a small positive constant.

3.4 Simulation
Let us consider an SISO nonlinear plant described by (Yasser et al., 2006 b)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ 2x_1 \sin(x_1) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_p$$
$$y_p = x_1 + \sin(x_1)$$
(33)

and the parameters $\mathbf{c}_{y_p} = \begin{bmatrix} 12 & 1 \end{bmatrix}^T$ in (13), $k_{y_p} = 20$ in (14), $\alpha = 0.1$ in (15), $n_i = 1$ in(17), $n_q = 5$ in (18), $\mu = 2$ in (19), $c = 0.001$ in (21) and (22), $J_{plant} = +1$ in (21), and $\Delta T = 0.01$ in (25) are all fixed. The switching speed for the corrective control of SMC is set to 0.02 seconds. We assume a first-order reference model in (10) with parameters $A_m = -10$, $B_m = -10$, and $C_m = 1$.

Fig. 1 and Fig. 2 show the outputs of the reference model $y_m(t)$ and the plant output $y_p(t)$ using the conventional method of SMC with an NN and a sign function. These figures show that the plant output $y_p(t)$ can follow the output of the reference model $y_m(t)$ closely but not smoothly, as chattering occurs as seen in Fig. 2.

Fig. 3 and Fig. 4 show the outputs of the reference model $y_m(t)$ and the plant output $y_p(t)$ using our proposed method. It can be seen that the plant output $y_p(t)$ can follow the output

of the reference model $y_m(t)$ closely and smoothly, as chattering has been eliminated as seen in Fig. 4.
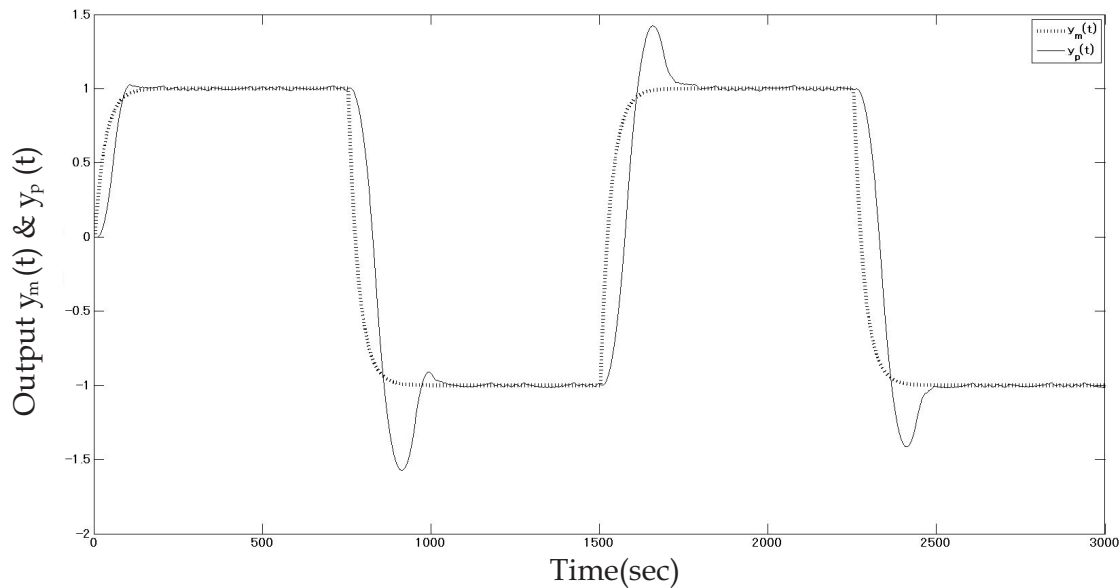

Fig. 1. $y_m(t)$ and $y_p(t)$ using SMC with NN and a sign function
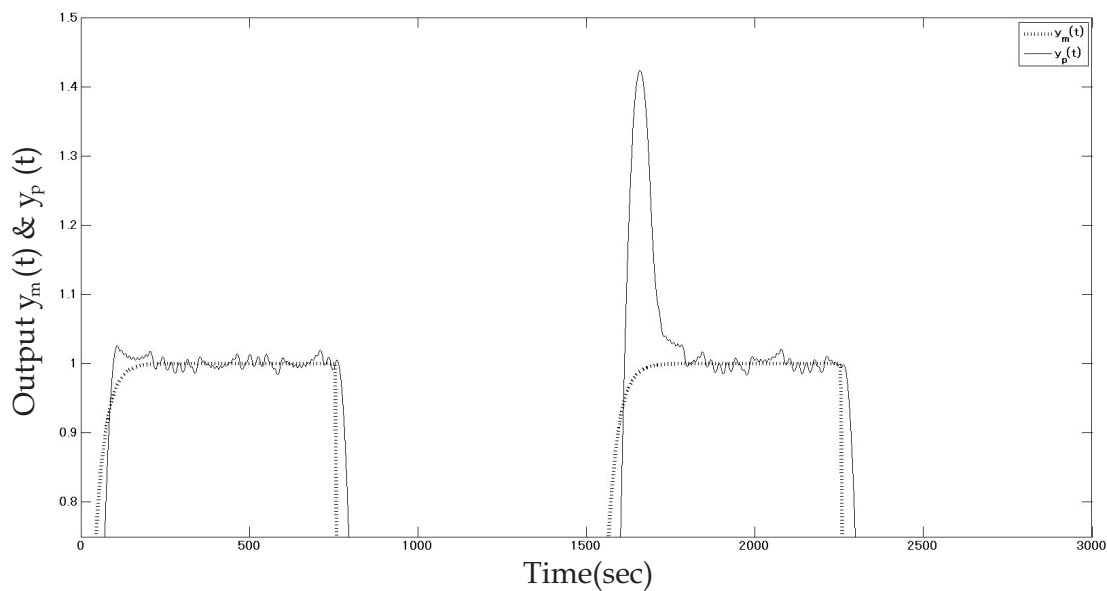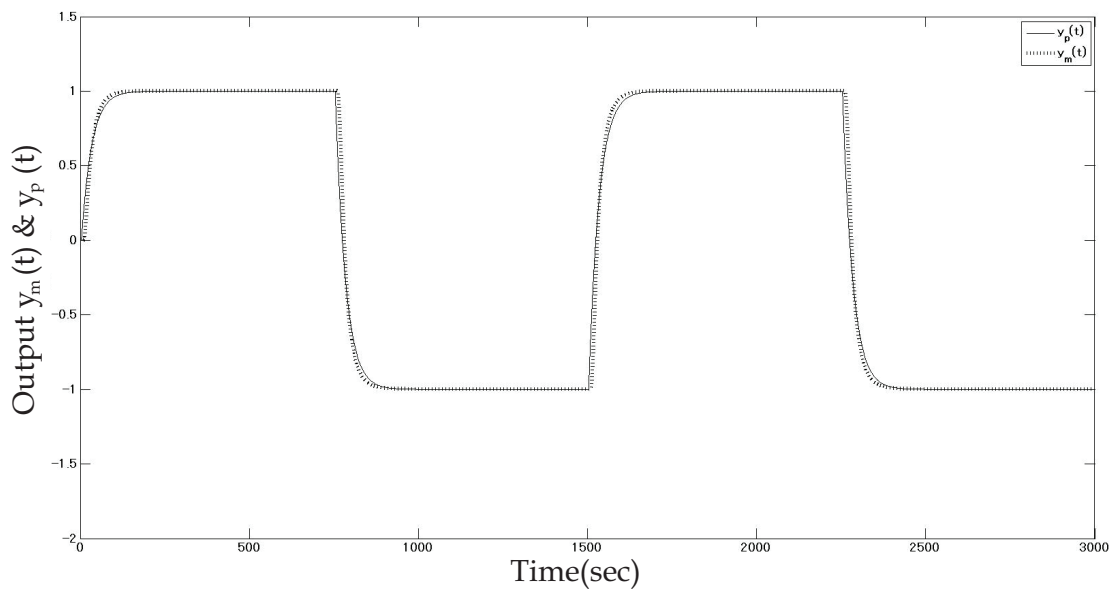

Fig. 2. Magnified upper parts of the curves in Fig. 1

4. Sliding mode control with a variable corrective control gain using Neural Networks

The method in this subsection applies an NN to produce the gain of the corrective control of SMC. Furthermore, the output of the switching function the corrective control of SMC is applied for the learning and training of the NN. There is no equivalent control of SMC is used in this second method.

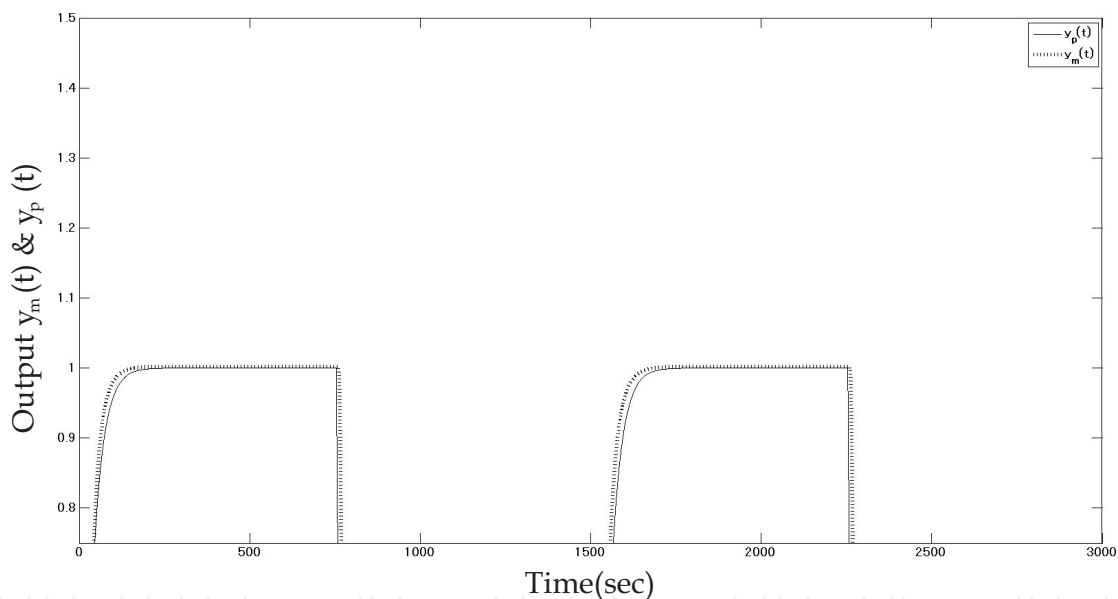Fig. 3. $y_m(t)$ and $y_p(t)$ using SMC with NN and the simplified distance function

Fig. 4. Magnified upper parts of the curves in Fig. 3

4.1 A variable corrective control gain using Neural Networks for chattering elimination

Using NN to produce a variable gain for a corrective control gain of SMC, instead of using a fixed gain in the conventional SMC, can eliminate the chattering. The switching function of the corrective control is used in the sliding mode backpropagation algorithm to adjust the weight of the NN. This method of SMC does not use any equivalent control of (7) in its control law. For the SISO nonlinear plant with BIBO described in (1), the control input of SMC with a variable corrective control gain using NN is given as

$$u_p(t) = u_{cV}(t) \tag{34}$$

$$u_{cV}(t) = k_{y_p V}(t) sign\left(S_{y_p}(t)\right) \tag{35}$$

where $u_{cV}(t)$ is the corrective control with variable gain using NN, and $k_{y_p V}(t)$ is the variable gain produced by NN described as

$$\begin{aligned} k_V(t) &= \alpha |u_{NNV}(t)| \\ &= \alpha f_{ZOH}(u_{NNV}(k)) \end{aligned} \tag{36}$$

where α is a positive constant, $u_{NNV}(t)$ is a continuous-time output of the NN, $u_{NNV}(k)$ is a discrete-time output of the NN, $|\cdot|$ is an absolute function, and $f_{ZOH}(\cdot)$ is a zero-order hold function.

As in subsection 3.2, we implement a sampler in front of the NN with an appropriate sampling period to obtain the discrete-time input of the NN, and a zero-order hold is implemented to transform the discrete-time output $u_{NNV}(k)$ of the NN back to the continuous-time output $u_{NNV}(t)$ of the NN.

The input $i(k)$ of the NN is given as in (16), and the dynamics of the NN are given as

$$h_{Vq}(k) = \sum_i i_i(k) m_{Viq}(k) \tag{37}$$

$$\begin{aligned} u_{NNV}(k) &= o_V(k) \\ &= \sum_i S_1(h_{Vq}(k)) m_{Vqj}(k) \end{aligned} \tag{38}$$

where $i_i(k)$ is the input to the i-th neuron in the input layer ($i = 1, \cdots, n_{Vi}$), $h_{Vq}(k)$ is the input to the q-th neuron in the hidden layer ($q = 1, \cdots, n_{Vq}$), $o_V(k)$ is the input to the single neuron in the output layer, n_{Vi} and n_{Vq} are the number of neurons in the input layer and the hidden layer, respectively, $m_{Viq}(k)$ are the weights between the input layer and the hidden layer, $m_{Vqj}(k)$ are the weights between the hidden layer and the output layer, and $S_1(\cdot)$ is a sigmoid function. The sigmoid function is chosen as in (19).

4.2 Sliding mode backpropagation for Neural Networks training

In the sliding mode backpropagation, the objective of the NN training is to minimize the error function $E_{y_p}(k)$ described in (20). The NN training is done by adapting $m_{Viq}(k)$ and $m_{Vqj}(k)$ as follows

$$\begin{aligned} \Delta m_{Vqj}(k) &= -c \cdot \frac{\partial E(k)}{\partial m_{Vqj}(k)} \\ &= c \cdot \left[y_m(k) - y_p(k) \right] \cdot J_{V plant} \cdot S_1(h_{Vq}(k)) \end{aligned} \tag{39}$$

$$\begin{aligned} \Delta m_{Viq}(k) &= -c \cdot \frac{\partial E(k)}{\partial m_{Viq}(k)} \\ &= c \cdot \left[y_m(k) - y_p(k) \right] \cdot J_{V plant} \cdot m_{Vqj}(k) \cdot \frac{\mu}{2}(1 - S_1^2(X)) \cdot i_i(k) \end{aligned} \tag{40}$$

where c is the learning parameter, and J_{Vplant} is described as

$$J_{Vplant} = sign\left(\frac{\partial y_p(k)}{\partial u_{NNV}(k)}\right) \cdot sign\left(S_{y_p}(k)\right)$$
$$= \overline{J}_{plant} \cdot sign\left(S_{y_p}(k)\right) \tag{41}$$

where $S_{y_p}(k)$ is the time-sampled form of $S_{y_p}(t)$ in (13).

4.3 Stability

For the stability analysis of our method, we start by defining its Lyapunov function and its derivation as follows

$$V_{SMCNN_V}(t) = V_{NN_V}(t) + V_{SMC_V}(t)$$
$$\dot{V}_{SMCNN_V}(t) = \dot{V}_{NN_V}(t) + \dot{V}_{SMC_V}(t) \tag{42}$$

where $V_{NNV}(t)$ is the Lyapunov function of the NN of our method, and $V_{SMCV}(t)$ is the Lyapunov function of SMC of our method.

For $\dot{V}_{NN_V}(t)$, we assume that it can be approximated as

$$\dot{V}_{NN_V}(t) \cong \frac{\Delta V_{NNV}(k)}{\Delta T} \tag{43}$$

where $\Delta V_{NNV}(k)$ is the derivation of a discrete-time Lyapunov function, and ΔT is a sampling time. According to (Yasser et al., 2006 b), $\Delta V_{NNV}(k)$ can be guaranteed to be negative definite if the learning parameter c satisfies the following conditions

$$0 < c < \frac{2}{n_{Vq}} \tag{44}$$

for the weights between the hidden layer and the output layer, $m_{Vqj}(k)$, and

$$0 < c < \frac{2}{n_{Vq}}\left[\max_k \|m_{Vqj}(k)\| \cdot \max_k \|i_i(k)\|\right]^{-2} \tag{45}$$

for the weights between the input layer and the hidden layer, $m_{Viq}(k)$. Furthermore, if the conditions in (44) and (45) are satisfied, the negativity of $\dot{V}_{VNN}(t)$ can also be increased by reducing ΔT in (43).

For $V_{SMCV}(t)$, it is defined as

$$V_{SMC_V}(t) = \frac{S_{y_p}^2(t)}{2}$$
$$\dot{V}_{SMC_V}(t) = S_{y_p}(t)\dot{S}_{y_p}(t). \tag{46}$$

Then we again use assumption 1. Thus, $\dot{V}_{SMC_V}(t)$ in (46) can be assured to be negative definite if

$$\dot{S}_{y_p}(t) \cong \dot{S}_{x_p}(t)$$
$$= -k'_{y_{pV}} S_{y_p}(t) \quad . \tag{47}$$

where $k'_{y_{pV}}$ is a positive constant. Based on the stability analysis method in subsection 3.3, we apply $(1) - (3)$, (34), (35), (29) and (30) to (28). Thus, $\dot{V}_{SMC}(t)$ can be described as

$$
\begin{aligned}
\dot{V}_{SMC_V}(t) &= S_{y_p}(t)\dot{S}_{x_p}(t) \\
&= S_{y_p}(t)\mathbf{c}_{x_p}^T \dot{\mathbf{e}}_{x_p}(t) \\
&= S_{y_p}(t)\mathbf{c}_{x_p}^T \left[\dot{\hat{\mathbf{x}}}_p(t) - \dot{\mathbf{x}}_p(t) \right] \\
&= S_{y_p}(t)\left[\mathbf{c}_{x_p}^T \dot{\hat{\mathbf{x}}}_p(t) - \mathbf{c}_{x_p}^T \mathbf{f}(\mathbf{x}_p(t)) - \mathbf{c}_{x_p}^T B_p u_p(t) \right] \\
&= S_{y_p}(t)\left[\mathbf{c}_{x_p}^T \dot{\hat{\mathbf{x}}}_p(t) - \mathbf{c}_{x_p}^T \mathbf{f}(\mathbf{x}_p(t)) - \mathbf{c}_{x_p}^T B_p \left[k_{y_{pV}}(t) sign\left(S_{y_p}(t) \right) \right] \right] \\
&= S_{y_p}(t)\left[\mathbf{c}_{x_p}^T \dot{\hat{\mathbf{x}}}_p(t) - \mathbf{c}_{x_p}^T \mathbf{f}(\mathbf{x}_p(t)) \right] - \overline{k}_{y_{pV}}(t) S_{y_p}(t) sign\left(S_{y_p}(t) \right)
\end{aligned} \quad . \tag{48}
$$

where $\overline{k}_{y_{pV}}(t) = \mathbf{c}_{x_p}^T B_p k_{y_p V}(t)$. $\dot{V}_{SMC}(t)$ in (48) is negative definite if $k_{y_p V}(t)$ produced by the NN is large enough. The reaching condition (Phuah et al., 2005 a) can be achieved if

$$S_{y_p}(t)\left[\mathbf{c}_{x_p}^T \dot{\hat{\mathbf{x}}}_p(t) - \mathbf{c}_{x_p}^T \mathbf{f}(\mathbf{x}_p(t)) \right] - \overline{k}_{y_{pV}}(t) S_{y_p}(t) sign\left(S_{y_p}(t) \right) \le \eta sign(S_{y_p}(t)) . \tag{49}$$

where η is a small positive constant.

4.4 Simulation

Let us consider an SISO nonlinear plant described in (33) and the parameters $\mathbf{c}_{y_p} = \begin{bmatrix} 9 & 1 \end{bmatrix}^T$ in (13), $\alpha = 1$ in (36), $n_{Vi} = 2$ in (37), $n_q = 5$ in (38), $\mu = 2$ in (19) and (40), $c = 0.01$ in (39) and (40), $\overline{J}_{plant} = +1$ in (41), and $\Delta T = 0.01$ in (43) are all fixed. The switching speed for the corrective control of SMC is set to 0.02 seconds. We assume a first-order reference model in (10) with parameters $A_m = -10$, $B_m = -10$, and $C_m = 1$.

Fig. 5 and Fig. 6 show the outputs of the reference model $y_m(t)$ and the plant output $y_p(t)$ using our proposed method. It can be seen that the plant output $y_p(t)$ can follow the output of the reference model $y_m(t)$ closely and smoothly, as chattering has been eliminated as seen in Fig. 6.

5. Conclusion

In this chapter, we proposed two new SMC strategies using NN for SISO nonlinear systems with BIBO has been proposed to deal with the problem of eliminating the chattering effect.

In the first method, to eliminate the chattering effect, it applied a method using a simplified distance function. Furthermore, we also proposed the application of an NN using the backpropagation algorithm to construct the equivalent control input of SMC.

The second method of this paper applied an NN to produce the gain of the corrective control of SMC. Furthermore, the output of the switching function the corrective control of

SMC was applied for the learning and training of the NN. There was no equivalent control of SMC used in this second method. The weights of the NN were adjusted using a sliding mode backpropagation algorithm, that was a backpropagation algorithm using the switching function of SMC for its plant sensitivity. Thus, this second method did not use the equivalent control law of SMC, instead it used a variable corrective control gain produced by the NN for the SMC.

Brief stability analysis was carried out for the two methods, and the effectiveness of our control methods was confirmed through computer simulations.
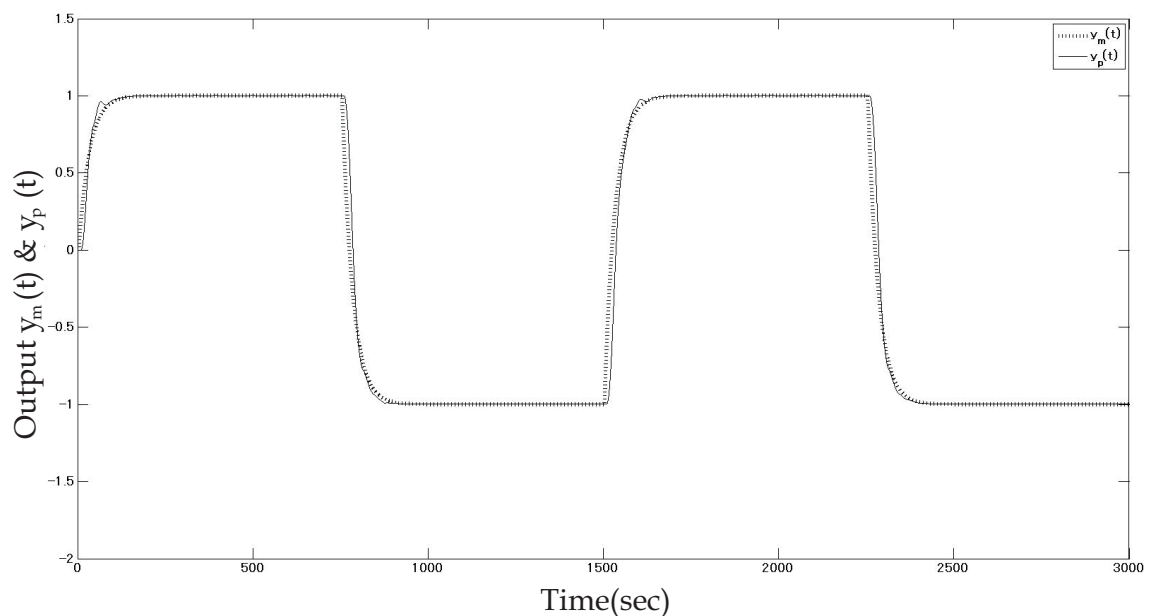
Fig. 5. $y_m(t)$ and $y_p(t)$ using SMC with a variable corrective gain using NN
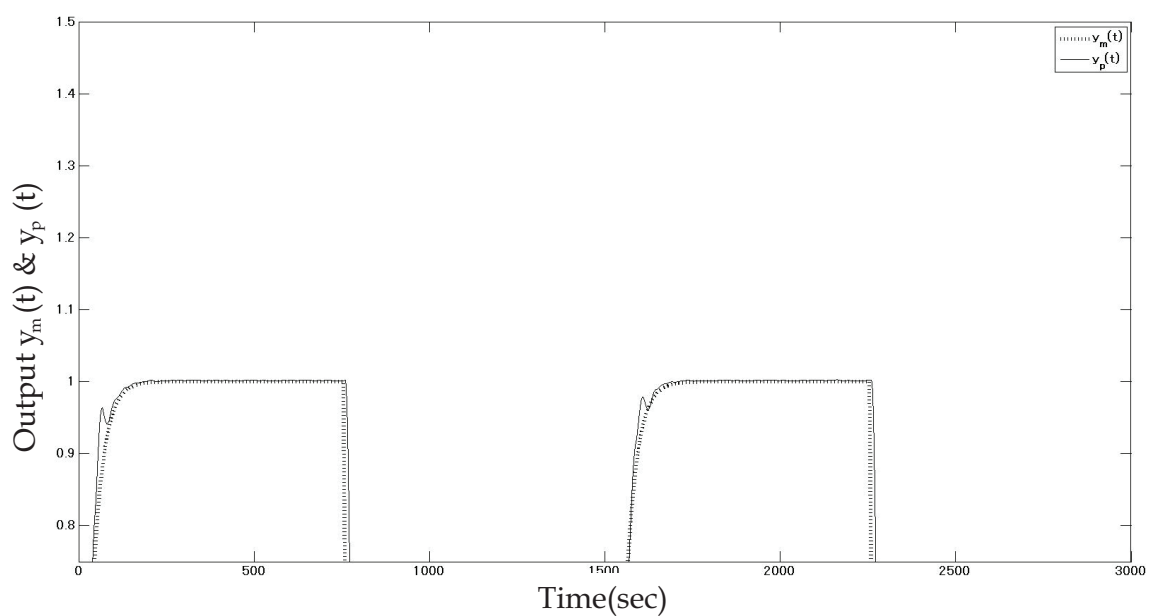
Fig. 6. Magnified upper parts of the curves in Fig. 5

6. References

Ertugrul, M. & Kaynak, O. (2000). Neuro-sliding mode control of robotic manipulators. Mechatronics, Vol. 10, page numbers 239–263, ISSN: 0957-4158

Hussain, M.A. & Ho, P.Y. (2004). Adaptive sliding mode control with neural network based hybrid models. Journal of Process Control, Vol. 14, page numbers 157−176, ISSN: 0959-1524

Phuah, J.; Lu, J. & Yahagi, T. (2005) (a). Chattering free sliding mode control in magnetic levitation system. IEEJ Transactions on Electronics, Information, and Systems, Vol. 125, No. 4, page numbers 600−606, ISSN: 0385-4221

Phuah, J.; Lu, J.; Yasser, M.; & Yahagi, T. (2005) (b). Neuro-sliding mode control for magnetic levitation systems, Proceedings of the 2005 IEEE International Symposium on Circuits and Systems (ISCAS 2005), page numbers 5130−5133, ISBN: 0-7803-8835-6, Kobe, Japan, May 2005, IEEE, USA

Slotine, J.E. & Sastry, S. S. (1983). Tracking control of nonlinear systems using sliding surface with application to robotic manipulators. International Journal of Control, Vol. 38, page numbers 465−492, ISSN: 1366-5820

Topalov, A.V.; Cascella, G.L.; Giordano, V.; Cupertion, F. & Kaynak, O. (2007). Sliding mode neuro-adaptive control of electric drives. IEEE Transactions on Industrial Electronnics, Vol. 54, page numbers 671−679, ISSN: 0278-0046

Utkin, V.I. (1977). Variable structure systems with sliding mode. IEEE Transactions on Automatic Control, Vol. 22, page numbers 212−222 , ISSN: 00189286

Yasser, M.; Trisanto, A.; Lu, J. & Yahagi T.(2006) (a). Adaptive sliding mode control using simple adaptive control for SISO nonlinear systems, Proceedings of the 2006 IEEE International Symposium on Circuits and Systems (ISCAS 2006), page numbers 2153−2156, ISBN: 0-7803-9390-2, Island of Kos, Greece, May 2006, IEEE, USA

Yasser, M.; Trisanto, A.; Haggag, A.; Lu, J.; Sekiya, H. & Yahagi, T. (2006) (c). An adaptive sliding mode control for a class of SISO nonlinear systems with bounded-input bounded-output and bounded nonlinearity, Proceedings of the 45th IEEE Conference on Decision and Control (45th CDC), page numbers 1599–1604, ISBN: 1-4244-0171-2, San Diego, USA, December 2006, IEEE, USA

Yasser, M., Trisanto, A., Lu, J. & Yahagi, T. (2006) (b). A method of simple adaptive control for nonlinear systems using neural networks. IEICE Transactions on Fundamentals, Vol. E89-A, No. 7, page numbers 2009−2018, ISSN: 1745-1337

Yasser, M., Trisanto, A., Haggag, A., Yahagi, T., Sekiya, H. & Lu, J. (2007). Sliding mode control using neural networks for SISO nonlinear systems, Proceedings of The SICE Annual Conference 2007, International Conference on Instrumentation, Control and Information Technology (SICE2007), page numbers 980–984, ISBN: 978-4-907764-27-2, Takamatsu, Japan, September 2007, SICE Japan

Young, K.D.; Utkin, V.I. & Ozguner, U. (1999). A control engineer's guide to sliding mode control. IEEE Transactions On Control System Technology, Vol. 7, No. 3, ISSN: 1063-6536

Sliding Mode Control

Edited by Prof. Andrzej Bartoszewicz

The main objective of this monograph is to present a broad range of well worked out, recent application studies as well as theoretical contributions in the field of sliding mode control system analysis and design. The contributions presented here include new theoretical developments as well as successful applications of variable structure controllers primarily in the field of power electronics, electric drives and motion steering systems. They enrich the current state of the art, and motivate and encourage new ideas and solutions in the sliding mode control area.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Muhammad Yasser, Marina Arifin and Takashi Yahagi (2011). Sliding Mode Control Using Neural Networks, Sliding Mode Control, Prof. Andrzej Bartoszewicz (Ed.), ISBN: 978-953-307-162-6, InTech, Available from: http://www.intechopen.com/books/sliding-mode-control/sliding-mode-control-using-neural-networks

INTECH
open science | open minds