Implementation of an Adaptive Neural Terminal Sliding Mode for Tracking Control of Magnetic Levitation Systems

THANH NGUYEN TRUONG, ANH TUAN VO, AND HEE-JUN KANG

School of Electrical Engineering, University of Ulsan, Ulsan 44610, South Korea

Corresponding author: Hee-Jun Kang (hj.kang@ulsan.ac.kr)

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant NRF-2019R1D1A3A03103528.

Abstract

In this article, an adaptive neural terminal sliding mode is implemented for tracking control of magnetic levitation systems with the presence of dynamical uncertainty and exterior perturbation. By proposing a novel fast terminal sliding manifold function with the dynamic coefficients, the system state variables quickly converge the equilibrium point on the manifold function. Besides, an adaptive, robust reaching control law combined with radial basis function neural network compensator drives the system fast approaching the sliding manifold function regardless of whether the initial value is near or far from the sliding manifold and reduces the chattering of the conventional terminal sliding mode control. With a design approach based on the combination of the proposed sliding manifold and the combined control law, the implemented control method provides a control performance with significant improvement in the terms of chattering reduction, high tracking accuracy, fast convergence along with simple design for real applications. The experimental work is implemented for a real magnetic levitation system to demonstrate the superior efficiency of the proposed terminal sliding mode control. The stable evidence of the proposed method is also completely verified by Lyapunov-based method.

Index Terms

Magnetic levitation systems, radial basis function neural network, terminal sliding mode control, nonlinear control systems.

I. INTRODUCTION

Magnetic levitation Systems (MLSs) have been widely applied for many real applications in the industrial field, such as rocket-guiding projects, gyroscopes, contactless melting, frictionless bearings, magnetic bearings, maglev trains, wafer distribution systems, vibration isolation systems, micro-robots, and so on. The fact is that MLSs have unstable characteristic and are represented by high nonlinear differential equations. Majority of control methods performed with MLSs are based on the model linearization method for the nominal working point, for example, PID control, the exact linearization control, and the state feedback linearization control [1]–[4]. With these control methods, the tracking performance can be deteriorated rapidly with increasing deviations from the nominal working point. However, to guarantee very long travel range and maintain good performance, it is necessary to be concerned about the nonlinear model instead of the linear one. Furthermore, the system parameter changes should be considered, such as the variation of suspending mass and the changes of resistance and inductance because of electromagnet heating process. Several nonlinear controls for MLSs have been launched in [5]–[10]. However, most of the studies have been only confirmed by simulation results, a few articles have been conducted in practice with no high control efficiency, or they have not been performed for trajectory tracking controls with more complicated reference paths such as sinusoidal or rest-to-rest. Therefore, control performance improvement for MLSs stills a challenge. Otherwise, all algorithms implemented in a real application usually were set up the initial position very nearby an indicated path, the magnitude of the sinusoidal path or the set-point has a small value. Furthermore, MLS has been only ensured asymptotic stability. For the finite-time stability, the control system for
MLS has certainly not been discussed, even by performance results.

Sliding mode control (SMC) with a robust and immutable characteristic is suitable for developing robust controllers to combat uncertainties and external disturbances. Therefore, SMC was commonly used for nonlinear systems such as spacecrafts, robotic manipulators, inverted pendulum, and marine surface vessels. However, SMC cannot guarantee invariant properties in the approach phase, in which there is the presence of unknown uncertain components [11]. Moreover, Most of SMCs developed so far are based on the theory of asymptotic Lyapunov, especially the linear switching manifolds that determine the desired control performance. The above theory is the asymptotic stability underpinned by the Lipschitz term. The nature of asymptotic stability implies that, during the variant of system dynamics, the closer to equilibrium, the slower the convergence of the state. This means that the system state will never reach equilibrium. So, if much higher convergence accuracy is required, a greater control force will be required and possibly not feasible once the device/driver facility is restricted. Terminal SMC (TSMC) is an enhanced variant of SMC. Those methods could provide convergence in finite-time. TSMC has attractive features such as robustness against uncertainty, fast convergence in finite-time, and high tracking accuracy [12]–[15]. Consequently, TSMC methods were widely performed for some real applications [16]–[19]. Unfortunately, the conventional TSMC still exists some shortcomings, including slow convergence rate when the position initials are far from the desired trajectory; the singularity problem is stated in [20]. Therefore, theoretical breakthroughs were proposed when introducing non-singular TSMC [13], [21] to solve singularity or fast TSMC [21]–[23] to accelerate the convergence rate. A development followed by a combination of NTSMC and FTSMC to produce the non-singular fast terminal sliding mode control (NFTSMC) [24]–[28] or finite-time control [49]. Those methods not only avoid the singularity but also reject glitch in the reaching stage with the arbitrary initial states. Accordingly, the control system constantly runs in sliding mode, and the immutable characteristic is always assured. Furthermore, the amount of time for the state variables to approach the equilibrium point can be arbitrarily set. Until now, very few NFTSMC methods are implemented for real MLSs [29], [30] or they are only simulated. In [29], [30], authors selected a nonsingular sliding surface with the fractional exponent in which the coefficients are required as positive odd integers and these nonsingular terminal sliding mode controls (NTSMCs) have slower convergence speed than fast terminal sliding mode control (FTSMC) In a new paper [31], a super-twisting combined with integral backstepping sliding mode control has been proposed for a MLS. Nonetheless, the validation of the controller is only analyzed by simulating their output in MATLAB/Simulink. The above-mentioned control approaches, i.e., SMC, TSMC, some variant TSMCs or NFTSMCs, applied high-frequency control switching that causes the unavoidable chattering phenomenon, which can lead to damage of actuators and of the system itself. Therefore, the chattering attenuation issue has become a popular topic. Some methods have been proposed for resolving this issue, such as the use of quasi-sliding mode [32], [33], low pass filters [34], [35], fuzzy-SMC [36], [37], neural networks-SMC (NN-SMC) [38]–[42], or super-twisting SMC methods [47], [48]. The method of using NN to reduce the chattering in the control input is to use a NN to approximate the unknown components that affect the system. Therefore, it is only necessary to use a reasonable sliding gain to compensate for the effects of the approximate error from NN. by using a small sliding value produces only negligible vibration in the control input system. Especially, radial basis function neural network (RBFNN) [43], [44] has many advantages. Compared with multilayer NN, RBFNN has a simpler design and faster convergence. Moreover, it has an online adjusting ability, good approximation, strong tolerance to input disturbance, classification, and so on. Therefore, it is suitable to apply to real systems without increasing the system’s computational burden. The enhanced control algorithms based fuzzy law can reduce the chattering phenomena while guaranteeing tracking performance. However, these methods significantly depend on the knowledge of experts. The enhanced adaptive control methods [50], [51] or full-order SMC combined with adaptive control [52] can also reject/reduce chattering behavior.

In our paper, a metal sphere is suspended in the air by electromagnetic force from MLS. There is no mechanical connection between the sphere and MLS. This system is highly nonlinear and an inherently unstable open loop. Therefore, to develop a method in adjustment and reference trajectory tracking of MLS is very challenging. Consequently, this article proposes a completely different approach from the pre-existing methods (such as [29]–[31]) for MLSs. The implemented control method is expected to provide a control performance with significant improvement in the terms of chattering reduction, high tracking accuracy, fast convergence along with simple design for real applications. The experimental work is implemented for a real magnetic levitation system to demonstrate the superior efficiency of the proposed adaptive neural terminal sliding mode control. The presented algorithm has important contributions as follows.

1) The structure of the presented control methodology is not too complex to be easily applied to real systems.
2) The presented controller inherits the advantages of both NFTSMC and RBFNN methods such as fast convergence speed regardless of the initial value, the ability to approximate the unknown functions of the system.
3) The combination of NFTSMC and RBFNN produces control inputs with lesser oscillation than the individual NFTSMC methods.
4) The effectiveness of the presented method has been experimentally verified.
5) The presented method provides high tracking accuracy for MLSs.
produce an electromagnetic force driver. The current flowing in the electromagnetic coil will input. This control voltage is converted into current by a electromagnet at distance. Voltage signal is the control force. Position of metal sphere is measured by using infrared vertical line under the effect of electromagnetic and gravity. Therefore, the metal sphere will be moved along the positioning stage, and so on.

To find the most efficient controller among the 4 methods applied to the MLSs including NFTSMC1, NFTSMC2, RBFNN-NFTSMC1, and RBFNN-NFTSMC2, their control performance will be compared. Here, the control methods such as NFTSMC1, and RBFNN-NFTSMC1 were briefly stated in an Appendix. NFTSMC2 and RBFNN-NFTSMC2 will be stated in detail in this article. The stable evidence of the proposed method is also completely verified by Lyapunov-based method.

The organization of our manuscript is as follows. Section II is the problem statement including the description of MLS, its nonlinear dynamic model, and the control target of this article. In Section III, the development of the controller and the stabilization of the system using TSMC and RBFNN are discussed in Section IV. Then, real-time implementation of the designed control method applying MATLAB/Simulink, Real Time Windows Target, Microsoft Visual C++ Professional, Control Toolbox, Real Time Workshop (RTW) software along with experimental results are presented in Section V.

II. PROBLEM STATEMENT

A. SYSTEM DESCRIPTION

Let us consider MLS displayed in Fig. 1. A metal sphere with mass \( m \) is positioned along the vertical line of the electromagnet at distance \( y \). Voltage signal is the control input. This control voltage is converted into current by a driver. The current flowing in the electromagnetic coil will produce an electromagnetic force \( F(y, I) \) to magnetize metal sphere. Therefore, the metal sphere will be moved along the vertical line under the effect of electromagnetic and gravity force. Position of metal sphere is measured by using infrared transmitters and detectors as shown in Fig. 1.

According to [6], [7], a simplest nonlinear model of MLS is presented as:

\[
m\ddot{y} = mg - F(y, I),
\]

where \( g \) denotes the acceleration due to the gravity, electromagnetic force \( F(y, I) \) in terms of winding current \( I \) and position of the metal sphere is computed as:

\[
F(y, I) = \lambda \left( \frac{I}{y} \right)^2,
\]

where \( \lambda \) is a constant depending on the coil electromagnet parameters.

The winding current and input voltage have the following linear relation:

\[
I = KU.
\]

Substituting (2) and (3) into (1) gives:

\[
m\ddot{y} = mg - \lambda K^2 \frac{1}{y^2} U^2.
\]

Dividing the two sides of Eq. (4) by \( m \) and letting \( \mu = \frac{\lambda K^2}{m} \), Eq. (4) becomes:

\[
\ddot{y} = g - \frac{\mu}{y^2} U^2.
\]

The parameter \( \mu \) is not precisely known, but it can be obtained by the approximated approaches. In addition, Eqs. (2) and (3) can also be obtained by reasonable assumptions. Thus, system (4) can be rewritten in general form as:

\[
\ddot{y} = g - \frac{\hat{\mu}}{y^2} U^2 + \Delta(y, t),
\]

where \( \hat{\mu} \) is estimate value of \( \mu \), and \( \Delta(y, t) \) is a function of dynamical uncertainty and exterior perturbation.

**Assumption 1:** The uncertain terms are assumed to be bounded by:

\[
|\Delta(y, t)| \leq \bar{\Delta},
\]

where \( \bar{\Delta} \) is a positive constant.

B. STUDY MOTIVATION

In our paper, a metal sphere is suspended in the air by electromagnetic force from MLS. There is no mechanical connection between the sphere and MLS. This system is highly nonlinear and an inherently unstable open loop. Therefore, to develop a method in adjustment and reference trajectory tracking of MLS is very challenging. Therefore, the target of this article is to develop a new, robust control method for MLSs in presence of dynamical uncertainties and external disturbances to further improve control performance such as chattering reduction in the control signal, high tracking accuracy, and fast convergence along with simple design method.
III. DESIGN OF THE PROPOSED CONTROL ALGORITHM

A. DESIGN OF THE SLIDING MANIFOLD

The tracking position error is defined as $y_e = y - y_r$ in which $y_r$ represents the reference trajectory. Therefore, the proposed sliding manifold is launched as follows:

$$s = \dot{y}_e + \frac{\alpha}{1 + e^{-\eta_1(y_e - \varepsilon)}}y_e$$
$$+ \frac{\beta}{1 + e^{\eta_2(y_e - \varepsilon)}}|y_e|^\delta \text{sign}(y_e),$$

(8)

where $s$ is the proposed sliding mode manifold, $\alpha$, $\beta$, $\eta_1$, $\eta_2$ are the positive constants, $0 < \delta < 1$, and $\varepsilon = \left(\frac{\beta}{\alpha}\right)^{\frac{1}{1-\delta}}$.

The stable evidence of the proposed sliding manifold is given as follows.

Once $s = 0$, then Eq. (8) becomes:

$$\dot{y}_e = \frac{\alpha}{1 + e^{-\eta_1(y_e - \varepsilon)}}y_e$$
$$- \frac{\beta}{1 + e^{\eta_2(y_e - \varepsilon)}}|y_e|^\delta \text{sign}(y_e).$$

(9)

**Evidence:**

Let us consider the following Lyapunov function:

$$V_1 = \frac{1}{2}y_e^2.$$  

(10)

Taking derivative of Eq. (10) with respect to time and noting Eq. (9), we can gain:

$$\dot{V}_1 = y_e \dot{y}_e$$
$$= y_e \left( -\frac{\alpha}{1 + e^{-\eta_1(y_e - \varepsilon)}}y_e$$
$$- \frac{\beta}{1 + e^{\eta_2(y_e - \varepsilon)}}|y_e|^\delta \text{sign}(y_e) \right)$$
$$= -\frac{\alpha}{1 + e^{-\eta_1(y_e - \varepsilon)}}y_e^2 - \frac{\beta}{1 + e^{\eta_2(y_e - \varepsilon)}}|y_e|^\delta + 1$$

\leq 0. \quad \text{ (11)}

Obviously, $\dot{V}_1 \leq 0$ and $V_1 > 0$. Therefore, control error variable $y_e$, $\dot{y}_e$ can reach the equilibrium point.

B. DESIGN OF TERMINAL SLIDING MODE CONTROL

The time derivative of the proposed TSM manifold in Eq. (8) is calculated as:

$$\dot{s} = \ddot{y}_e + \frac{2\alpha}{1 + e^{-\eta_1(y_e - \varepsilon)}}\dot{y}_e$$
$$+ \frac{2\alpha \eta_1 y_e \text{sign}(y_e) e^{-\eta_1(y_e - \varepsilon)}}{(1 + e^{-\eta_1(y_e - \varepsilon)})^2}y_e$$
$$+ \frac{2\beta \delta}{1 + e^{\eta_2(y_e - \varepsilon)}}|y_e|^{\delta-1} \dot{y}_e$$
$$- \frac{2\beta \eta_2 e^{\eta_2(y_e - \varepsilon)}}{(1 + e^{\eta_2(y_e - \varepsilon)})^2}|y_e|^\delta.$$  

(12)

Substituting Eq. (6) into Eq. (12) gives:

$$\dot{s} = G - \ddot{y}_e + \Delta (y, t) - \ddot{y}_e + \frac{2\alpha}{1 + e^{-\eta_1(y_e - \varepsilon)}}\dot{y}_e$$
$$+ \frac{2\alpha \eta_1 y_e \text{sign}(y_e) e^{-\eta_1(y_e - \varepsilon)}}{(1 + e^{-\eta_1(y_e - \varepsilon)})^2}y_e$$
$$+ \frac{2\beta \delta}{1 + e^{\eta_2(y_e - \varepsilon)}}|y_e|^{\delta-1} \dot{y}_e$$
$$- \frac{2\beta \eta_2 e^{\eta_2(y_e - \varepsilon)}}{(1 + e^{\eta_2(y_e - \varepsilon)})^2}|y_e|^\delta.$$  

(13)

For the desired control object, TSMC is developed as follows:

$$U = \sqrt{\frac{y^2}{\mu}} (u_{eq} + u_r).$$

(14)

Here, the term of $u_{eq}$ is designed as:

$$u_{eq} = g + \frac{2\alpha}{1 + e^{-\eta_1(y_e - \varepsilon)}}\dot{y}_e$$
$$+ \frac{2\alpha \eta_1 y_e \text{sign}(y_e) e^{-\eta_1(y_e - \varepsilon)}}{(1 + e^{-\eta_1(y_e - \varepsilon)})^2}y_e$$
$$+ \frac{2\beta \delta}{1 + e^{\eta_2(y_e - \varepsilon)}}|y_e|^{\delta-1} \dot{y}_e$$
$$- \frac{2\beta \eta_2 e^{\eta_2(y_e - \varepsilon)}}{(1 + e^{\eta_2(y_e - \varepsilon)})^2}|y_e|^\delta - \ddot{y}_d,$$  

(15)

and the term of $u_r$ is selected as:

$$u_r = \sigma s + (\ddot{\hat{A}} + \kappa) \text{sign}(s).$$

(16)

To check the correctness and stability of TSM signal system, the following evidence will be given.

**Evidence:**

Applying TSM signal system (14) - (16) to Eq. (13) gains:

$$\dot{s} = \Delta (y, t) - (\ddot{\hat{A}} + \kappa) \text{sign}(s) - \sigma s. \quad \text{ (17)}$$

Selecting the following Lyapunov function, then taking its time derivative, we obtain:

$$\dot{V}_2 = s \dot{s}.$$  

(18)

Adding Eq. (17) to Eq. (18) yields:

$$\dot{V}_2 = s (\Delta (y, t) - (\ddot{\hat{A}} + \kappa) \text{sign}(s) - \sigma s)$$
$$= \Delta (y, t) s - \ddot{\hat{A}} |s| - \kappa |s| - \sigma s^2$$
$$\leq -\kappa |s| - \sigma s^2$$
$$\leq 0.$$

(19)

Obviously, $\dot{V}_2 \leq 0$ and $V_2 > 0$. It means that $s$, $\dot{s}$ can reach the sliding manifold ($s \to 0$ and $\dot{s} \to 0$) and $y_e$, $\dot{y}_e$ can reach the equilibrium point ($y_e \approx 0$ and $\dot{y}_e \approx 0$). Commonly, it is not easy to precisely yield the model of dynamical uncertainties or disturbances in providing an exact function to the nominal control. Moreover, the boundaries of those unknown components are also challenging. To overcome the mentioned challenges, an RBFNN is applied to approximate unknown functions, while an adaptive control method will estimate the unknown boundary values of the estimated errors generated by NN. RBFNN is stated concisely as follows.
C. RADIAL BASIS FUNCTION NEURAL NETWORK

RBFNN has a simple structure with three layers: input layer, hidden layer and output layer as illustrated in Fig. 2. Compared with multilayer NN, RBFNN has a simpler design and faster convergence. Moreover, it has online adjusting ability, good approximation, strong tolerance to input disturbance, classification, and so on. Therefore, it is suitable for applying to real systems.

RBFNN is adaptively applied to approximate the unknown dynamical uncertainties and external disturbances, as follows:

\[ N^*(x) = \Delta(y,t). \tag{20} \]

We set \( \hat{N}^*(x) \) as an estimate function of \( N^*(x) \). \( \hat{N}^*(x) \) and \( N^*(x) \) are expressed, respectively by

\[ \hat{N}^*(x) = \hat{\varphi}^T \Psi(x), \tag{21} \]

\[ N^*(x) = \varphi^T \Psi(x) + e. \tag{22} \]

Here, \( \hat{\varphi} \) is the adaptable parameter vector. \( \varphi \in \mathbb{R}^{n \times m} \) denotes the weight matrix, \( \Psi(x) \) denotes the nonlinear function of the hidden nodes, and \( e \in \mathbb{R}^d \) denotes an estimate error. The RBFNN input is chosen as \( x = [y_e, \dot{y}_e]^T \).

We select a Gaussian function for the nonlinear function of the hidden nodes as:

\[ \Psi(x) = \exp \left( -\frac{(x - \eta_l)^T}{d^2} (x - \eta_l) \right), \quad l = 1, 2, \ldots, m \tag{23} \]

where, \( d \) and \( \eta \) stand for width and centre of Gaussian function, respectively.

The optimal parameter \( \varphi^* \) is defined by:

\[ \varphi^*_N = \arg \min \left\{ \sup_{x \in \Theta_x} \left| N^*(x) - \hat{N}^*(x, \hat{\varphi}) \right| \right\}. \tag{24} \]

Consequently, RBFNN (20) and (21) can correctly estimate the arbitrary function of \( N^*(x) \) which will be stated in Lemma 1.

**Lemma 1:** For any given real continuous function \( N^*(x) \) on the compact set \( \Theta_x \in \mathbb{R}^d \) and arbitrary positive constant \( e > 0 \), there exist a NN approximator \( \hat{N}^*(x) \) which holds a similar formula as Eqs. (20) and (21) such that

\[ \sup_{x \in \Theta_x} \left| N^*(x) - \hat{N}^*(x, \hat{\varphi}) \right| < e. \tag{25} \]

D. DESIGN OF AN ADAPTIVE NEURAL TERMINAL SLIDING MODE CONTROL

To obtain the expected control performance with the effective elimination of uncertain terms effects and updating ability of TSMC, the adaptive TSMC combined RBFNN is developed as follows

\[ U = \frac{\sqrt{\gamma^2}}{\tilde{\mu}} (u_{eq} + u_{ar} + u_c). \tag{26} \]

The term of \( u_{eq} \) is designed as:

\[ u_{eq} = g + \frac{2\alpha}{1 + e^{-\eta_1(y_e|y_e| - \varepsilon)}} \dot{y}_e + \frac{2\alpha \eta_1 \dot{y}_e \text{sign}(y_e) e^{-\eta_1(y_e|y_e| - \varepsilon)}}{(1 + e^{-\eta_1(y_e|y_e| - \varepsilon)})^2} y_e + \frac{2\beta \delta}{1 + e^{\eta_2(\dot{y}_e|\dot{y}_e| - \varepsilon)}} |y_e|^{\delta - 1} y_e - \frac{2\beta \eta_2 \dot{y}_e e^{\eta_2(\dot{y}_e|\dot{y}_e| - \varepsilon)}}{(1 + e^{\eta_2(\dot{y}_e|\dot{y}_e| - \varepsilon)})^2} |y_e|^{\delta - 2} \dot{y}_e, \tag{27} \]

the term of \( u_{ar} \) is selected as:

\[ u_{ar} = \sigma s + \tilde{\kappa} \text{sign}(s), \tag{28} \]

and

\[ u_c = \varphi^T \Psi(x). \tag{29} \]

The adaptive updating rule of sliding gain and RBFNN are given, respectively

\[ \dot{\kappa} = \omega^{-1} |s|, \tag{30} \]

\[ \dot{\varphi} = \sigma^{-1} s \Psi(s), \tag{31} \]

where \( \dot{\kappa} \) is the estimate value of \( \kappa \), \( \kappa \) is selected to be always greater the estimated error of RBFNN as follows: \( |s| < \kappa \). \( \omega \) and \( \sigma \) are adaption gains.

To check the correctness and stability of TSM signal system, the following evidence will be given.

**Evidence:**

We respectively determine the estimated error of the adaptive law and NN weight estimate error as:

\[ \tilde{\kappa} = \kappa - \kappa \quad \text{and} \quad \tilde{\varphi} = \varphi - \varphi. \tag{32} \]

By using Eq. (22), the time derivative of the sliding manifold in Eq. (13) becomes:

\[ \dot{s} = G - \frac{\dot{\mu}}{\gamma^2} U^2 + \varphi^T \Psi(x) + e + \frac{2\alpha}{1 + e^{-\eta_1(y_e|y_e| - \varepsilon)}} \dot{y}_e + \frac{2\alpha \eta_1 \dot{y}_e \text{sign}(y_e) e^{-\eta_1(y_e|y_e| - \varepsilon)}}{(1 + e^{-\eta_1(y_e|y_e| - \varepsilon)})^2} y_e + \frac{2\beta \delta}{1 + e^{\eta_2(\dot{y}_e|\dot{y}_e| - \varepsilon)}} |y_e|^{\delta - 1} y_e - \frac{2\beta \eta_2 \dot{y}_e e^{\eta_2(\dot{y}_e|\dot{y}_e| - \varepsilon)}}{(1 + e^{\eta_2(\dot{y}_e|\dot{y}_e| - \varepsilon)})^2} |y_e|^{\delta - 2} \dot{y}_e. \tag{33} \]
Substituting the control input from Eqs. (26) - (29) into Eq. (33) yields:

\[ \dot{s} = \dot{\varphi}^T \Psi(x) + e - \hat{k} \text{sign}(s) - \sigma s. \quad (34) \]

For stabilization proof of the control input from Eqs. (26) - (29), the positive-definite Lyapunov functional is selected as:

\[ V_3 = \frac{1}{2} s^2 + \frac{1}{2} \omega_k^2 + \frac{1}{2} \sigma \dot{\varphi}^2. \quad (35) \]

The time derivative of Eq. (35) is calculated along with the result of Eq. (32) as:

\[ \dot{V}_3 = s \dot{s} + \omega (\hat{k} - k) \dot{k} - \sigma \dot{\varphi}. \quad (36) \]

Substituting Eqs. (30), (31), and (34) into Eq. (36) gives

\[ \dot{V}_3 = s \left( \dot{\varphi}^T \Psi(x) + e - \hat{k} \text{sign}(s) - \sigma s \right) \\
+ \omega \left( \hat{k} - k \right) \dot{k} - \sigma \dot{\varphi} \\
= s \left( \dot{\varphi}^T \Psi(x) + e - \hat{k} \text{sign}(s) - \sigma s \right) \\
+ \left( \hat{k} - k \right) |s| - \dot{\varphi} s \Psi(s) \\
= s \dot{\varphi}^T \Psi(x) + es - \hat{k} |s| + \hat{k} |s| \\
- \kappa |s| - \sigma s^2 - \dot{\varphi} s \Psi(s) \\
= es - \kappa |s| - \sigma s^2 < 0. \quad (37) \]

The proof of stability is completed.

**Remark 1**: For the convenience of naming in the analysis of experimental results, we call the controller in Eqs. (14) - (16) named NFTSMC2 and the controller in Eqs. (26) - (31) named RBFNN-NFTSMC2.

**Remark 2**: The presented controller is not only applicable to MLSs but also to a class of second-order nonlinear systems such as robot manipulators, inverted pendulum, anti-synchronization of chaotic complex systems, piezo positioning stage, and so on.

**IV. EXPERIMENTAL RESULTS AND DISCUSSION**

Experimental study of an MLS has been implemented to investigate the effectiveness of the proposed control algorithm. We set up the system (6) with the nominal parameters according to [6], reported in Table 1:

| System parameter | Value | Unit |
|------------------|-------|------|
| \( \eta \)       | 9.81  | \( \text{m/s}^2 \) |
| \( m \)          | 0.02  | \( \text{kg} \) |
| \( \lambda \)     | \( 2.48315625 \times 10^{-5} \) | \( \text{Nm}^2/\text{A}^2 \) |
| \( K \)          | \( 1.05 \) | \( \text{A/V} \) |
| \( \mu \)        | \( 0.00136884 \) | \( (\text{N.m}^2)/(\text{kg.V}^2) \) |

The metal sphere is controlled to track the following reference trajectory:

\[ y_{r1} = 15 \text{ (mm)}, \quad (38) \]
\[ y_{r2} = 15 + 2.5 \sin(0.4\pi t) \text{ (mm)}. \quad (39) \]

For Eq. (39), it means that \( y_{r2\min} = 12.5 \text{ (mm)} \), and \( U_{max} < 4.5 \text{ (V)} \) is the maximum control voltage. Moreover, a real value of \( \mu = 0.00134557 \) is assumed according to the experimental results in [6]. Accordingly, the following suitable assumption of the uncertain terms is defined:

\[ |\Delta(y, t)| \leq \frac{\mu - \hat{\mu}}{y_{\min}^2} U_{max}^2 = 3.3 = \bar{\Delta} \quad (40) \]

with the initial value of \( y_0 = 26 \text{ (mm)} \).

The MLS in experiments was designed by Feedback Instrument [45] shown in Figure 4. Experimental platform in Fig. 4 includes a mechanical component (symbolized as number 1), an analogue control interface (symbolized as number 2),
To implement the proposed method, the program of the proposed system is coded in the PC using software tools from MathWorks Inc., including MATLAB/Simulink, Real Time Windows Target, Microsoft Visual C++ Professional, Control Toolbox, and Real Time Workshop (RTW). In addition, to achieve the executable file from the control law model, the essential processes are performed in Fig. 5 [45] as follows: 1) The control algorithm is designed by using the MATLAB/Simulink of Mathworks; 2) Using Real Time Workshop to build a C++ source program of the proposed controller from Simulink; 3) C++ compilers will build and connect the created program by RTW to generate an executable program; 4) Through I/O board, Real-Time Windows Target transfers interfaces and the executable program to hardware device; 5) the bi-directional flow of signals in / out from the model and I / O from the I / O panel will be administered by Real-Time Windows Target.

To find the most efficient controller among the 4 methods applied to the MLSs including NFTSMC1, NFTSMC2, RBFNN-NFTSMC1, and RBFNN-NFTSMC2, their control performance was compared. Here, the control methods such as NFTSMC1, and RBFNN-NFTSMC1 were briefly stated in an Appendix. NFTSMC2 and RBFNN-NFTSMC2 were stated in detail in this article. There is a remark to note is that all four control methods in this article have not been applied to a real MLS. Therefore, all four methods are first applied to the above MLS. This also is a new and important contribution to our article.

The selection of control parameters for four control algorithms is reported in Table 2. The experimental performance under four control methods for a metal sphere by tracking the reference trajectories in Eqs. (38) - (39) is shown in Figs. 5 - 9. With a comparison between the real trajectories and the desired trajectories of the metal sphere, it is seen that all four controllers
FIGURE 7. Control input signals of four controllers.

FIGURE 8. The uncertainties compensation value of RBFNN-NFTSMC1 and RBFNN-NFTSMC2.

TABLE 2. The control parameters.

| Controller          | Symbol                  | Value                     |
|---------------------|-------------------------|---------------------------|
| NFTSMC1             | $\alpha$, $\beta$, $\delta$, $\sigma$, $\Delta + \kappa$ | 10, 5, 1.5, 50, 4         |
| NFTSMC2             | $\alpha$, $\beta$, $\eta_1$, $\eta_2$, $\epsilon$, $\sigma$, $\Delta + \kappa$ | 2, 1.2, 0.6, 1.2, 1.3, 0.21213, 40, 4 |
| RBFNN-NFTSMC1       | $m$, $d$, $\eta_0$, $\omega$ | 10, 5, 1.5, 50, 0.001     |
| RBFNN-NFTSMC2       | $\alpha$, $\beta$, $\delta$, $\eta_1$, $\eta_2$, $\epsilon$, $\sigma$, $\omega$ | 5, 0.1, 0.02$f^2$, 0.3$f^3$, 0.10$

seem to be provided good tracking performance as shown in Figs. 5a and 5b. However, in order to find the best controller of the 4 control methods, we need to consider in detail the tracking accuracy and convergence rate as shown in Figures 6a and 6b. Firstly, we investigate NFTSMC1 and NFTSMC2 and recognize that 1) NFTSMC1 has a fast convergence time with $t \approx 0.5s$ and tracking error on the order $10^{-3}mm$; 2) NFTSMC2 has a faster convergence time and a little smaller tracking error than NFTSMC1. Therefore, NFTSMC2 is selected to develop the proposed controller. Secondly, NFTSMC1 or NFTSMC2 is combined with RBFNN to generate RBFNN-NFTSMC1 or RBFNN-NFTSMC2. Then, they are compared with each other. From the compared performance in Figs. 6a and 6b, it is clear that RBFNN-NFTSMC2 has a better tracking accuracy and faster convergence speed than RBFNN-NFTSMC1. RBFNN-NFTSMC2 has a fast convergence time with $t \approx 0.5s$ and tracking error on the order $10^{-4}mm$. Consequently, in terms of tracking accuracy and convergence speed, RBFNN-NFTSMC2 is the best controller among four controllers.

Regarding the chattering problem, we found that the chattering phenomena of both NFTSMC1 and NFTSMC2 controllers are generated similarly when both controllers use the same sliding value (with $\Delta + \kappa = 4$) to compensate for total uncertainty (with $|\Delta (\gamma, t)| \leq 3.3$). RBFNN-NFTSMC1
and RBFNN-NFTSMC2 have smaller chattering phenomena than two above controllers because total uncertainty is compensated by RBFNN. Therefore, they only use a minor sliding gain with $\kappa = 0.001$. The uncertainties compensation value of RBFNN-NFTSMC1 and RBFNN-NFTSMC2 are shown in Fig. 8a (straight-line trajectory) and 8b (sinusoidal trajectory). Look at Fig. 8, it is seen that the compensated value significantly impacts the control performance.

The updated switching gain value of RBFNN-NFTSMC2 is shown in Fig. 9. As shown in Fig. 9, it is observed that this value is quickly adapted and stabilized when the system is stable.

From experimental performance, it is concluded that RBFNN-NFTSMC2 have best control performance among four control methods. The proposed controller provided fast convergence speed, high tracking accuracy, and less chattering phenomena.

V. SOME REMARKABLE CONCLUSION

This article proposed a completely different approach from the pre-existing methods for tracking control of uncertain MLSs along with experimental validation was implemented. By proposing a novel fast terminal sliding manifold function with the dynamic coefficients, the system state variables quickly converge the equilibrium point on the manifold function. Besides, a robust reaching control law combined with RBFNN compensator drives the system fast approaching the sliding manifold function regardless of whether the initial value is near or far from the sliding manifold and reduces the chattering of the conventional TSMC. With a design approach based on the combination of the proposed sliding manifold and the combined control law, the implemented control method provides a control performance with significant improvement in the terms of chattering reduction, high tracking accuracy, fast convergence along with simple design for real applications. The experimental work is implemented for a real magnetic levitation system to demonstrate the superior efficiency of the proposed TSMC. The stable evidence of the proposed method is also completely verified by Lyapunov-based method. From experimental performance, it is concluded that RBFNN-NFTSMC2 has best control performance among four compared control methods. The proposed controller provided fast convergence speed, high tracking accuracy, and less chattering phenomena.

The presented control method can be extended to a class of second-order nonlinear systems. Therefore, for future study, we will apply the proposed control algorithm to a real industrial robot which is currently available in our lab and will also propose a fault-tolerant control method for MLSs and robot systems.

APPENDICES

A. DESIGN OF NON-SINGULAR FAST TERMINAL SLIDING MODE CONTROL 1 (NFTSMC1)

The tracking position error is defined as in which represents the reference trajectory. Therefore, the sliding manifold is selected according to [46] as follows:

$$s = \dot{y}_e + \alpha y_e + \beta |y_e|^{\delta - 1} y_e.$$  \hspace{1cm} (41)

The time derivative of Eq. (41) is calculated as:

$$\dot{s} = \ddot{y}_e + \alpha \dot{y}_e + \beta \delta |y_e|^{\delta - 1} \dot{y}_e.$$  \hspace{1cm} (42)

Substituting system (6) into Eq. (42) gives:

$$\dot{s} = g - \frac{\mu}{y^2} U^2 + \Delta (y, t) - \ddot{y}_d$$

$$+ \left( \alpha + \beta \delta |y_e|^{\delta - 1} \right) \dot{y}_e.$$  \hspace{1cm} (43)

For the desired control object, TSMC input is designed as follows:

$$U = \frac{\sqrt{y^2 (u_{eq} + u_r)}}{\mu}.$$  \hspace{1cm} (44)
Here, the term of $u_{eq}$ is designed as:

$$u_{eq} = g - \frac{\dot{\mu}}{\sqrt{2}} U^2 \hat{y}_d + \left(\alpha + \beta \delta |y_e|^{\beta - 1}\right) \dot{y}_e. \quad (45)$$

and the term of $u_r$ is selected as:

$$u_r = \sigma s + (\hat{\Lambda} + \kappa) \text{sign}(s). \quad (46)$$

### B. DESIGN OF RADIAL BASIS FUNCTION NEURAL NETWORK NON-SINGULAR FAST TERMINAL SLIDING MODE CONTROL 1 (RBFNN-NFTSMC1)

RBFNN is adaptively applied to approximate the unknown dynamical uncertainties and external disturbances as Eqs. (20) - (22). Therefore, Eq. (43) is rewritten as:

$$\dot{s} = g - \frac{\dot{\mu}}{\sqrt{2}} U^2 + \varphi^T \Psi(x) - \hat{y}_d + \left(\alpha + \beta \delta |y_e|^{\beta - 1}\right) \dot{y}_e. \quad (47)$$

To obtain the expected control performance with the effective elimination of uncertain terms effects, TSMC combined RBFNN is designed as follows:

$$U = \sqrt{\frac{2}{\dot{\mu}}} \left(u_{eq} + u_r + u_c\right). \quad (48)$$

The term of $u_{eq}$ is designed as:

$$u_{eq} = g - \frac{\dot{\mu}}{\sqrt{2}} U^2 - \hat{y}_d + \left(\alpha + \beta \delta |y_e|^{\beta - 1}\right) \dot{y}_e. \quad (49)$$

Here, the term of $u_r$ is selected as:

$$u_r = \kappa \text{sign}(s) + \sigma s, \quad (50)$$

and

$$u_c = \varphi^T \Psi(x). \quad (51)$$

### REFERENCES

[1] A.-V. Duka, M. Dalūa, and S.-E. Oltean, “IMC based PID control of a magnetic levitation system,” *Procedia Technol.*, vol. 22, pp. 592–599, Jan. 2016.

[2] A. Ghosh, T. R. Krishnan, P. Tejasw, A. Mandal, J. K. Pradhan, and S. Ranasingh, “Design and implementation of a 2-DOF PID compensation for magnetic levitation systems,” *ISA Trans.*, vol. 53, no. 4, pp. 1216–1222, Jul. 2014.

[3] W. Barie and J. Chiasson, “Linear and nonlinear state-space controllers for magnetic levitation,” *Int. J. Syst. Sci.*, vol. 27, no. 11, pp. 1153–1163, Nov. 1996.

[4] F. Zhang and K. Suyama, “Nonlinear feedback control of magnetic levitation system by exact linearization approach,” in *Proc. Int. Conf. Control Appl.*, Sep. 1995, pp. 267–268.

[5] T. Bächle, S. Hentzel, and K. Graichen, “Nonlinear model predictive control of a magnetic levitation system,” *Control Eng. Pract.*, vol. 21, no. 9, pp. 1250–1258, Sep. 2013.

[6] A. El Hajaji and M. Ouladsine, “Modeling and nonlinear control of magnetic levitation systems,” *IEEE Trans. Ind. Electron.*, vol. 48, no. 4, pp. 831–838, Aug. 2001.

[7] C.-L. Kuo, T.-H.-S. Li, and N. R. Guo, “Design of a novel fuzzy sliding-mode control for magnetic ball levitation system,” *J. Intell. Robotic Syst.*, vol. 42, no. 3, pp. 295–316, Mar. 2005.

[8] B. Design of Radial Basis Function Neural Network Non-Singular Fast Terminal Sliding Mode Control 1 (RBFNN-NFTSMC1)
[33] M. Van, H.-J. Kang, and K.-S. Shin, “Novel quasi-continuous super-twisting high-order sliding mode controllers for output feedback tracking control of robot manipulators,” Proc. Inst. Mech. Eng., C. J. Mech. Eng. Sci., vol. 228, no. 17, pp. 3240–3257, Dec. 2014.

[34] D. Zhao, C. Li, and Q. Zhu, “Low-pass-filter-based position synchronization sliding mode control for multiple robotic manipulator systems,” Proc. Inst. Mech. Eng., J. J. Syst. Control Eng., vol. 225, no. 8, pp. 1136–1148, Dec. 2011.

[35] M.-L. Tseng and M.-S. Chen, “Chattering reduction of sliding mode control by low-pass filtering the control signal,” Asian J. Control, vol. 12, no. 3, pp. 392–398, Feb. 2010.

[36] Q. P. Ha, Q. H. Nguyen, D. C. Rye, and H. F. Durrant-Whyte, “Fuzzy sliding-mode controllers with applications,” IEEE Trans. Ind. Electron., vol. 48, no. 1, pp. 38–46, Feb. 2001.

[37] B. Yoo and W. Ham, “Adaptive fuzzy sliding mode control of nonlinear system,” IEEE Trans. Fuzzy Syst., vol. 6, no. 2, pp. 315–321, May 1998.

[38] A. Izadbaksh, S. Khorashadizadeh, and P. Kheirkhahan, “Tracking control of electrically driven robots using a model-free observer,” Robotica, vol. 37, no. 4, pp. 729–755, Apr. 2019.

[39] L. Qiao and W. Zhang, “Adaptive second-order fast nonsingular terminal sliding mode control for fully actuated autonomous underwater vehicles,” IEEE J. Ocean. Eng., vol. 44, no. 2, pp. 363–385, Apr. 2019.

[40] Q. V. Doan, T. D. Le, and A. T. Vo, “Synchronization full-order terminal sliding mode control for an uncertain 3-DOF planar parallel robotic manipulator,” Appl. Sci., vol. 9, no. 9, p. 1756, Apr. 2019.

[41] J. Yang, S. Li, J. Su, and X. Yu, “Continuous nonsingular terminal sliding mode control for systems with mismatched disturbances,” Automatica, vol. 49, no. 7, pp. 2287–2291, Jul. 2013.

[42] J. Fei and H. Ding, “Adaptive sliding mode control of dynamic system using RBF neural network,” Nonlinear Dyn., vol. 70, no. 2, pp. 1563–1573, Oct. 2012.

[43] H.-C. Lu, C.-H. Tsai, and M.-H. Chang, “Radial basis function neural network with sliding mode control for robotic manipulators,” in Proc. IEEE Int. Conf. Syst., Man Cybern., Oct. 2010, pp. 1209–1215.

[44] S. Elanayar V. T. and Y. C. Shin, “Radial basis function neural network with sliding mode control for an uncertain nonlinear system,” IEEE Trans. Syst. Man Cybern. B, Cybern., vol. 38, no. 5, pp. 1191–1205, Sep. 2008.

[45] H.-C. Lu, C.-H. Tsai, and M.-H. Chang, “Radial basis function neural network with sliding mode control for robotic manipulators,” in Proc. IEEE Int. Conf. Syst., Man Cybern., Oct. 2010, pp. 1209–1215.

[46] S. Elanayar V. T. and Y. C. Shin, “Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 4, pp. 834–846, Apr. 2016.

[47] M. B. Naumovic and B. R. Veselic, “Magnetic levitation system in control engineering education,” Facta Univ. Ser. Autom. Control Robot., vol. 7, no. 4, pp. 422–426, 2012.

[48] T. N. Truong and M.-S. Chen, “Chattering reduction of sliding mode control for nonlinear systems,” IEEE Trans. Ind. Electron., vol. 49, no. 9, pp. 1729–1735, Sep. 2002.

[49] M.-L. Tseng and M.-S. Chen, “Chattering reduction of sliding mode control for nonlinear systems,” IEEE Trans. Ind. Electron., vol. 49, no. 5, pp. 1071–1077, Oct. 2002.

[50] M.-L. Tseng and M.-S. Chen, “Chattering reduction of sliding mode control for nonlinear systems,” IEEE Trans. Ind. Electron., vol. 49, no. 5, pp. 1071–1077, Oct. 2002.

[51] H. Pan, H. Li, W. Sun, and Z. Wang, “Adaptive fault-tolerant compensation control and its application to nonlinear suspension systems,” IEEE Trans. Syst., Man, Cybern. Syst., vol. 50, no. 5, pp. 1766–1776, May 2020.

[52] A. T. Vo and H.-J. Kang, “Adaptive neural integral full-order terminal sliding mode control for an uncertain nonlinear system,” IEEE Access, vol. 7, pp. 42238–42246, 2019.