IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 26, 2019, accepted March 22, 2019, date of publication March 29, 2019, date of current version April 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2907565

Adaptive Neural Integral Full-Order Terminal Sliding Mode Control for an Uncertain Nonlinear System

ANH TUAN VO AND HEE-JUN KANG
School of Electrical Engineering, University of Ulsan, Ulsan 44610, South Korea

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

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1A3B03930496).

ABSTRACT This paper reports the design of a control system for a class of general nonlinear second-order systems. The significant problems of singularity and chattering phenomenon, which limit the use of the conventional terminal sliding mode control (TSMC) in real applications due to the order of the sliding surface, need to be addressed. In addition, the effects of disturbances and uncertainties need to be removed, and the response rates are increased. Therefore, the integral full-order terminal sliding mode (IFOTSM) surface was proposed. To track the specified trajectories with high accuracy, a control approach is developed for the class of general nonlinear second-order systems by utilizing an IFOTSM surface and an adaptive compensator. The unknown dynamic model is derived based on a radial basis function neural network (RBFNN). Consequently, our controller provides good performance with minimum position errors, robustness against uncertainties, and work without a precise dynamic model. The simulated examples were performed to analyze the effectiveness of the control approach for position pathway tracking control of a 2-DOF parallel manipulator.

INDEX TERMS Radial basis function neural network, adaptive compensator, full-order terminal sliding mode control, disturbance, uncertainty, the class of general nonlinear second-order systems, 2-DOF parallel manipulator.

I. INTRODUCTION

The increasingly rigorous performance requirements of industrial applications highlight the importance of enhanced control systems developed for uncertain nonlinear systems that are normally subject to various nonlinearities, external disturbances, and uncertainties. Studies on the class of general nonlinear second-order systems have proposed many control methods focused on attaining the desired performance against various uncertainties, including external disturbances. Sliding mode control (SMC) has been validated to provide high robustness against uncertainties and disturbances for nonlinear systems [1]–[4]. Accordingly, SMC is usually applied to industrial application systems [5]–[8]. Nonetheless, several challenges of the traditional SMC still exist such as the requirement of an exact dynamic model, singularity, chattering occurrence, and unidentified convergence time. Various studies have focused on treating these challenges. For the system state variables to reach the prescribed SMC surface within a definite time, the terminal sliding mode control (TSMC), based on the nonlinear sliding mode function, has been used [9], [10]. However, the TSMC convergence time is slower than the traditional SMC convergence time, and still encompasses a singularity phenomenon. To handle convergence time and a singularity glitch, numerous fast TSMC (FTSMC) [11], [12] and nonsingular TSMC (NTSMC) [13], [14] systems have been applied to magnetic levitation systems [15], chaos control [16], [17], and robotic manipulators [2], [18]–[20]. Private control manners such as FTSMC or NTSMC have only focused on the resolution of individual weaknesses or neglected to handle the other weaknesses of the traditional SMC. For that reason, the nonsingular fast TSMC (NFTSMC) has been developed for controlling uncertain nonlinear second-order

The associate editor coordinating the review of this manuscript and approving it for publication was Dusmanta Kumar Mohanta.

systems [8], [11], [18], [21]–[24]. NFTSMC can deal with several drawbacks of the traditional SMC or other control systems based on TSMC. Nonetheless, chattering is not removed by applying a high-frequency reaching control term to the control input of the above systems, which include TSMC, FTSMC, NTSMC, and NFTSMC. As a result, several useful control systems have been proposed by applying full-order sliding mode control (FOSMC) [25]–[28], or high-order sliding mode control (HOSMC) [29], [30].

Two of the major challenges in designing a control system according to SMC or TSMC is knowing the bounds of modeling disturbances and dynamic uncertainties and computing an exact dynamic model, which is not known in advance for practical systems. To approximate this unknown model, several computing attempts have been proposed, such as neural networks [8], [31]–[33] and fuzzy logic systems [7], [34], due to their approximation abilities.

In traditional SMC and traditional TSMC, the drawbacks have been considered individually or ignored. In response, this work focuses on the synchronized resolve of SMC and TSMC drawbacks, including the condition for an exact dynamic model, the existence of a singularity, chattering occurrence, and finite-time convergence.

Consequently, the objective is to develop a controller for the class of general nonlinear second-order systems. The suggested system has the following major advantages: 1) it inherits the benefits of RBFNN and IFOSMC, including good performance with minimum position errors, robustness against uncertainties, and work without a precise dynamic model; 2) it consists of a control input system with chattering reduction; 3) compared to RBFNN-SMC and RBFNN-TSMC, ARBFNN-IFOSMC provides better performance and stronger resistance against disturbances and uncertainties; and 4) stability and tracking error convergence of the class of general nonlinear second-order systems was fully confirmed by the Lyapunov benchmark.

The remainder of this report is arranged as follows. After the introduction, the preliminaries and problem formulations are stated, followed by the design approach for the proposed controller, where the proposed system is applied to allow position pathway tracking control simulation for a 2-DOF parallel manipulator. The tracking performance is compared with those of the RBFNN-SMC and RBFNN-TSMC to evaluate the effectiveness of the proposed control system. Finally, conclusions are provided.

II. PROBLEM FORMULATIONS

A class of general nonlinear second-order systems is considered as follows:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = F(x,t) + D(x,t) + Q(x,t)u_{in}, \end{cases} \tag{1}$$

where $x = \left[x_1, x_2\right]^T$ represents the system state vector, $F(x,t) \in R^n$, $Q(x,t) \in R^{n\times n}$ are the smooth nonlinear vector fields, and $D(x,t) \in R^n$ represents the disturbances and uncertainties.

The target of this article is to develop a control system such that the controlled variables of the system follow the designated trajectory with minimum position errors, robustness against uncertainties, and work without a precise dynamic model. Accordingly, the tracking positional error is defined as:

$$e_1 = x_1 - x_d. \tag{2}$$

The following assumption is essential for developing the control algorithm in the next part.

Assumption 1: We assume that the lumped uncertain components are first-order differentiable and have the existence of the defined positive constant satisfying the following condition:

$$\left|\dot{D}(x,t)\right| \le \Xi, \tag{3}$$

where Ξ is the defined positive constant.

III. DESIGN SYNTHESIS OF THE CONTROL SYSTEM

In this part, a control method is developed for the class of general nonlinear second-order systems, which is described by in the two following steps.

A. DESIGN OF THE INTEGRAL FULL-ORDER TERMINAL SLIDING MODE (IFOTSM) SURFACE

First, with the tracking position error from Eq. (2), an IFOTSM surface is proposed as:

$$s = \dot{e}_2 + \int_0^t \left(\omega_1 e_1^{[\mu_1]} + \omega_2 e_2^{[\mu_2]}\right) d\sigma, \tag{4}$$

where $s = [s_1, s_2, \ldots, s_n]^T \in R^{n\times 1}$ is a sliding variable,ω_1, ω_2 are positive constants, $0 < \mu_1 < 1$, $\mu_2 = 2\mu_1/(1+\mu_1)$, $e_1 = [e_{11}, e_{12}, \ldots, e_{1n}]^T \in R^{n\times 1}$ represents the tracking position errors, $e_2 = [e_{21}, e_{22}, \ldots, e_{2n}]^T \in R^{n\times 1}$ represents the tracking velocity error, $\dot{e}_2$ is the time derivative of e_2, and $e^{[\mu_{(\cdot)}]}$ is defined as follows:

$$e^{[\mu_{(\cdot)}]} = |e|^{\mu_{(\cdot)}}\, sgn\,[e]. \tag{5}$$

where $\mu_1 > 0$, $\mu_2 > 0$ and $sgn\,[e] = \begin{cases} 1 & \text{if } e > 0 \\ -1 & \text{if } e < 0 \\ 0 & \text{if } e = 0 \end{cases}$

With Eq. (2), Eq. (1) can be expressed in the following error state space form as:

$$\begin{cases} \dot{e}_1 = e_2 \\ \dot{e}_2 = F(x,t) + D(x,t) + Q(x,t)\,u_{in} - \ddot{x}_d. \end{cases} \tag{6}$$

Substituting Eq. (6) into the IFOTSM surface (4) yields:

$$s = F(x,t) + D(x,t) + Q(x,t)\,u_{in} - \ddot{x}_d + \int_0^t \left(\omega_1 e_1^{[\mu_1]} + \omega_2 e_2^{[\mu_2]}\right) d\sigma. \tag{7}$$

B. DESIGN OF THE INTEGRAL FULL-ORDER TERMINAL SLIDING MODE CONTROL (IFOTSMC)

For system (1) to operate with quality performance, the control input system is designed as follows:

$$u_{in} = -Q^{-1}(x,t)\left(u_{eq} + u_{sw}\right). \tag{8}$$

Here, the equivalent control term is designed as:

$$u_{eq} = F(x,t) - \ddot{x}_d + \int_0^t \left(\omega_1 e_1^{[\mu_1]} + \omega_2 e_2^{[\mu_2]}\right) d\sigma, \tag{9}$$

and the switching control term is designed as:

$$\dot{u}_{sw} = (\Xi + \varpi)\, sgn(s). \tag{10}$$

Accordingly, the following theorem is formed to complete the proof.

Theorem 1: Consider a class of general nonlinear second-order systems (1). If the proposed control input system is designed for system (1) as Eqs. (8)-(10), then system (1) is guaranteed to have stability.

Proof: Applying the control input system (8)-(10) to Eq. (7) gives:

$$\begin{aligned} s &= F(x,t) + D(x,t) - \begin{pmatrix} u_{sw} + F(x,t) - \ddot{x}_d \\ + \int_0^t \left(\omega_1 e_1^{[\mu_1]} + \omega_2 e_2^{[\mu_2]}\right) d\sigma \end{pmatrix} \\ &\quad -\ddot{x}_d + \int_0^t \left(\omega_1 e_1^{[\mu_1]} + \omega_2 e_2^{[\mu_2]}\right) d\sigma \\ &= D(x,t) - u_{sw}. \end{aligned} \tag{11}$$

Taking the time derivative of Eq. (11) gives:

$$\begin{aligned} \dot{s} &= \dot{D}(x,t) - \dot{u}_{sw} \\ &= \dot{D}(x,t) - (\Xi + \varpi)\, sgn(s). \end{aligned} \tag{12}$$

Consider the following Lyapunov function as follows:

$$L = \frac{s^T s}{2}. \tag{13}$$

With the result of Eq. (12), the time derivative of Eq. (13) is derived as

$$\begin{aligned} \dot{L} &= s^T \dot{s} \\ &= s^T \left(\dot{D}(x,t) - (\Xi + \varpi)\, sgn(s)\right) \\ &= \left(\dot{D}(x,t)\, s - \Xi |s|\right) - \varpi |s| \le -\varpi |s|. \end{aligned} \tag{14}$$

Based on Eq. (14), the requirement for the Lyapunov stability benchmark [35] is guaranteed, wherein proof of stability is confirmed.

Nonetheless, the design approach requires an exact dynamic mode of $F(x,t)$ and satisfies Assumption 1. It is not trivial to precisely estimate dynamic uncertainties, external disturbances, and provide an exact dynamic function in the control system. To handle these challenges, a robust control approach will be developed for the class of general nonlinear second-order systems based on an IFOTSM surface and RBFNN. Here, two nonlinear terms in the IFOTSM surface and an adaptive compensator will be used to compensate for the effects of the dynamic uncertainties, disturbances, and error from the RBFNN, while an RBFNN will be utilized to approximate an unknown dynamic model.

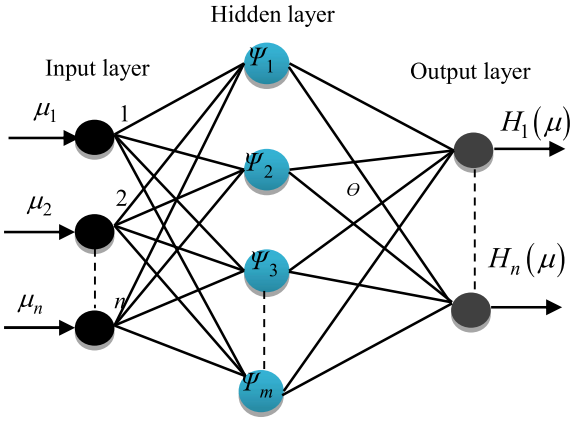

FIGURE 1. The architecture of an RBFNN.

C. RADIAL BASIS FUNCTION NEURAL NETWORK

RBFNNs have major advantages, including a highly parallel structure, robust tolerance to external disturbances and uncertainties, nonlinear function approximation [23], and online adaptation capability. Compared to other neural networks, RBFNN has a simpler and quicker convergence rate. An RBFNN includes three layers, input, hidden and output, which are shown in Figure 1.

The following RBFNN output is defined as:

$$H(\mu) = \theta^T \Psi(\mu) + \varepsilon(\mu), \tag{15}$$

where $\mu \in R^n$ and $H(\mu)$ correspond to the RBFNN input and output. $\theta^T \in R^{n\times m}$ represents the weight matrix linking the hidden layer and the output layer, $\Psi(\mu)$ represents the nonlinear function of the hidden nodes, and $\varepsilon(\mu) \in R^n$ represents an approximation error.

A Gaussian function is defined for the nonlinear function as follows:

$$\Psi(\mu) = \exp\left(-(\mu - \eta_l)^T (\mu - \eta_l)/d_l^2\right), l = 1, 2, \ldots, m, \tag{16}$$

where d and η correspond to the width and center of the Gaussian function.

D. DESIGN OF AN ADAPTIVE RADIAL BASIS FUNCTION NEURAL NETWORK INTEGRAL FULL-ORDER TERMINAL SLIDING MODE CONTROL (ARBFNN-IFOTSMC)

In this report, RBFNN is used to approximate the unknown dynamic model as follows:

$$h(x) = F(x,t). \tag{17}$$

Define $\hat{h}(x)$ as an approximated function of $h(x)$. $\hat{h}(x)$ can be described by a RBFNN, as follows:

$$\hat{h}(x) = \int_0^t \hat{\theta}^T \Psi(x)\, dt. \tag{18}$$

Here, $\hat{\theta}$ is the adaptable parameter vector.

The optimal parameter θ^* can be described as follows:

$$\theta_H^* = \arg\min\left\{\sup_{x\in\Theta_x}\left|h(x) - \hat{h}\left(x,\hat{\theta}\right)\right|\right\}. \tag{19}$$

Accordingly, RBFNN (18) can exactly approximate the arbitrary value of $h(x)$, which is given by the following Lemma [23].

Lemma 1: For arbitrary positive constant $\varepsilon > 0$ and any real continuous function $h(X)$ on the compact set $\Theta_X \in R^n$, there is a neural approximator existence $\hat{h}(X)$ that holds a similar form as Eq. (18), such that

$$\sup_{X\in\Theta_X}\left|h(X) - \hat{h}\left(X,\hat{\theta}\right)\right| < \varepsilon. \tag{20}$$

Consequently, the unknow dynamic model can be described as

$$\dot{e}_2 = \int_0^t \theta^{*T}\Psi(x)\,dt + Q(x,t)\,u_{in} - \ddot{x}_d + \Gamma, \tag{21}$$

where $\Gamma = D(x,t) + \varepsilon$ is the lumper uncertainty, including disturbances, dynamic uncertainties, and NN approximation error. To facilitate the next design step, the time derivative of the lumper uncertainty is assumed to be bounded by an unknown positive constant, $\left|\dot{\Gamma}\right| \le \Pi$.

The proposed control law is designed as follows:

$$u_{in} = -Q^{-1}(x,t)\left(u_{eq} + u_{asw}\right). \tag{22}$$

Here, the equivalent control law is constructed as

$$\tau_{eq}(t) = \int_0^t \hat{\theta}^T\Psi(x)\,dt - \ddot{x}_d + \int_0^t\left(\omega_1 e_1^{[\mu_1]} + \omega_2 e_2^{[\mu_2]}\right)d\sigma, \tag{23}$$

and u_{asw} is an adaptive compensator for substituting the control term of u_{sw} in Eq. (12), describing u_{asw} as

$$\dot{u}_{asw} = \left(\hat{\Pi} + \varpi\right) sgn(s). \tag{24}$$

The adaptive laws are designed as

$$\dot{\hat{\Pi}} = \gamma^{-1}|s|, \tag{25}$$

$$\dot{\hat{\theta}} = \kappa^{-1} s\Psi(x), \tag{26}$$

where $\hat{\Pi}$ is the estimated value of the design parameter Π, ϖ is a positive constant, and γ, κ indicate the adaptive parameters.

The control design approach for the robot system is described in Theorem 2 below.

Theorem 2: For the system (1), if suitable IFOTSM surfaces are proposed as (4) and the control input signal is constructed as (22)–(24) with its parameter updating rules designed as (25) and (26), then the stability of the system (1) is secured with the desired performance, and the tracking errors reach zero.

Proof: Identify the adaptive estimation error and NN weight approximation error, respectively, as follows:

$$\tilde{\Pi} = \hat{\Pi} - \Pi, \tag{27}$$

$$\tilde{\theta} = \theta^* - \hat{\theta}. \tag{28}$$

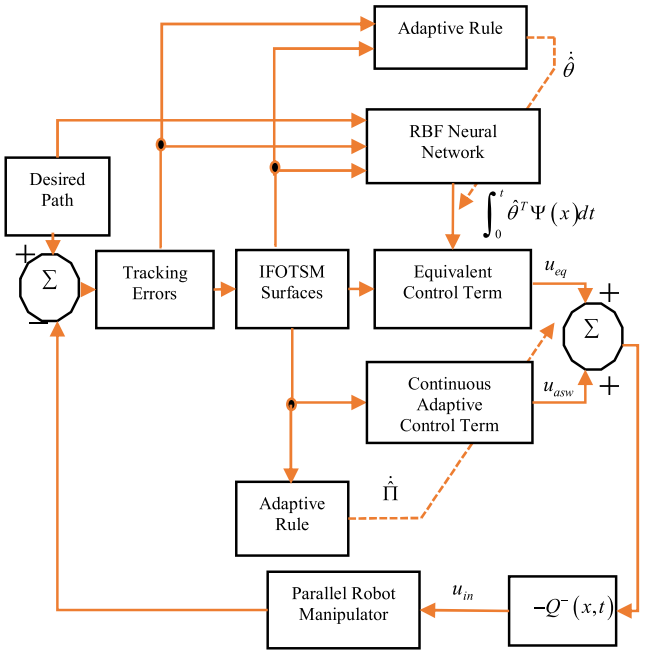

FIGURE 2. Block diagram of the proposed control scheme.

The sliding surface in Eq. (7) is rewritten as:

$$s = \int_0^t \theta^{*T}\Psi(x)\,dt + Q(x,t)\,u_{in} - \ddot{x}_d + \Gamma + \int_0^t\left(\omega_1 e_1^{[\mu_1]} + \omega_2 e_2^{[\mu_2]}\right)d\sigma. \tag{29}$$

Substituting control laws (22)–(24) into Eq. (29) provides:

$$s = \int_0^t \tilde{\theta}^T\Psi(x)\,dt - u_{asw} + \Gamma. \tag{30}$$

Taking the time derivative of Eq. (30) gives:

$$\dot{s} = \tilde{\theta}^T\Psi(x) - \left(\hat{\Pi} + \varpi\right) sgn(s) + \dot{\Gamma}. \tag{31}$$

The positive-definite Lyapunov functional is selected as:

$$L_2 = \frac{s^T s}{2} + \frac{\gamma\tilde{\Pi}^T\tilde{\Pi}}{2} + \frac{\kappa\tilde{\theta}^T\tilde{\theta}}{2}. \tag{32}$$

With the result of Eq. (31), the time derivative of Eq. (32) is derived as:

$$\begin{aligned}\dot{L}_2 &= s^T\dot{s} + \gamma\tilde{\Pi}^T\dot{\tilde{\Pi}} - \kappa\tilde{\theta}^T\dot{\hat{\theta}}\\ &= s^T\left(\tilde{\theta}^T\Psi(x) - \left(\hat{\Pi} + \varpi\right) sgn(s) + \dot{\Gamma}\right)\\ &\quad + \gamma\left(\hat{\Pi} - \Pi\right)\dot{\hat{\Pi}} - \kappa\tilde{\theta}^T\dot{\hat{\theta}}\\ &= s^T\tilde{\theta}^T\Psi(x) - \hat{\Pi}|s| - \varpi|s| + \dot{\Gamma}s\\ &\quad + \gamma\left(\hat{\Pi} - \Pi\right)\dot{\hat{\Pi}} - \kappa\tilde{\phi}^T\dot{\hat{\theta}}.\end{aligned} \tag{33}$$

Applying the updating laws (25)–(26) to (33) yields:

$$\begin{aligned}\dot{L}_2 &= -\hat{\Pi}|s| - \varpi|s| + \Pi s + \left(\hat{\Pi} - \Pi\right)|s|\\ &= -\varpi|s| + \left(\dot{\Gamma}s - \Pi|s|\right)\\ &\le -\varpi|s|.\end{aligned} \tag{34}$$

As shown in Eq. (34), if the constant of ϖ is selected to be greater than zero, $\dot{L}_2$ will be negative-definite. According to the Lyapunov benchmark [35], $\dot{L}_2$ becoming negative-definite implies that s and $\tilde{\Pi}$ reach zero, and the tracking error variables thus approach zero as well. Therefore, Theorem 3 is proven.

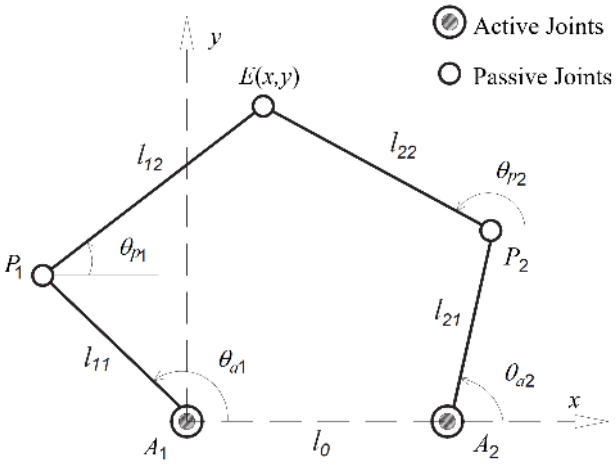

FIGURE 3. The kinematic illustration of the robotic system.

IV. NUMERICAL SIMULATION RESULTS

While the ARBFNN-IFOTSMC can be applied for a class of second-order nonlinear system such as serial robotic manipulators, parallel robotic manipulators, spacecraft, we consider a five-bar manipulator acting on a horizontal plane as an example that was presented in [36], [37] and its kinematic illustration was shown in Fig. 3. The dynamic model of the planar five-bar manipulator is given by [37]:

$$\hat{M}_a\ddot{\theta}_a + \hat{C}_a\dot{\theta}_a + \Delta\tau_a = \tau_a, \tag{35}$$

where $\theta_a = \left[\theta_{a1},\ \theta_{a2}\right]^T$ is the active joint angle vector; $\dot{\theta}_a = \left[\dot{\theta}_{a1},\ \dot{\theta}_{a2}\right]^T$ is the active joint velocity vector; $\ddot{\theta}_a = \left[\ddot{\theta}_{a1},\ \ddot{\theta}_{a2}\right]^T$ is the active joint acceleration vector; $\hat{M}_a \in R^{2\times2}$ is the estimated inertia matrix; $\hat{C}_a \in R^{2\times2}$ is the estimated centripetal Coriolis matrix; $\Delta\tau_a$ is the vector of modeling errors and uncertainties; and $\tau_a \in R^{2\times1}$ is the actuator output related to the control input u_{in}. The detailed computations of $\hat{M}_a$ and $\hat{C}_a$ were presented in [37]. The vector $\Delta\tau_a$ from Eq. (35) is presented as the following:

$$\Delta\tau_a = \Delta M_a\ddot{\theta}_a + \Delta C_a\dot{\theta}_a + F_a, \tag{36}$$

where ΔM_a and ΔC_a are the bounded modeling errors and F_a is the friction force.

The robot in Eq. (37) is rewritten with the following expression:

$$\ddot{\theta}_a = \hat{M}^{-1}(\theta_a)\left[\tau_a - \hat{C}_a\dot{\theta}_a - \Delta\tau_a\right]. \tag{37}$$

Then, we assign $u_{in} = \tau_a$ as the control input, $x = [x_1,\ x_2]^T$ as the state vector in which $x_1,\ x_2$ are corresponding to $\theta_a, \dot{\theta}_a \in R^{2\times1}$. The robotic dynamic of Eq. (35) can be described in the following state space form as:

$$\begin{cases}\dot{x}_1 = x_2\\ \dot{x}_2 = F(x,t) + D(x,t) + Q(x,t)u_{in},\end{cases} \tag{38}$$

where $F(x,t) = \hat{M}^{-1}(\theta_a)\left[-\hat{C}_a\dot{\theta}_a\right] \in R^{2\times1}$, $Q(x,t) = \hat{M}^{-1}(\theta_a) \in R^{2\times2}$ are the smooth nonlinear vector fields and $D(x,t) = \hat{M}^{-1}(\theta_a)[-\Delta\tau_a] \in R^{2\times1}$ presents the disturbances and uncertainties.

To evaluate the effectiveness of ARBFNN-IFOTSMC, its performance is compared with those of RBFNN-SMC and RBFNN-TSMC. The design of RBFNN-SMC and RBFNN-TSMC are represented in Appendices A and B, respectively. Simulation studies were performed on a MATLAB–Simulink environment with a fixed-step size of 10^{-3} and the mechanical model of the 2-DOF parallel manipulator is built using a SimMechanics toolbox. This 2-DOF parallel manipulator operates on the horizontal plane. Hence, the end-effector of the parallel manipulator is controlled to track a circular path on the XY plane. The link parameters in the mechanical model are set as Table 1.

TABLE 1. parameters of the mechanical model.

Depiction	Parameters	Value
Link lengths	$l_{11} = l_{21}$	*0.102(m)*
	$l_{12} = l_{22}$	*0.18(m)*
	l_0	*0.132(m)*
Distances from the joint to the center of mass of the links	r_{11}	*0.05(m)*
	r_{21}	*0.055(m)*
	$r_{12} = r_{22}$	*0.09(m)*
Masses of the links	m_{11}	*0.8(kg)*
	m_{21}	*0.78(kg)*
	m_{12}	*1.17(kg)*
	m_{22}	*1.2(kg)*
Inertia tensors of the links	I_{z11}	*0.0027(kg.m^2)*
	I_{z21}	*0.0031(kg.m^2)*
	$I_{z12} = I_{z22}$	*0.0013(kg.m^2)*

To establish the modeling errors ΔM_a, ΔC_a simulations were performed with different parameters, both in the mechanical model of the robot as well as in the controllers: $\hat{r}_{i1} = 0.9r_{i1}$ and $\hat{r}_{i2} = 0.9r_{i2}$ in which $\hat{r}_{i1}$, $\hat{r}_{i2}$ $(i = 1,\ 2)$ are utilized for determining $\hat{M}_a$, $\hat{C}_a$.

In the mechanical model, the frictions of the system involve viscous friction and Coulomb friction torques that are computed from the following formula:

$$f_{ai} = F_{ci}sgn\left(\dot{\theta}_{ai}\right) + F_{vi}\dot{\theta}_{ai}, \quad (i = 1,\ 2), \tag{39}$$

where the coefficients are selected as $F_{c1} = F_{c2} = 0.3$, and $F_{v1} = F_{v2} = 0.7$.

In addition, the 2-DOF parallel manipulator system is also troubled by external disturbance forces $d_a(t) = \left[d_{a1}(t)\ d_{a2}(t)\right]^T = \left[2\ 2\right]^T$ at $t = 3.0\ s$.

The selected control parameters of three different control methodologies: RBFNN-SMC, RBFNN-TSMC, and ARBFNN-IFOTSMC are stated in Table 2.

TABLE 2. selected parameters of the control methods.

Control Method	Control Parameters	Control Parameter Values
RBFNN-SMC (46)	α, K, K_V	$10, 5, diag(20,20)$
RBFNN-TSMC (54)	$\alpha, (1/\beta)$	$0.8, diag(2,2)$
	K_V, ϖ, Ξ	$diag(20,20), 0.1, 5$
The Proposed Method (22)-(26) (ARBFNN-IFOTSMC)	$\omega_1, \omega_2, \mu_1, \mu_2$	$20, 7, 0.5, 0.6$
	ϖ, γ, κ	$0.1, 0.05, 0.5 I_{7\times7}$

I is a unit matrix.

The RBFNN had ten neurons in the input layer, seven neurons in the hidden layer, and two neurons in the output layer. The weight of RBFNN is initialed with a zero value.

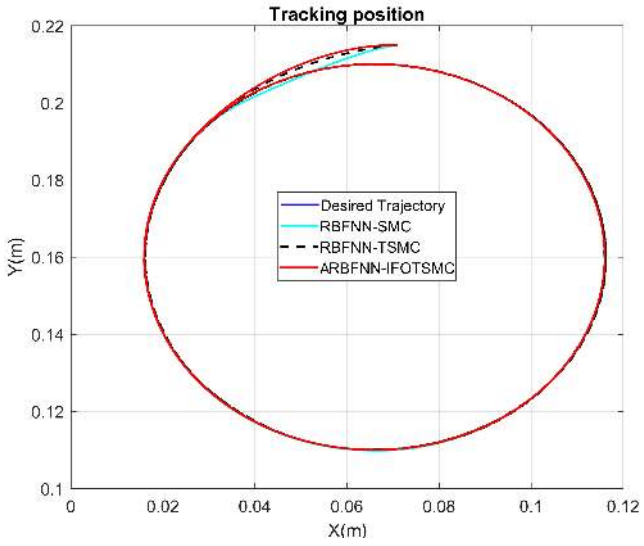

FIGURE 4. The desired trajectory and real trajectory of the end-effector.

The simulation studies were performed to compare the control methods in two terms of their positional accuracies and the resulting chattering phenomenon in their control input systems when the parallel manipulator tracked a desired circular path. Fig. 4 illustrates performance comparison of the position trajectory tracking when using three different control methodologies: RBFNN-SMC, RBFNN-TSMC, and ARBFNN-IFOTSMC. The end-effector of the robot manipulator has the initial position at the top of the designated circular path. For that reason, the initial position of the end effector of the robot manipulator is not on the specified circular trajectory, the real path of the robot in the operation had

FIGURE 5. The tracking errors of the end-effector in the X-direction.

FIGURE 6. The tracking errors of the end-effector in the Y-direction.

a malfunction at the starting of motion. The simulation results of the tracking errors of the end-effector on the X-direction and Y-direction are depicted in Figs. 5 and 6, respectively. These controllers can be used to track the specified trajectory. The tracking accuracy of the ARBFNN-IFOTSMC had the least amount of tracking errors, on the order of 10^{-6}–10^{-7}, in the presence of uncertainties and external disturbances.

The control input signals for all control types, including RBFNN-SMC, RBFNN-TSMC, and ARBFNN-IFOTSMC are depicted in Fig. 7. In Figs 7a and 7b, the RBFNN-SMC and RBFNN-TSMC offer a discontinuous control signal with serious chattering behavior. On the contrary, ARBFNN-IFOTSMC offers a continuous control signal for the robot manipulator, as depicted in Fig. 7c.

The adaptive parameters of the adaptive compensator in the proposed system are depicted in Fig. 8; alteration of the influences of external disturbances and uncertainties leads to adaptive parameters with corresponding alteration. These adaptive parameters will converge to constant values along with the system converging to the IFOTSM surface.

Remark 1: Throughout simulation analyses and comparison among those of RBFNN-SMC, RBFNN-TSMC, and ARBFNN-IFOTSMC, the simulated results and tracking performance comparison could be expected to exhibit the

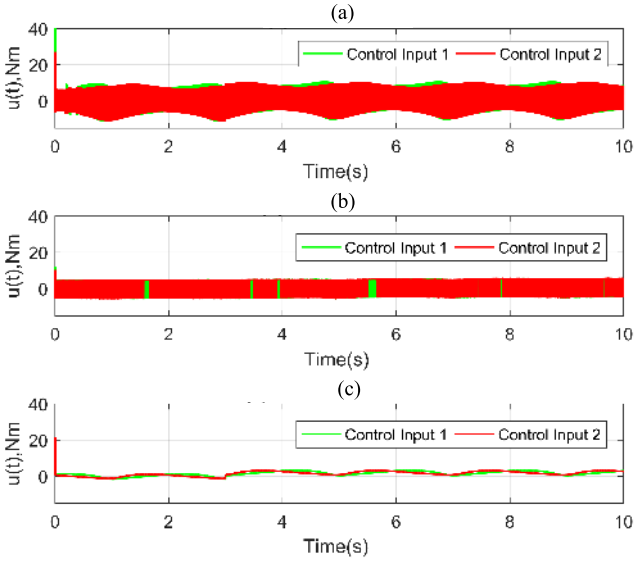

FIGURE 7. Control input signals: (a) RBFNN-SMC, (b) RBFNN-TSMC, and (c) ARBFNN-IFOTSMC.

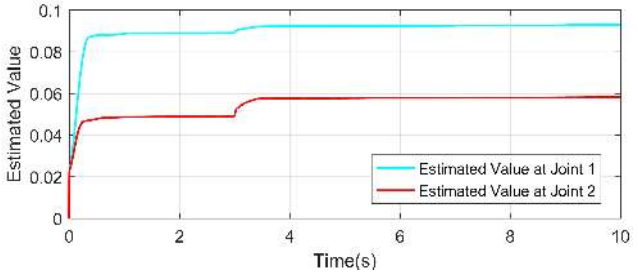

FIGURE 8. Variation of the adaptive gains at Joint 1 and Joint 2.

effectiveness and viability of our proposed control algorithm. In future studies, the ARBFNN-IFOTSMC will be applied to the real robot manipulator and compared with other state-of-the-art control systems to validate the effectiveness of this control methodology. We will also consider the effects of the measurement devices and errors associated with it.

V. CONCLUSIONS

This study reports the design of the control system for the class of general nonlinear second-order systems. The suggested system has the following major advantages: 1) it receives the advantages of both RBFNN and IFOSMC, including good performance with minimum position errors, robustness against uncertainties, and work with a precise dynamic model; 2) it consists of a control input system with chattering reduction; 3) ARBFNN-IFOSMC provides better performance and stronger resistance against disturbances and uncertainties compared to RBFNN-SMC and RBFNN-TSMC; and 4) stability and tracking error convergence of the class of general nonlinear second-order systems was fully confirmed by the Lyapunov benchmark.

APPENDIX A
DESIGN RBFNN-SMC

Set $x_1 - x_d$ as the tracking positional error and x_d as the desired trajectory values. Then, define the sliding variable and time derivative of the sliding variable as follows:

$$s = \dot{e} + \alpha e, \tag{40}$$

$$\dot{s} = \ddot{e} + \alpha \dot{e}, \tag{41}$$

where α is a positive constant.

With Eq. (38), Eq. (41) can be expressed as follows:

$$\begin{aligned} \dot{s} &= \dot{x}_2 - \ddot{x}_d + \alpha \left(\dot{x}_1 - \dot{x}_d \right) \\ &= F\left(x, t \right) + D\left(x, t \right) + Q\left(x, t \right) u_{in} - \ddot{x}_d + \alpha \left(\dot{x}_1 - \dot{x}_d \right). \end{aligned} \tag{42}$$

To obtain the desired performance for the robot system, the controller is designed as follows:

$$u_{in} = -Q^{-1}\left(x, t \right) \left(u_{eq} + u_{sw} \right). \tag{43}$$

The equivalent control signal of u_{eq} is computed in the case of $\dot{s} = 0$ and $D\left(x, t \right) = 0$. Consequently, the term of the equivalent control is designed as follows:

$$u_{eq} = F\left(x, t \right) - \ddot{x}_d + \alpha \left(\dot{x}_1 - \dot{x}_d \right) + K_V s, \tag{44}$$

and the switching control term is designed as:

$$u_{sw} = K sgn\left(s \right), \tag{45}$$

where K is a positive constant and K_V is a diagonal matrix.

We will utilize a neural network to approximate the nonlinear unknown dynamic function of the robotic system $F\left(x, t \right)$. Accordingly, the controller of Eq. (43) becomes

$$u_{in} = -Q^{-1}\left(x, t \right) \begin{pmatrix} \hat{\theta}^T \Psi\left(x \right) - \ddot{x}_d + \alpha \left(\dot{x}_1 - \dot{x}_d \right) \\ + K_V s + g sgn\left(s \right) \end{pmatrix}. \tag{46}$$

APPENDIX B
DESIGN RBFNN-TSMC

The following is the design approach of RBFNN-TSMC.

$e = x_1 - x_d$ is the tracking error and $\dot{e} = x_2 - \dot{x}_d$ is the tracking velocity error. Then, the NFTSM surface was selected:

$$s = e + \beta^{-1} sig\left(\dot{e} \right)^{\alpha}, \tag{47}$$

where $s = \left[s_1, s_2 \right]^T \in R^{2\times 1}$ is the sliding surface, $sig\left(\dot{e} \right)^{\alpha} = \left(\left| \dot{e}_1 \right|^{\alpha} sgn\left(\dot{e}_1 \right), \ldots, \left| \dot{e}_n \right|^{\alpha} sgn\left(\dot{e}_n \right) \right)$, $1 < \alpha < 2$.

For system (35) to operate with the desired performance, the controller is designed as follows:

$$u_{in} = -Q^{-1}\left(x, t \right) \left(u_{eq} + u_{sw} \right). \tag{48}$$

The equivalent control signal of u_{eq} is computed in the case of $\dot{s} = 0$ and $D\left(x, t \right) = 0$. Therefore, the time derivative of the sliding surface is described as follows:

$$\dot{s} = \dot{e} + \alpha \beta^{-1} \left| \dot{e} \right|^{\alpha - 1} \ddot{e}. \tag{49}$$

From Eq. (38), $\ddot{e}$ can be express as:

$$\ddot{e} = F\left(x, t \right) + D\left(x, t \right) + Q\left(x, t \right) u_{in} - \ddot{x}_d. \tag{50}$$

Substituting (50) into (49) gives:

$$\dot{s} = \dot{e} + \alpha\beta^{-1} |\dot{e}|^{\alpha-1} (F(x,t) + D(x,t) + Q(x,t) u_{in} - \ddot{x}_d). \quad (51)$$

Once the robot model $F(x,t)$ is exactly calculated, then the u_{eq} control input signal can be defined as follows:

$$u_{eq} = F(x,t) - \ddot{x}_d + \beta\alpha^{-1}\dot{e}^{2-\alpha} + K_V s. \quad (52)$$

The switching control term is designed as:

$$u_{sw} = (\Xi + \varpi) sgn(s), \quad (53)$$

where K_V is a diagonal matrix.

We will utilize a neural network to approximate the nonlinear unknown dynamic function of the robotic system $F(x,t)$. Accordingly, the controller of Eq. (48) becomes

$$u_{in} = -Q^{-1}(x,t) \begin{pmatrix} \hat{\theta}^T \Psi(x) - \ddot{x}_d + \beta \frac{1}{\alpha} \dot{e}^{2-\alpha} \\ + K_V s + (\Xi + \varpi) sgn(s) \end{pmatrix}. \quad (54)$$

REFERENCES

[1] Y. Chang, "Adaptive sliding mode control of multi-input nonlinear systems with perturbations to achieve asymptotical stability," *IEEE Trans. Autom. Control*, vol. 54, no. 12, pp. 2863–2869, Dec. 2009.

[2] S. Islam and X. P. Liu, "Robust sliding mode control for robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 58, no. 6, pp. 2444–2453, Jun. 2011.

[3] Q. Meng, T. Zhang, X. Gao, and J.-Y. Song, "Adaptive sliding mode fault-tolerant control of the uncertain stewart platform based on offline multibody dynamics," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 3, pp. 882–894, Jun. 2014.

[4] B. Yang, T. Yu, H. Shu, J. Dong, and L. Jiang, "Robust sliding-mode control of wind energy conversion systems for optimal power extraction via nonlinear perturbation observers," *Appl. Energy*, vol. 210, pp. 711–723, Jan. 2018.

[5] V. I. Utkin, *Sliding Modes in Control and Optimization*. Berlin, Germany: Springer Science & Business Media, 2013. doi: 10.1007/978-3-642-84379-2.

[6] C. Edwards and S. Spurgeon, *Sliding Mode Control: Theory and Applications*. Boca Raton, FL, USA: CRC Press, 1998.

[7] A. T. Vo, H.-J. Kang, and T. D. Le, "An adaptive fuzzy terminal sliding mode control methodology for uncertain nonlinear second-order systems," in *Proc. Int. Conf. Intell. Comput.*, 2018, pp. 123–135.

[8] A. T. Vo and H.-J. Kang, "An adaptive neural non-singular fast-terminal sliding-mode control for industrial robotic manipulators," *Appl. Sci.*, vol. 8, no. 12, p. 2562, 2018.

[9] Y. Tang, "Terminal sliding mode control for rigid robots," *Automatica*, vol. 34, no. 1, pp. 51–56, 1998.

[10] Y. Wu, X. Yu, and Z. Man, "Terminal sliding mode control design for uncertain dynamic systems," *Syst. Control Lett.*, vol. 34, no. 5, pp. 281–287, 1998.

[11] S. Mobayen, "Fast terminal sliding mode controller design for nonlinear second-order systems with time-varying uncertainties," *Complexity*, vol. 21, no. 2, pp. 239–244, 2015.

[12] T. Madani, B. Daachi, and K. Djouani, "Modular-controller-design-based fast terminal sliding mode for articulated exoskeleton systems," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 3, pp. 1133–1140, May 2017.

[13] S. Eshghi and R. Varatharajoo, "Nonsingular terminal sliding mode control technique for attitude tracking problem of a small satellite with combined energy and attitude control system (CEACS)," *Aerosp. Sci. Technol.*, vol. 76, pp. 14–26, May 2018.

[14] A. Safa, R. Y. Abdolmalaki, S. Shafiee, and B. Sadeghi, "Adaptive non-singular terminal sliding mode controller for micro/nanopositioning systems driven by linear piezoelectric ceramic motors," *ISA Trans.*, vol. 77, pp. 122–132, Jun. 2018.

[15] X.-T. Tran and H.-J. Kang, "Arbitrary finite-time tracking control for magnetic levitation systems," *Int. J. Adv. Robot. Syst.*, vol. 11, no. 10, p. 157, 2014.

[16] M. P. Aghababa, "Finite-time chaos control and synchronization of fractional-order nonautonomous chaotic (hyperchaotic) systems using fractional nonsingular terminal sliding mode technique," *Nonlinear Dyn.*, vol. 69, nos. 1–2, pp. 247–261, 2012.

[17] H. Wang, Z.-Z. Han, Q.-Y. Xie, and W. Zhang, "Finite-time chaos control via nonsingular terminal sliding mode control," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 14, no. 6, pp. 2728–2733, 2009.

[18] V. A. Tuan and H.-J. Kang, "A new finite time control solution for robotic manipulators based on nonsingular fast terminal sliding variables and the adaptive super-twisting scheme," *J. Comput. Nonlinear Dyn.*, vol. 14, no. 3, 2018, Art. no. 031002.

[19] M. Van, S. S. Ge, and H. Ren, "Finite time fault tolerant control for robot manipulators using time delay estimation and continuous nonsingular fast terminal sliding mode control," *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1681–1693, Jul. 2017.

[20] M. Van, M. Mavrovouniotis, and S. S. Ge, "An adaptive backstepping nonsingular fast terminal sliding mode control for robust fault tolerant control of robot manipulators," *IEEE Trans. Syst., Man, Cybern. Syst.*, to be published.

[21] L. Yang and J. Yang, "Nonsingular fast terminal sliding-mode control for nonlinear dynamical systems," *Int. J. Robust Nonlinear Control*, vol. 21, no. 16, pp. 1865–1879, Nov. 2011.

[22] A. T. Vo and H.-J. Kang, "A chattering-free, adaptive, robust tracking control scheme for nonlinear systems with uncertain dynamics," *IEEE Access*, vol. 7, pp. 10457–10466, 2019.

[23] T. Chen and H. Chen, "Approximation capability to functions of several variables, nonlinear functionals, and operators by radial basis function neural networks," *IEEE Trans. Neural Netw.*, vol. 6, no. 4, pp. 904–910, Jul. 1995.

[24] A. T. Vo and H.-J. Kang, "An adaptive terminal sliding mode control for robot manipulators with non-singular terminal sliding surface variables," *IEEE Access*, vol. 7, pp. 8701–8712, 2018.

[25] Y. Feng, M. Zhou, X. Zheng, F. Han, and X. Yu, "Full-order terminal sliding-mode control of MIMO systems with unmatched uncertainties," *J. Franklin Inst.*, vol. 355, no. 2, pp. 653–674, Jan. 2018.

[26] Y. Feng, F. Han, and X. Yu, "Chattering free full-order sliding-mode control," *Automatica*, vol. 50, no. 4, pp. 1310–1314, 2014.

[27] X. Xiang, C. Liu, H. Su, and Q. Zhang, "On decentralized adaptive full-order sliding mode control of multiple UAVs," *ISA Trans.*, vol. 71, pp. 196–205, Nov. 2017.

[28] Z. Song, C. Duan, J. Wang, and Q. Wu, "Chattering-free full-order recursive sliding mode control for finite-time attitude synchronization of rigid spacecraft," *J. Franklin Inst.*, vol. 356, no. 2, pp. 998–1020, Mar. 2018.

[29] V. Utkin, "Discussion aspects of high-order sliding mode control," *IEEE Trans. Autom. Control*, vol. 61, no. 3, pp. 829–833, Mar. 2016.

[30] A. Chalanga, S. Kamal, L. M. Fridman, B. Bandyopadhyay, and J. A. Moreno, "Implementation of super-twisting control: Super-twisting and higher order sliding-mode observer-based approaches," *IEEE Trans. Ind. Electron.*, vol. 63, no. 6, pp. 3677–3685, Jun. 2016.

[31] A. T. Vo, H.-J. Kang, and V.-C. Nguyen, "An output feedback tracking control based on neural sliding mode and high order sliding mode observer," in *Proc. 10th Int. Conf. Hum. Syst. Interact. (HSI)*, Jul. 2017, pp. 161–165.

[32] R. Sun, J. Wang, D. Zhang, and X. Shao, "Neural network-based sliding mode control for atmospheric-actuated spacecraft formation using switching strategy," *Adv. Space Res.*, vol. 61, no. 3, pp. 914–926, Feb. 2018.

[33] J. Liu and X. Wang, *Advanced Sliding Mode Control for Mechanical Systems*. Berlin, Germany: Springer, 2012. doi: 10.1007/978-3-642-20907-9.

[34] Q. Shen, B. Jiang, and V. Cocquempot, "Adaptive fuzzy observer-based active fault-tolerant dynamic surface control for a class of nonlinear systems with actuator faults," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 2, pp. 338–349, Apr. 2014.

[35] A. Polyakov and L. Fridman, "Stability notions and Lyapunov functions for sliding mode control systems," *J. Franklin Inst.*, vol. 351, no. 4, pp. 1831–1865, 2014.

[36] T. D. Le, H.-J. Kang, Y.-S. Suh, and Y.-S. Ro, "An online self-gain tuning method using neural networks for nonlinear PD computed torque controller of a 2-DOF parallel manipulator," *Neurocomputing*, vol. 116, pp. 53–61, Sep. 2013.

[37] T. D. Le, H.-J. Kang, and Y.-S. Suh, "Chattering-free neuro-sliding mode control of 2-DOF planar parallel manipulators," *Int. J. Adv. Robot. Syst.*, vol. 10, no. 1, p. 22, 2013.

ANH TUAN VO received the B.S. degree in electrical engineering from the Da Nang University of Technology, Da Nang, Vietnam, in 2008. He is currently pursuing the Ph.D. degree with the School of Electrical Engineering, University of Ulsan, Ulsan, South Korea. His research interests include intelligent control, sliding mode control, robot fault diagnosis, and fault tolerant control.

HEE-JUN KANG received the B.S. degree in mechanical engineering from Seoul National University, South Korea, in 1985, and the M.S. and Ph.D. degrees in mechanical engineering from The University of Texas at Austin, USA, in 1988 and 1991, respectively. Since 1992, he has been a Professor of electrical engineering with the University of Ulsan. His current research interests include sensor-based robotic application, robot calibration, haptics, robot fault diagnosis, and mechanism analysis.