Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2012, Article ID 342739, 23 pages
doi:10.1155/2012/342739

Research Article

Robust Adaptive Neurocontrol of SISO Nonlinear Systems Preceded by Unknown Deadzone

J. Humberto Pérez-Cruz,[1] E. Ruiz-Velázquez,[1] José de Jesús Rubio,[2] and Carlos A. de Alba-Padilla[1]

[1] *Centro Universitario de Ciencias Exactas e Ingenierías, Universidad de Guadalajara, Boulevord Marcelino García Barragán 1421, 44430 Guadalajara, JAL, Mexico*
[2] *Sección de Estudios de Posgrado e Investigación, ESIME UA-IPN, Avenida de las Granjas 682, Col. Santa Catarina, 02250 México DF, Mexico*

Correspondence should be addressed to J. Humberto Pérez-Cruz, phhhantom2001@yahoo.com.mx

Received 21 June 2012; Revised 6 August 2012; Accepted 16 August 2012

Academic Editor: Jung-Fa Tsai

In this study, the problem of controlling an unknown SISO nonlinear system in Brunovsky canonical form with unknown deadzone input in such a way that the system output follows a specified bounded reference trajectory is considered. Based on universal approximation property of the neural networks, two schemes are proposed to handle this problem. The first scheme utilizes a smooth adaptive inverse of the deadzone. By means of Lyapunov analyses, the exponential convergence of the tracking error to a bounded zone is proven. The second scheme considers the deadzone as a combination of a linear term and a disturbance-like term. Thus, the estimation of the deadzone inverse is not required. By using a Lyapunov-like analyses, the asymptotic converge of the tracking error to a bounded zone is demonstrated. Since this control strategy requires the knowledge of a bound for an uncertainty/disturbance term, a procedure to find such bound is provided. In both schemes, the boundedness of all closed-loop signals is guaranteed. A numerical experiment shows that a satisfactory performance can be obtained by using any of the two proposed controllers.

1. Introduction

The deadzone is a nonsmooth nonlinearity commonly found in many practical systems such as hydraulic positioning systems [1], pneumatic servo systems [2], DC servo motors, among others. When the deadzone is not considered explicitly during the design process, the performance of the control system could be degraded by an increase of the steady-state error, the presence of limit cycles, or inclusive instability [3–6]. A direct way of compensating the deleterious effect of the deadzone is by calculating its inverse. However, this is not an easy question because in many practical situations, both the parameters and the output of the deadzone are unknown. To overcome this problem, in a pioneer work [3], Tao and Kokotović proposed to employ an adaptive inverse of the deadzone. This scheme was applied to linear systems in transfer function form. Cho and Bai [7] extended this work and achieved a perfect asymptotic adaptive cancellation of the deadzone. However, their work assumed that the deadzone output was measurable. In [8], the work of Tao and Kokotović was extended to linear systems in a state space form with nonmeasurable deadzone output. In [9], a new smooth parameterization of the deadzone was proposed and a class of SISO systems with completely known nonlinear functions and with linearly parameterized unknown constants was controlled by using backstepping technique. In order to avoid the construction of the adaptive inverse, in [10], the same class of nonlinear systems as in [9] was controlled by means of a robust adaptive approach and by modeling the deadzone as a combination of a linear term and a disturbance-like term. The controller design in [10] is based on the assumption that maximum and minimum values for the deadzone parameters are a priori known. However, a specific procedure to find such bounds is not provided. In order to extend the class of systems previously considered in [9, 10], in this paper, we propose the development of two controllers based on universal approximation property of the neural networks. The first scheme utilizes a smooth adaptive inverse of the deadzone as in [9]. By means of Lyapunov analyses, the exponential convergence of the tracking error to a bounded zone is proven. The second scheme considers the deadzone as a combination of a linear term and a disturbance-like term as in [10]. Thus, the estimation of the deadzone inverse is not required. By using the Lyapunov-like analyses, the asymptotic converge of the tracking error to a bounded zone is demonstrated. Since this control strategy requires the knowledge of a bound for an uncertainty/disturbance term, a procedure to find such bound is provided. In both schemes, the boundedness of all closed-loop signals is guaranteed. A numerical experiment with a second-order nonlinear system shows that a satisfactory performance can be obtained by using any of the two proposed controllers.

2. Preliminaries and Problem Statement

In this study, the system which will be controlled is composed of an unknown nonlinear plant preceded by an actuator with an unknown deadzone in such a way that the deadzone output is the input to the plant. Consider that the n-order dynamics of the nonlinear plant can be represented as follows:

$$y^{(n)}(t) = f\left(y(t), \dot{y}(t), \ldots, y^{(n-1)}(t)\right) + bu(t) + \xi(t), \tag{2.1}$$

where the scalar $y(t)$ is the output of interest, $y^{(i)}(t)$ for $i = 1, \ldots, n-1$ represents the ith derivative of $y(t)$—each one of these derivatives are assumed measurable, $u(t) \in \Re$ is the deadzone output (and the input to the plant), $f : \Re^n \rightarrow \Re$ is an unknown but continuous nonlinear function, b is an unknown constant, and $\xi(t) \in \Re$ is an unknown but bounded disturbance. Defining the state variables as $x_1(t) := y(t),\ x_2(t) := \dot{y}(t), \ldots, x_n(t) := y^{(n-1)}(t)$, (2.1) can be expressed as follows:

$$\text{PLANT:} \begin{cases} \dot{x}_i(t) = x_{i+1}(t), & i = 1, \ldots, n-1, \\ \dot{x}_n(t) = f(x(t)) + bu(t) + \xi(t), & \\ y(t) = x_1(t), & \end{cases} \tag{2.2}$$

where $x(t) \in \Re^n$ is the measurable state vector for $t \in \Re^+ := \{t : t \geq 0\}$, which is defined as $x(t) := [x_1(t), x_2(t), \ldots, x_n(t)]^T = [x_1(t), \dot{x}_1(t), \ldots, x_1^{(n-1)}(t)]^T$. The nonsymmetric deadzone can be represented by

$$\text{DEADZONE: } u(t) = \text{DZ}(v(t)) = \begin{cases} m_r(v(t) - b_r) & v(t) \geq b_r, \\ 0 & b_l < v(t) < b_r, \\ m_l(v(t) - b_l) & v(t) \leq b_l, \end{cases} \tag{2.3}$$

where m_r and m_l are the right and left constant slopes for the deadzone characteristic, b_r and b_l represent the right and left constant breakpoints, $u(t) \in \Re$ and $v(t) \in \Re$ are the output and the input of the deadzone, respectively. Note that $v(t)$ is the actual control input to the global system formed by the actuator and the plant. In accordance with [3, 4], the deadzone model (2.3) is a static simplification of diverse physical phenomena with negligible fast dynamics. Hereafter, it is considered that the following assumptions are valid.

Assumption 2.1. Without loss of generality, the unknown constant b is assumed positive.

Assumption 2.2. The deadzone output $u(t)$ is not available for measurement.

Assumption 2.3. Although the deadzone parameters b_r, b_l, m_r, and m_l are unknown constants, we can assure that $b_r > 0$, $b_l < 0$, $m_r > 0$, and $m_l > 0$.

The objective that we are trying to achieve is to determine a control signal $v(t)$ such that the output of the plant (2.2), $y(t) = x_1(t)$, follows a specified reference trajectory $y_r(t)$, and, at the same time, all closed-loop signals stay bounded.

2.1. Smooth Parameterization of the Deadzone

A direct way of compensating the deleterious effect of the deadzone is by calculating its inverse. From (2.3), the deadzone inverse can be obtained as

$$v(t) = \text{DZ}^{-1}(u(t)) = \begin{cases} \dfrac{u(t) + m_r b_r}{m_r} & u(t) \geq 0, \\ (b_l, b_r) & u(t) = 0, \\ \dfrac{u(t) + m_l b_l}{m_l} & u(t) \leq 0. \end{cases} \tag{2.4}$$

Notwithstanding, clearly this is a discontinuous function. A smooth approximation of (2.4) was presented in [9] as

$$v(t) \approx \frac{u(t) + m_r b_r}{m_r} \phi_r(u(t)) + \frac{u(t) + m_l b_l}{m_l} \phi_l(u(t)), \tag{2.5}$$

where

$$\phi_r(u(t)) := \frac{\exp(u(t)/\varepsilon_o)}{\exp(u(t)/\varepsilon_o) + \exp(-u(t)/\varepsilon_o)},$$
$$\phi_l(u(t)) := \frac{\exp(-u(t)/\varepsilon_o)}{\exp(u(t)/\varepsilon_o) + \exp(-u(t)/\varepsilon_o)} \tag{2.6}$$

and ε_o is a positive constant chosen by the designer. Since both the parameters and the output of the deadzone are unknown, approximation (2.5) cannot be utilized directly. To overcome this problem, a smooth parameterization of the deadzone was proposed in [9]. For completeness, this parameterization is explained here. Note that (2.3) can be expressed alternatively as

$$u(t) = -\theta^T \omega(t), \tag{2.7}$$

where $\theta = [m_r, m_r b_r, m_l, m_l b_l]^T$, $\omega(t) = [-\sigma_r(t)v(t), \sigma_r(t), -\sigma_l(t)v(t), \sigma_l(t)]^T$:

$$\sigma_r(t) = \begin{cases} 1 & \text{if } v(t) \geq b_r, \\ 0 & \text{otherwise}, \end{cases} \qquad \sigma_l(t) = \begin{cases} 1 & \text{if } v(t) \leq b_l, \\ 0 & \text{otherwise}. \end{cases} \tag{2.8}$$

Given that $u(t)$, θ, and $\omega(t)$ are unknown, the deadzone output $u(t)$ is approximated by

$$u_d(t) := -\widehat{\theta}^T(t)\widehat{\omega}(t), \tag{2.9}$$

where $\widehat{\theta}(t) := [\widehat{m_r}, \widehat{m_r b_r}, \widehat{m_l}, \widehat{m_l b_l}]^T$ is an estimation of θ and $\widehat{\omega}(t) := [-\phi_r(v(t))v(t), \phi_r(v(t)), -\phi_l(v(t))v(t), \phi_l(v(t))]^T$. From (2.7) and (2.9), $u(t)$ can be expressed as

$$u(t) = u_d(t) + \left(\widehat{\theta}(t) - \theta\right)^T \widehat{\omega}(t) + d_N(t), \tag{2.10}$$

where $d_N(t) := \theta^T(\widehat{\omega}(t) - \omega(t))$. Although $d_N(t)$ is an unknown term, its boundedness can be guaranteed [11]. Consider that the positive constant $\overline{d}_N$ is an upper bound for $d_N(t)$, that is, $|d_N(t)| \leq \overline{d}_N$. From (2.9), $v(t)$ can be expressed in terms of $u_d(t)$ as

$$v(t) = \frac{u_d(t) + \widehat{m_r b_r}}{\widehat{m_r}} \phi_r(u_d(t)) + \frac{u_d(t) + \widehat{m_l b_l}}{\widehat{m_l}} \phi_l(u_d(t)). \tag{2.11}$$

This expression can be utilized only if vector $\widehat{\theta}(t)$ is properly determined. To avoid the singularity problem, note that $\widehat{m_r}$ and $\widehat{m_l}$ must always be different from zero.

2.2. Deadzone Representation as a Linear Term and a Disturbance-Like Term

For the particular case when $m := m_r = m_l$, the deadzone model (2.3) can alternatively be described as [10, 12]

$$u(t) = mv(t) + d(t), \tag{2.12}$$

where $d(t)$ is given by

$$d(t) = \begin{cases} -mb_r & v(t) \geq b_r, \\ -mv(t) & b_l < v(t) < b_r, \\ -mb_l & v(t) \leq b_l. \end{cases} \tag{2.13}$$

Note that (2.13) is the negative of a saturation function. Thus, although $d(t)$ could not be exactly known, its boundedness can be assured. Consider that the positive constant $\overline{d}$ is an upper bound for $d(t)$, that is, $|d(t)| \leq \overline{d}$.

3. Neurocontroller Design and Stability Analyses

Based on the universal approximation property of the neural networks, two control schemes are presented in this section to handle the trajectory tracking problem.

Assumption 3.1. The reference trajectory $y_r(t)$ and their first n-derivatives are continuous and bounded. Besides, all these variables are available for the design.

Given the reference trajectory $y_r(t)$ and their first $(n-1)$-derivatives, the vector $x_r(t)$ can be defined as $x_r(t) := [x_{r,1}(t), x_{r,2}(t), \ldots, x_{r,n}(t)]^T = [y_r(t), \dot{y}_r(t), \ldots, y_r^{(n-1)}(t)]^T$. Let us define the filtered tracking error $r(t)$ as

$$r(t) := \left(\frac{d}{dt} + \lambda_r\right)^{n-1} e_1(t), \tag{3.1}$$

where $e_1(t)$ is the first element of the tracking error vector $e(t)$ which is defined simply as $e(t) := x(t) - x_r(t) = [e_1(t), e_2(t), \ldots, e_n(t)]^T = [e_1(t), \dot{e}_1(t), \ldots, e_1^{(n-1)}(t)]^T$ and λ_r is a positive constant chosen by the designer. Note that $r(t)$ can also be expressed as

$$r(t) = \Lambda_r^T e(t), \tag{3.2}$$

where $\Lambda_r \in \Re^n$ is a constant vector given by $\Lambda_r := [\lambda_r^{n-1}, (n-1)\lambda_r^{n-2}, \ldots, 1]^T$.

Remark 3.2. Note that, from (3.1), $r(t)$ can be considered as the input to a stable linear system whose output is $e_1(t)$. Consequently, if $r(t) \in L_\infty$, then $e_1(t), e(t) \in L_\infty$. Specifically, $e(t)$ has the following properties proven in [13]: (i) $e(t)$ converges exponentially to zero, if $r(t) = 0$, (ii) if $e(0) = 0$ and $|r(t)| \leq \varsigma$ where ς is a positive constant, then $e(t)$ belongs to a compact set R given by $R = \{e(t) \in \Re^n$ such that $|e_i(t)| \leq 2^{i-1}\lambda_r^{i-n}\varsigma$, $i = 1, \ldots, n\}$ for $\forall t \geq 0$, and (iii) if $e(0) \neq 0$ and $|r(t)| \leq \varsigma$, then $e(t)$ will converge to R within a time-constant $(n-1)/\lambda_r$.

The first derivative of $r(t)$ can be calculated as

$$\dot{r}(t) = \left(\frac{d}{dt} + \lambda_r\right)^n e_1(t) = \dot{x}_n(t) - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t), \tag{3.3}$$

where $\overline{\Lambda}_r := [0, \lambda_r^{n-1}, (n-1)\lambda_r^{n-2}, \ldots, (n-1)\lambda_r]^T$.

Now then, it is well known that any unknown continuous function can be approximated on a compact set Ω by a neural network as follows [14–16]:

$$f(x) = W^*\sigma(x) + \eta(x), \quad \forall x \in \Omega \subset \Re^n, \tag{3.4}$$

where $\sigma(\cdot)$ is the activation vector function with sigmoidal components, that is, $\sigma(\cdot) := [\sigma_1(\cdot), \ldots, \sigma_s(\cdot)]^T$:

$$\sigma_j(x(t)) := \frac{a_{\sigma j}}{1 + \exp\left(-\sum_{i=1}^{n} c_{\sigma j,i} x_i(t)\right)} - d_{\sigma j} \quad \text{for } j = 1, \ldots, s, \tag{3.5}$$

where $a_{\sigma j}$, $c_{\sigma j,i}$, and $d_{\sigma j}$ are positive constants which can be specified by the designer, $\eta : \Re^n \to \Re$ is the approximation error which is bounded by $|\eta(x)| < \overline{\eta}$ for all $x \in \Omega$, $\overline{\eta}$ is a positive constant, and $W^* \in \Re^{1\times s}$ is an unknown constant vector. Typically, W^* is considered as the optimal weight such that

$$W^* = \arg\min_{W\in\Re^{1\times s}} \left\{ \sup_{x\in\Omega} \left| f(x) - W\sigma(x) \right| \right\}. \tag{3.6}$$

By substituting (3.4) into (2.2), the following alternative representation for the plant dynamics can be obtained:

$$\dot{x}_n(t) = W^*\sigma(x) + bu(t) + \eta(x) + \xi(t). \tag{3.7}$$

By substituting (3.7) into (3.3), we get

$$\dot{r}(t) = W^*\sigma(x) + bu(t) - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t) + \eta(x) + \xi(t). \tag{3.8}$$

3.1. Scheme I

A control scheme which does not require the specific knowledge of the upper bound for the term $\eta(x) + \xi(t) + bd_N(t)$ is developed below.

In order to take into account the effect of the deadzone, the adaptive parameterization model (2.10) is substituted into (3.8):

$$\dot{r}(t) = W^{*}\sigma(x) + bu_d(t) + b\left(\hat{\theta}(t) - \theta\right)^T \hat{\omega}(t) - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t) + \eta(x) + \xi(t) + bd_N(t). \quad (3.9)$$

Consider that $u_d(t)$ is chosen as

$$u_d(t) = \hat{p}(t)\overline{u}_d(t), \quad (3.10)$$

where $\hat{p}(t)$ is an online estimation of $p := 1/b$ and $\overline{u}_d(t)$ is selected as

$$\overline{u}_d(t) = -W(t)\sigma(x(t)) - kr(t) + \dot{x}_{r,n}(t) - \overline{\Lambda}_r^T e(t), \quad (3.11)$$

where $W(t)$ is an online estimation of W^* and k is a positive constant. Note that

$$\begin{aligned} u_d(t) &= \hat{p}(t)\overline{u}_d(t) = (\hat{p}(t) + p - p)\overline{u}_d(t) \\ &= (p - \{p - \hat{p}(t)\})\overline{u}_d(t) = p\overline{u}_d(t) - \tilde{p}(t)\overline{u}_d(t), \end{aligned} \quad (3.12)$$

where $\tilde{p}(t) := p - \hat{p}(t)$. If (3.12) is substituted into (3.9), the following is obtained:

$$\begin{aligned} \dot{r}(t) &= W^{*}\sigma(x) + b(p\overline{u}_d(t) - \tilde{p}(t)\overline{u}_d(t)) + b\left(\hat{\theta}(t) - \theta\right)^T \hat{\omega}(t) \\ &\quad - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t) + \eta(x) + \xi(t) + bd_N(t) \\ &= W^{*}\sigma(x) + \overline{u}_d(t) - b\tilde{p}(t)\overline{u}_d(t) + b\left(\hat{\theta}(t) - \theta\right)^T \hat{\omega}(t) \\ &\quad - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t) + \eta(x) + \xi(t) + bd_N(t). \end{aligned} \quad (3.13)$$

Now, by substituting (3.11) into (3.13) and reducing like terms, the filtered tracking error dynamics can be expressed as

$$\begin{aligned} \dot{r}(t) &= W^{*}\sigma(x(t)) - W(t)\sigma(x(t)) - kr(t) + \dot{x}_{r,n}(t) - \overline{\Lambda}_r^T e(t) - b\tilde{p}(t)\overline{u}_d(t) \\ &\quad + b\left(\hat{\theta}(t) - \theta\right)^T \hat{\omega}(t) - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t) + \eta(x) + \xi(t) + bd_N(t) \\ &= \widetilde{W}(t)\sigma(x(t)) - kr(t) - b\tilde{p}(t)\overline{u}_d(t) + b\left(\hat{\theta}(t) - \theta\right)^T \hat{\omega}(t) + \delta(t), \end{aligned} \quad (3.14)$$

where $\widetilde{W}(t) := W^* - W(t)$ and $\delta(t) := \eta(x) + \xi(t) + bd_N(t)$ is an unknown term but bounded by the positive constant $\overline{\delta}$, that is, $|\delta(t)| \leq \overline{\delta}$.

Once the filtered tracking error dynamics has been determined, the following Lyapunov function candidate is suggested:

$$V(t) = \frac{1}{2}r^2(t) + \frac{1}{2k_1}\widetilde{W}(t)\widetilde{W}^T(t) + \frac{1}{2k_2}\widetilde{p}^2(t) + \frac{1}{2k_3}\widetilde{\theta}^T(t)\widetilde{\theta}(t), \tag{3.15}$$

where $\widetilde{\theta}(t) := \theta - \widehat{\theta}(t)$, and k_1, k_2, and k_3 are positive constants. The first derivative of $V(t)$ is

$$\dot{V}(t) = r(t)\dot{r}(t) + \frac{d}{dt}\left(\frac{1}{2k_1}\widetilde{W}(t)\widetilde{W}^T(t)\right) + \frac{d}{dt}\left(\frac{1}{2k_2}\widetilde{p}^2(t)\right) + \frac{d}{dt}\left(\frac{1}{2k_3}\widetilde{\theta}^T(t)\widetilde{\theta}(t)\right). \tag{3.16}$$

The first derivative of $(1/2k_1)\widetilde{W}(t)\widetilde{W}^T(t)$ can be calculated as follows.

$$\frac{d}{dt}\left(\frac{1}{2k_1}\widetilde{W}(t)\widetilde{W}^T(t)\right) = \frac{1}{2k_1}\dot{\widetilde{W}}(t)\widetilde{W}^T(t) + \frac{1}{2k_1}\widetilde{W}(t)\dot{\widetilde{W}}^T(t). \tag{3.17}$$

Since $\dot{\widetilde{W}}(t)\widetilde{W}^T(t)$ is a scalar and the transpose of a scalar is the same scalar, then

$$\frac{d}{dt}\left(\frac{1}{2k_1}\widetilde{W}(t)\widetilde{W}^T(t)\right) = \frac{1}{2k_1}\widetilde{W}(t)\dot{\widetilde{W}}^T(t) + \frac{1}{2k_1}\widetilde{W}(t)\dot{\widetilde{W}}^T(t) = \frac{1}{k_1}\widetilde{W}(t)\dot{\widetilde{W}}^T(t). \tag{3.18}$$

Proceeding in a similar way for $(d/dt)((1/2k_3)\widetilde{\theta}^T(t)\widetilde{\theta}(t))$, it can be determined that

$$\frac{d}{dt}\left(\frac{1}{2k_3}\widetilde{\theta}^T(t)\widetilde{\theta}(t)\right) = \frac{1}{k_3}\widetilde{\theta}^T(t)\dot{\widetilde{\theta}}(t). \tag{3.19}$$

Substituting (3.14), (3.18), and (3.19) into (3.16) yields

$$\begin{aligned}\dot{V}(t) = {} & r(t)\widetilde{W}(t)\sigma(x(t)) - kr^2(t) - b\widetilde{p}(t)\overline{u}_d(t)r(t) + b\left(\widehat{\theta}(t) - \theta\right)^T\widehat{\omega}(t)r(t) \\ & + \delta(t)r(t) + \frac{1}{k_1}\widetilde{W}(t)\dot{\widetilde{W}}^T(t) + \frac{b}{k_2}\widetilde{p}(t)\dot{\widetilde{p}}(t) + \frac{b}{k_3}\widetilde{\theta}^T(t)\dot{\widetilde{\theta}}(t).\end{aligned} \tag{3.20}$$

Consider that the learning laws for $W(t)$, $\widehat{p}(t)$, and $\widehat{\theta}(t)$ are chosen as

$$\dot{W}(t) = k_1 r(t)\sigma^T(x(t)) - \ell_1\left(W(t) - W^0\right), \tag{3.21}$$

$$\dot{\widehat{p}}(t) = -k_2\overline{u}_d(t)r(t) - \ell_2\left(\widehat{p}(t) - p^0\right), \tag{3.22}$$

$$\dot{\widehat{\theta}}(t) = \operatorname{Proj}\left\{-k_3 r(t)\widehat{\omega}(t) - \ell_3\left(\widehat{\theta}(t) - \theta^0\right)\right\}, \tag{3.23}$$

where ℓ_1, ℓ_2, and ℓ_3 are positive constants, W^0, p^0, and θ^0 are ideally (but not necessarily) good estimations of W^*, p, and θ, respectively, and Proj represents a smooth projection operation as in [17] or [18] in order to avoid that $\widehat{m_r}$ and $\widehat{m_l}$ can be equal to zero. In accordance with (3.21), (3.22), and (3.23) and taking into account that $\dot{\widetilde{W}}(t) = -\dot{W}(t)$, $\dot{\tilde{p}}(t) = -\dot{\hat{p}}(t)$, and $\dot{\tilde{\theta}}(t) = -\dot{\hat{\theta}}(t)$, (3.20) can be expressed as

$$\begin{aligned}\dot{V}(t) = &-kr^2(t) + \delta(t)r(t) + b\left(\hat{\theta}(t) - \theta\right)^T \hat{\omega}(t)r(t) + \frac{\ell_1}{k_1}\widetilde{W}(t)\left(W(t) - W^0\right)^T \\ &+ \frac{\ell_2 b}{k_2}\tilde{p}(t)\left(\hat{p}(t) - p^0\right) - \frac{b}{k_3}\tilde{\theta}^T(t)\mathrm{Proj}\left\{-k_3 r(t)\hat{\omega}(t) - \ell_3\left(\hat{\theta}(t) - \theta^0\right)\right\}.\end{aligned} \tag{3.24}$$

In [11], it is mentioned that the projection operation has the following property:

$$-\left(\theta - \hat{\theta}(t)\right)^T \mathrm{Proj}\{\psi(t)\} \le -\left(\theta - \hat{\theta}(t)\right)^T \psi(t). \tag{3.25}$$

On the other hand, the truthfulness of the following inequalities was proven in [9]

$$\begin{aligned}\frac{\ell_2 b}{k_2}\tilde{p}(t)\left(\hat{p}(t) - p^0\right) &\le -\frac{\ell_2}{2k_2}b\tilde{p}^2(t) + \frac{\ell_2}{2k_2}b\left(p - p^0\right)^2, \\ \frac{\ell_3}{k_3}\left(\theta - \hat{\theta}(t)\right)^T\left(\hat{\theta}(t) - \theta_0\right) &\le -\frac{\ell_3}{2k_3}\tilde{\theta}^T(t)\tilde{\theta}(t) + \frac{\ell_3}{2k_3}\left(\theta - \theta^0\right)^T\left(\theta - \theta^0\right).\end{aligned} \tag{3.26}$$

Likewise, it can be demonstrated that

$$\begin{aligned}\frac{\ell_1}{k_1}\widetilde{W}(t)\left(W_t - W^0\right)^T &\le -\frac{\ell_1}{2k_1}\widetilde{W}(t)\widetilde{W}^T(t) + \frac{\ell_1}{2k_1}\left(W^* - W^0\right)\left(W^* - W^0\right)^T, \\ \delta(t)r(t) &\le \frac{1}{2}r^2(t) + \frac{1}{2}\delta^2(t) \le \frac{1}{2}r^2(t) + \frac{1}{2}\overline{\delta}^2.\end{aligned} \tag{3.27}$$

If the inequalities (3.25), (3.26), (3.27) are substituted into (3.24), we obtain

$$\begin{aligned}\dot{V}(t) \le &-\frac{(2k-1)}{2}r^2(t) - \frac{\ell_1}{2k_1}\widetilde{W}(t)\widetilde{W}^T(t) - \frac{\ell_2}{2k_2}b\tilde{p}^2(t) - \frac{\ell_3}{2k_3}\tilde{\theta}^T(t)\tilde{\theta}(t) \\ &+ \frac{\ell_1}{2k_1}\left(W^* - W^0\right)\left(W^* - W^0\right)^T + \frac{\ell_2}{2k_2}b\left(p - p^0\right)^2 + \frac{\ell_3}{2k_3}\left(\theta - \theta^0\right)^T\left(\theta - \theta^0\right) + \frac{1}{2}\overline{\delta}^2.\end{aligned} \tag{3.28}$$

If $k > 0.5$ and defining $\alpha := \min\{(2k-1), \ell_1, \ell_2, \ell_3\}$ and

$$\beta := \left(\frac{\ell_1}{2k_1}\right)\left(W^* - W^0\right)\left(W^* - W^0\right)^T + \frac{\ell_2}{2k_2} b\left(p - p^0\right)^2 + \frac{\ell_3}{2k_3}\left(\theta - \theta^0\right)^T\left(\theta - \theta^0\right) + \frac{1}{2}\bar{\delta}^2, \tag{3.29}$$

the following bound as a function of $V(t)$ can finally be determined for $\dot{V}(t)$:

$$\dot{V}(t) \leq -\alpha V(t) + \beta \tag{3.30}$$

(3.30) can be rewritten in the following form:

$$\dot{V}(t) + \alpha V(t) \leq \beta. \tag{3.31}$$

Multiplying both sides of the last inequality by $\exp(\alpha t)$, it is possible to obtain

$$\exp(\alpha t)\dot{V}(t) + \alpha \exp(\alpha t) V(t) \leq \beta \exp(\alpha t). \tag{3.32}$$

The left-hand side of (3.32) can be rewritten as

$$\frac{d}{dt}\left(\exp(\alpha t) V(t)\right) \leq \beta \exp(\alpha t) \tag{3.33}$$

or, equivalently, as

$$d\left(\exp(\alpha t) V(t)\right) \leq \beta \exp(\alpha t) dt. \tag{3.34}$$

Integrating both sides of the last inequality from 0 to t yields

$$\exp(\alpha t) V(t) - V(0) \leq \int_0^t \beta \exp(\alpha \tau) d\tau. \tag{3.35}$$

Adding $V(0)$ to both sides of the last inequality, we obtain

$$\exp(\alpha t) V(t) \leq V(0) + \int_0^t \beta \exp(\alpha \tau) d\tau. \tag{3.36}$$

Multiplying both sides of the inequality (3.36) by $\exp(-\alpha t)$ yields

$$V(t) \leq \exp(-\alpha t)V(0) + \exp(-\alpha t)\int_0^t \beta \exp(\alpha\tau)d\tau, \tag{3.37}$$

and, consequently

$$V(t) \leq V(0)\exp(-\alpha t) + \frac{\beta}{\alpha}(1 - \exp(-\alpha t)). \tag{3.38}$$

As α and β are positive constants, the right-hand side of the last inequality can be bounded by $V(0) + \beta/\alpha$. Thus, $V(t) \in L_\infty$ and since by construction $V(t)$ is a nonnegative function, the boundedness of $r(t)$, $\widetilde{W}(t)$, $\tilde{p}(t)$, and $\tilde{\theta}(t)$ can be guaranteed. Because W^*, p, and θ are bounded, $W(t) = W^* - \widetilde{W}(t)$, $\hat{p}(t) = p - \tilde{p}(t)$, and $\hat{\theta}(t) = \theta - \tilde{\theta}(t)$ must be bounded too. If $r(t) \in L_\infty$, from Remark 3.2, we can assure that $e(t) \in L_\infty$ and converges to R. From (3.11), it can be seen that $\overline{u}_d(t)$ is formed by bounded terms and consequently $\overline{u}_d(t) \in L_\infty$. From (3.10), it can be seen that $u_d(t)$ is the product of two bounded variables. Therefore, $u_d(t) \in L_\infty$. As $u_d(t), \hat{\theta}(t) \in L_\infty$, and $\widehat{m_r}$ and $\widehat{m_l}$ are always different from zero, from (2.11), the boundedness of $v(t)$ can be concluded. Now, note that the following is true: $(1/2)r^2(t) \leq V(t)$. Taking into account this fact and from (3.38), we get

$$|r(t)| \leq \sqrt{2V(0)\exp(-\alpha t) + \frac{2\beta}{\alpha}(1 - \exp(-\alpha t))}. \tag{3.39}$$

By taking the limit as $t \to \infty$ of the last inequality, we can guarantee that $|r(t)|$ converges exponentially fast to a zone bounded by the term $\sqrt{2\beta/\alpha}$. Based on this fact together with Remark 3.2, we can conclude that $e_1(t) = y(t) - y_r(t)$ converges exponentially fast to a region around zero bounded by the term $(1/\lambda_r^{n-1})\sqrt{2\beta/\alpha}$. Thus, the following theorem has been proven.

Theorem 3.3. *If Assumptions 2.1–3.1 are satisfied, $k > 0.5$, and the control law* (2.11), (3.10), (3.11) *with the learning laws* (3.21), (3.22), *and* (3.23) *are applied to the system formed by* (2.2)-(2.3), *then the following hold:*

(a) *the filtered tracking error, tracking error, the weights, the estimations of the control gain reciprocal and the deadzone parameters, and the control signal are bounded:*

$$r(t), e(t), W(t), \hat{p}(t), \hat{\theta}(t), v(t) \in L_\infty, \tag{3.40}$$

(b) *the actual tracking error $y(t) - y_r(t)$ converges exponentially fast to a region around zero bounded by the term:*

$$\frac{1}{\lambda_r^{n-1}}\sqrt{\frac{2\beta}{\alpha}}. \tag{3.41}$$

3.2. Scheme II

A simpler scheme can be obtained by using the model (2.12) and the tuning error $r_\varepsilon(t)$ instead of the model (2.10) and the filtered tracking error $r(t)$, respectively. However, the implementation of this scheme requires necessarily the a priori knowledge of a good bound for the term $\eta(x) + \xi(t) + bd(t)$. The development of this scheme is explained below.

If the deadzone model (2.12) is substituted into (3.8), we have

$$\dot{r}(t) = W^*\sigma(x(t)) + bmv(t) - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t) + \eta(x) + \xi(t) + bd(t). \tag{3.42}$$

Note that, by using the model (2.12), the actual control input $v(t)$ appears now directly into the error dynamics (3.42) only multiplied by a constant gain bm.

Consider that $v(t)$ is chosen as

$$v(t) = \widehat{q}(t)\overline{v}(t), \tag{3.43}$$

where $\widehat{q}(t)$ is an online estimation of $q := 1/bm$ and $\overline{v}(t)$ is selected as

$$\overline{v}(t) = -W(t)\sigma(x(t)) - kr(t) - k^* \operatorname{sat}\left(\frac{r(t)}{\varepsilon}\right) + \dot{x}_{r,n}(t) - \overline{\Lambda}_r^T e(t), \tag{3.44}$$

where $W(t)$ is an online estimation of W^*; k, k^*, and ε are positive constants, and $\operatorname{sat}(\cdot)$ represents a saturation function given by

$$\operatorname{sat}(z) = \begin{cases} 1 & \text{for } z \geq 1, \\ z & \text{for } -1 < z < 1, \\ -1 & \text{for } z \leq -1. \end{cases} \tag{3.45}$$

Note that

$$v(t) = \widehat{q}(t)\overline{v}(t) = \left(\widehat{q}(t) + q - q\right)\overline{v}(t) = q\overline{v}(t) - \widetilde{q}(t)\overline{v}(t), \tag{3.46}$$

where $\widetilde{q}(t) := q - \widehat{q}(t)$.

Substituting (3.46) into (3.42) yields

$$\begin{aligned} \dot{r}(t) &= W^*\sigma(x) + bm\left(q\overline{v}(t) - \widetilde{q}(t)\overline{v}(t)\right) - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t) + \eta(x) + \xi(t) + bd(t) \\ &= W^*\sigma(x) + \overline{v}(t) - bm\widetilde{q}(t)\overline{v}(t) - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^T e(t) + \eta(x) + \xi(t) + bd(t). \end{aligned} \tag{3.47}$$

If (3.44) is substituted into (3.47) and reducing like terms, the filtered tracking error dynamics can be expressed as

$$\begin{aligned}\dot{r}(t) &= W^{*}\sigma(x(t)) - W(t)\sigma(x(t)) - kr(t) - k^{*}\,\mathrm{sat}\left(\frac{r(t)}{\varepsilon}\right)\\ &\quad + \dot{x}_{r,n}(t) - \overline{\Lambda}_r^{T} e(t) - bm\tilde{q}(t)\overline{v}(t) - \dot{x}_{r,n}(t) + \overline{\Lambda}_r^{T} e(t) + \eta(x) + \xi(t) + bd(t)\\ &= \widetilde{W}(t)\sigma(x(t)) - kr(t) - k^{*}\,\mathrm{sat}\left(\frac{r(t)}{\varepsilon}\right) - bm\tilde{q}(t)\overline{v}(t) + \zeta(t),\end{aligned} \tag{3.48}$$

where $\widetilde{W}(t) := W^{*} - W(t)$ and $\zeta(t) := \eta(x) + \xi(t) + bd(t)$ is an unknown term bounded by the positive constant $\overline{\zeta}$, that is, $|\zeta(t)| \leq \overline{\zeta}$. Consider the following Lyapunov function candidate:

$$V(t) = \frac{1}{2}r_{\varepsilon}^{2}(t) + \frac{1}{2k_1}\widetilde{W}(t)\widetilde{W}^{T}(t) + \frac{bm}{2k_2}\tilde{q}^{2}(t), \tag{3.49}$$

where k_1 and k_2 are positive constants and $r_{\varepsilon}(t)$ is the tuning error defined as [10]

$$r_{\varepsilon}(t) = r(t) - \varepsilon\,\mathrm{sat}\left(\frac{r(t)}{\varepsilon}\right). \tag{3.50}$$

Taking into account (3.18), the first derivative of $V(t)$ can be calculated as

$$\dot{V}(t) = r_{\varepsilon}(t)\dot{r}_{\varepsilon}(t) + \frac{1}{k_1}\widetilde{W}(t)\dot{\widetilde{W}}^{T}(t) + \frac{bm}{k_2}\tilde{q}(t)\dot{\tilde{q}}(t). \tag{3.51}$$

It can be demonstrated that $r_{\varepsilon}(t)\dot{r}_{\varepsilon}(t) = r_{\varepsilon}(t)\dot{r}(t)$. By substituting (3.48) into the last equality and the resulting expression into (3.51), we get

$$\begin{aligned}\dot{V}(t) &= \widetilde{W}(t)\sigma(x(t))r_{\varepsilon}(t) - kr(t)r_{\varepsilon}(t) - k^{*}r_{\varepsilon}(t)\,\mathrm{sat}\left(\frac{r(t)}{\varepsilon}\right)\\ &\quad - bm\tilde{q}(t)\overline{v}(t)r_{\varepsilon}(t) + r_{\varepsilon}(t)\zeta(t) + \frac{1}{k_1}\widetilde{W}(t)\dot{\widetilde{W}}^{T}(t) + \frac{bm}{k_2}\tilde{q}(t)\dot{\tilde{q}}(t).\end{aligned} \tag{3.52}$$

Consider that $\dot{\widetilde{W}}(t)$ and $\dot{\tilde{q}}(t)$ are chosen as

$$\dot{\widetilde{W}}(t) = -k_1\sigma^{T}(x(t))r_{\varepsilon}(t), \tag{3.53}$$

$$\dot{\tilde{q}}(t) = k_2\overline{v}(t)r_{\varepsilon}(t). \tag{3.54}$$

If (3.53) and (3.54) are substituted into (3.52), we obtain

$$\dot{V}(t) = -kr(t)r_{\varepsilon}(t) - k^{*}r_{\varepsilon}(t)\,\mathrm{sat}\left(\frac{r(t)}{\varepsilon}\right) + r_{\varepsilon}(t)\zeta(t). \tag{3.55}$$

Now, from (3.50),

$$r(t) = r_\varepsilon(t) + \varepsilon \,\text{sat}\left(\frac{r(t)}{\varepsilon}\right). \tag{3.56}$$

Besides,

$$r_\varepsilon(t)\zeta(t) \le |r_\varepsilon(t)\zeta(t)| = |r_\varepsilon(t)||\zeta(t)| \le |r_\varepsilon(t)|\overline{\zeta}. \tag{3.57}$$

Substituting (3.56) and (3.57) into (3.55) yields

$$\begin{aligned}\dot{V}(t) &= -k\left(r_\varepsilon(t) + \varepsilon \,\text{sat}\left(\frac{r(t)}{\varepsilon}\right)\right) r_\varepsilon(t) - k^* r_\varepsilon(t)\,\text{sat}\left(\frac{r(t)}{\varepsilon}\right) + r_\varepsilon(t)\zeta(t) \\ &\le -k r_\varepsilon^2(t) - k\varepsilon r_\varepsilon(t)\,\text{sat}\left(\frac{r(t)}{\varepsilon}\right) - k^* r_\varepsilon(t)\,\text{sat}\left(\frac{r(t)}{\varepsilon}\right) + |r_\varepsilon(t)|\overline{\zeta}.\end{aligned} \tag{3.58}$$

Considering that $r_\varepsilon(t)\,\text{sat}(r(t)/\varepsilon) = |r_\varepsilon(t)|$, (3.58) can be expressed as

$$\dot{V}(t) \le -k r_\varepsilon^2(t) - k\varepsilon|r_\varepsilon(t)| - k^*|r_\varepsilon(t)| + |r_\varepsilon(t)|\overline{\zeta}. \tag{3.59}$$

Note that, if k^* and k are selected in such a way that $k^* + k\varepsilon > \overline{\zeta}$, then

$$\dot{V}(t) \le -k r_\varepsilon^2(t) - \left(k^* + k\varepsilon - \overline{\zeta}\right)|r_\varepsilon(t)| \le -k r_\varepsilon^2(t). \tag{3.60}$$

Because k is a positive constant, the last inequality implies that $\dot{V}(t) \le 0$. Based on this fact, it is possible to establish that $V(t) \le V(0)$ and, therefore, $V(t) \in L_\infty$. Since $V(t)$ is a nonnegative function, $r_\varepsilon(t)$, $\widetilde{W}(t)$ and $\widetilde{q}(t)$ belongs to L_∞. Because $W(t) = W^* - \widetilde{W}(t)$, and $\widehat{q}(t) = q - \widetilde{q}(t)$, and as W^* and q are constants, the boundedness of $W(t)$ and $\widehat{q}(t)$ can be guaranteed. From the definition of tuning error (3.50) and as $\varepsilon\,\text{sat}(r(t)/\varepsilon)$ is a bounded term, if $r_\varepsilon(t) \in L_\infty$, then $r(t) \in L_\infty$. Keeping in view the above fact, and on the basis of Remark 3.2, the boundedness of $e(t)$ can be assured. Now, it can be seen from (3.44) that $\overline{v}(t)$ is formed by bounded terms. Thus, $\overline{v}(t) \in L_\infty$. Likewise, it can be seen from (3.43) that $v(t)$ is the product of two bounded variables and consequently $v(t)$ is also bounded. On the other hand, an inspection of (3.48) reveals that $\dot{r}(t) \in L_\infty$. From (3.50), this means that $\dot{r}_\varepsilon(t)$ is bounded too. Integrating both sides of (3.60) from 0 to t yields

$$V(t) - V(0) \le -k\int_0^t r_\varepsilon^2(\tau)\,d\tau. \tag{3.61}$$

Note that the last inequality can be expressed as

$$\int_0^t r_\varepsilon^2(\tau)\,d\tau \le \frac{V(0) - V(t)}{k}. \tag{3.62}$$

Since $V(t)$ is a nonnegative function, the following is true:

$$V(t) \geq 0 \Longrightarrow V(t) - V(0) \geq -V(0) \Longrightarrow V(0) - V(t) \leq V(0). \tag{3.63}$$

Substituting (3.63) into (3.62) yields

$$\int_0^t r_\varepsilon^2(\tau)d\tau \leq \frac{V(0)}{k} \tag{3.64}$$

taking the limit as $t \rightarrow \infty$ of both sides of the last inequality, finally, we obtain

$$\int_0^\infty r_\varepsilon^2(\tau)d\tau \leq \frac{V(0)}{k}. \tag{3.65}$$

This means that $r_\varepsilon(t) \in L_2$. As $r_\varepsilon(t) \in L_2 \cap L_\infty$ and $\dot{r}_\varepsilon(t) \in L_\infty$, from Barbalat's Lemma, we can conclude that $r_\varepsilon(t)$ converges asymptotically to zero. From definition (3.50), this implies that $r(t)$ asymptotically converges to a region around zero bounded by ε. In view of the above and from Remark 3.2, we can conclude that $e_1(t) = y(t) - y_r(t)$ converges asymptotically to a region around zero bounded by the term $\varepsilon/\lambda_r^{n-1}$. Thus, the following theorem has been proven.

Theorem 3.4. *If Assumptions 2.1–3.1 are satisfied, $k^* + k\varepsilon > \overline{\zeta}$, and the control input* (3.43), (3.44) *with the learning laws $\dot{W}(t) = k_1\sigma^T(x(t))r_\varepsilon(t)$, $\dot{\hat{q}}(t) = -k_2\overline{v}(t)r_\varepsilon(t)$ are applied to the system* (2.2)-(2.3), *then*

(a) *the tuning error, the filtered tracking error, tracking error, the weights, the* bm *reciprocal estimation, and the control signal are bounded:*

$$r_\varepsilon(t), r(t), e(t), W(t), \hat{q}(t), v(t) \in L_\infty, \tag{3.66}$$

(b) *the actual tracking error $y(t) - y_r(t)$ converges asymptotically to a region around zero bounded by the term*

$$\frac{\varepsilon}{\lambda_r^{n-1}}, \tag{3.67}$$

where ε and λ_r are positive constants chosen by the designer.

3.2.1. Estimation of a Bound for $\zeta(t)$

Certainly, compared with the scheme I, the structure of the scheme II is simpler. Nevertheless, the implementation of this last scheme requires a good estimation of $\overline{\zeta}$ in order to guarantee the theoretical performance provided by Theorem 3.4. Here, we propose an offline practical procedure to achieve this goal.

Consider that some experimental data $(x_{t_i}, v(t_i))_{i=1,\ldots,N}$ are available (see Remark 3.5). By substituting (2.12) into (3.7), we obtain

$$\dot{x}_n(t) = W^*\sigma(x(t)) + bmv(t) + \zeta(t). \tag{3.68}$$

Certainly, if $\dot{x}_n(t)$, W^*, and bm could be known, $\zeta(t)$ can be completely determined. By hypothesis, $x_n(t)$ is known. However, because of the noise, $\dot{x}_n(t)$ must not be calculated directly from $x_n(t)$. Instead, a robust differentiation strategy should be used such as sliding modes [19], high-order sliding modes [20, 21], or smoothing by least squares, among others. Hereafter, the estimation of $\dot{x}_n(t)$ is denoted by $\dot{\overline{x}}_n(t)$. Once $\dot{\overline{x}}_n(t)$ is obtained, let us consider the following regression model:

$$W^*\sigma(x(t_i)) + bmv(t_i) = \dot{\overline{x}}_n(t_i) + \chi(t_i), \tag{3.69}$$

where $\chi(t_i)$ is simply an error term. By defining $X_{t_i} := [\sigma^T(x(t_i)), v(t_i)]$, and $\Phi := [W^*, bm]$, (3.69) can be expressed as

$$\Phi X_{t_i}^T = \dot{\overline{x}}_n(t_i) + \chi(t_i). \tag{3.70}$$

By using least-squares method, Φ can be estimated as

$$\overline{\Phi} = \left[\overline{W^*}, \overline{bm}\right] = \arg\min_{\Phi} \sum_{i=1}^{N} \chi^2(t_i) = \left(\left(X^T X\right)^{-1} X^T Y\right)^T, \tag{3.71}$$

where $X := [X_{t_1}, \ldots, X_{t_N}]^T$ and $Y = [\dot{\overline{x}}_n(t_1), \ldots, \dot{\overline{x}}_n(t_N)]^T$. Once $\overline{\Phi}$ is determined, $\zeta(t)$ can be approximated as

$$\zeta(t) \approx \dot{\overline{x}}_n(t) - \overline{W^*}\sigma(x(t)) - \overline{bm}v(t). \tag{3.72}$$

Next, $\overline{\zeta}$ can be estimated from (3.72).

Remark 3.5. These experimental data can be generated by trying to use the neurocontroller II with relatively large values for the constant parameters. A first attempt could be to let $k = 100$ and $k^* = 50$. If the tracking is not satisfactory, then larger values could be tried.

Remark 3.6. Alternatively, in [22], the determination of an error term $\delta(t)$ is achieved by using of sliding modes.

4. Numerical Example

In this section, the proposed neurocontrollers are tested by simulation on the following second order nonlinear system:

$$\begin{aligned}\dot{x}_1(t) &= x_2(t),\\ \dot{x}_2(t) &= -2.3\left(\frac{1-\exp(-x_1(t))}{1+\exp(-x_2(t))}\right)+3.7\frac{x_2(t)\sin(x_1(t)x_2(t))\cos(x_2(t))}{x_1^2(t)+x_2^2(t)+1}\\ &\quad + 1.5x_1(t)x_2(t)+0.7x_1(t)x_2^3(t)\sin(2x_1(t))+0.4x_1^2(t)x_2(t)+3.5u(t)+\xi(t).\end{aligned} \tag{4.1}$$

The initial condition for system (4.1) is $x_1(0) = 1,\ x_2(0) = -1$; $u(t)$ is the deadzone output whose parameters are $m_r = m_l = 1.5,\ b_r = 2.5,\ b_l = -2$; $\xi(t)$, the disturbance term is selected as $\xi(t) = \sin(10t)$. The following reference trajectory is proposed $y_r(t) = -\cos(t)+1.5\cos(2t)-0.5$. The first and second derivative of $y_r(t)$ can be calculated analytically. In another case, a robust differentiation method must be used. It is very important to mention that the nonlinear system (4.1), the exact values for deadzone parameters, and the disturbance term are assumed completely unknown for the neurocontrollers during the design and simulation process. That is, the system (4.1) is only used as a data generator.

With respect to the tuning process, we must recognize that, similarly to many techniques of adaptive control, we do not have a systematic procedure in order to find the proper values for the controller parameters. Instead, an approach by trial and error is used. Thus, through various iterative simulations, the following values for the parameters of the neurocontroller I were found:

$$\begin{gathered} k = 100, \quad \varepsilon_0 = 0.2, \quad k_1 = 400, \quad l_1 = 2, \quad k_2 = 10, \quad l_2 = 1,\\ k_3 = 100, \quad l_3 = 1, \quad \lambda_r = 1, \quad W^0 = [1,0,1,0.5],\\ W(0) = W^0, \quad \theta^0 = [1,1,1,-1]^T, \quad \theta(0) = \theta^0, \quad p^0 = 1, \quad p(0) = p^0,\\ \sigma_j(x(t)) = \frac{2}{\left(1+\exp\left(-\sum_{i=1}^{2} c_{\sigma j,i}x_i(t)\right)\right)} - 1 \quad \text{for } j = 1,2,3,4,\\ c_{\sigma 1,1} = 1, \quad c_{\sigma 1,2} = 1, \quad c_{\sigma 2,1} = 0.5, \quad c_{\sigma 2,2} = 0.5,\\ c_{\sigma 3,1} = 2, \quad c_{\sigma 3,2} = 2, \quad c_{\sigma 4,1} = 0.4, \quad c_{\sigma 4,2} = 0.8. \end{gathered} \tag{4.2}$$

Notice that a great freedom is allowed in order to select W^0, θ^0, and p^0. However, the designer should be aware of that as these parameters take values increasingly different from the optimal ones, the parameter β in Theorem 3.3 becomes larger. On the other hand, we have seen that by setting $W(0) = W^0$, $\theta(0) = \theta^0$, and $p(0) = p^0$, a more regular behavior of the closed-loop system can be obtained [23, 24].

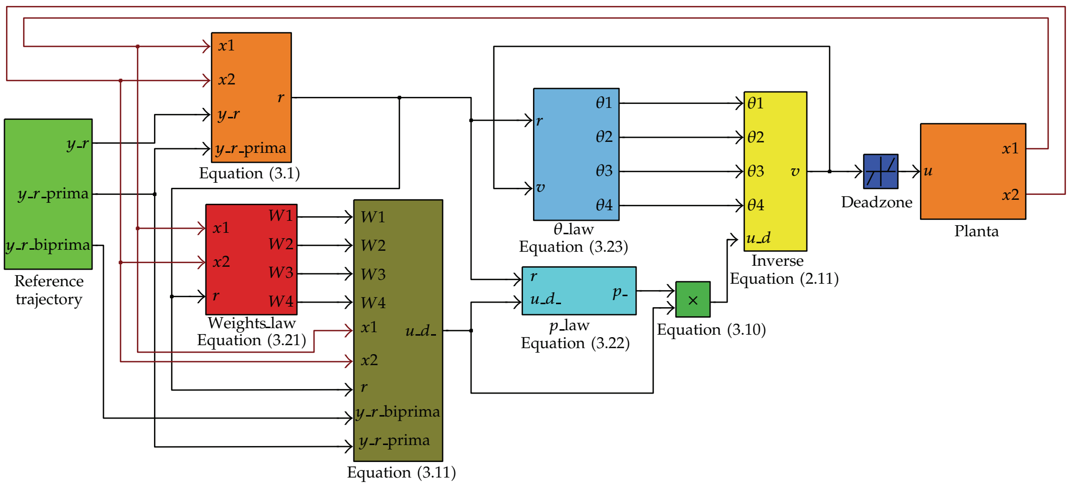

Figure 1: Simulink block diagram for the neurocontroller I.

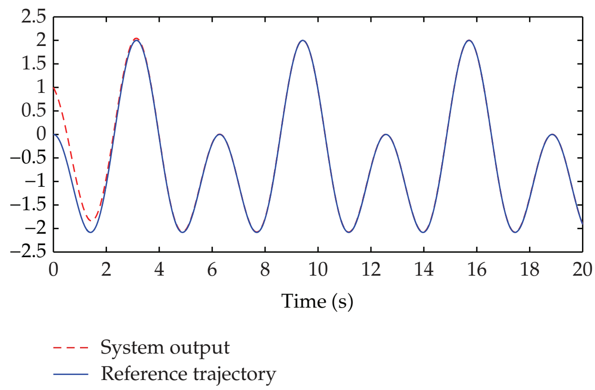

Figure 2: Tracking process for the neurocontroller I.

The simulation is carried out by means of Simulink with ode45 method, relative tolerance equal to $1e-6$, and absolute tolerance equal to $1e-8$. A Simulink block diagram for the neurocontroller based on scheme I is shown in Figure 1. The results of the tracking process are presented in Figures 2–4 for the first 20 seconds of the simulation. In Figure 2, the output of nonlinear system (4.1), $y(t) = x_1(t)$, is represented by dashed line whereas the reference trajectory $y_r(t)$ is represented by solid line. In spite of the difference between $y(0)$ and $y_r(0)$, the tracking process shows a satisfactory behavior. This can be verified more specifically in Figure 3 where the actual tracking error, $y(t)-y_r(t)$, is illustrated. In order to appreciate better the quality of the tracking process, a zoom of Figure 3 is presented in Figure 4. From Figures 3 and 4, we can appreciate that the actual tracking error converges fast to a zone bounded by 0.02. Finally, the control signal $v(t)$ acting as the input of deadzone is shown in Figure 5.

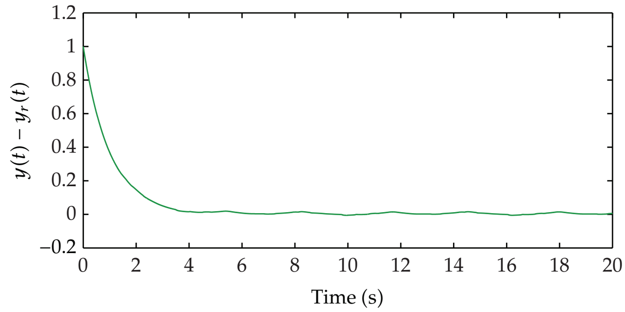

Figure 3: Tracking error evolution for the neurocontroller I.

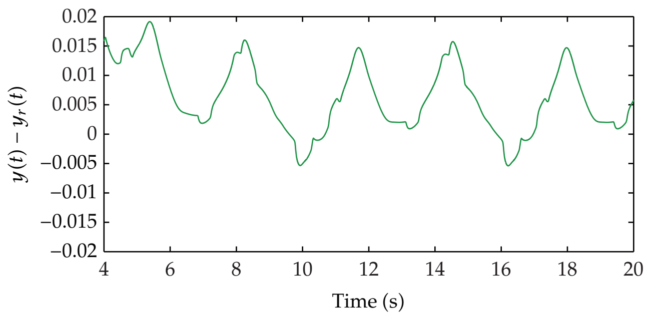

Figure 4: Tracking error evolution for the neurocontroller I (zoom of Figure 3).

For the case of the neurocontroller II, its parameters are selected very similarly to the neurocontroller I parameters in order to provide a proper reference for comparison. The parameters of the neurocontroller II are

$$
\begin{gathered}
k = 100, \quad k^* = 30, \quad \varepsilon = 0.1, \quad k_1 = 400, \quad k_2 = 10, \quad \lambda_r = 1, \\
W^0 = [1, 0, 1, 0.5], \quad W(0) = W^0, \quad q^0 = 1, \quad q(0) = q^0, \\
\sigma_j(x(t)) = \frac{2}{\left(1 + \exp\left(-\sum_{i=1}^{2} c_{\sigma j,i} x_i(t)\right)\right)} - 1 \quad \text{for } j = 1, 2, 3, 4, \\
c_{\sigma 1,1} = 1, \quad c_{\sigma 1,2} = 1, \quad c_{\sigma 2,1} = 0.5, \quad c_{\sigma 2,2} = 0.5, \\
c_{\sigma 3,1} = 2, \quad c_{\sigma 3,2} = 2, \quad c_{\sigma 4,1} = 0.4, \quad c_{\sigma 4,2} = 0.8.
\end{gathered}
\tag{4.3}
$$

A Simulink block diagram for the neurocontroller based on scheme II is shown in Figure 6. Since a very similar performance is obtained, the signals of the actual tracking error for the two neurocontrollers are presented together in Figure 7. In Figure 8, a zoom of Figure 7 is presented. As can be seen from Figures 7 and 8, the actual tracking error for the controller II converges fast to a zone bounded by 0.02. This is a remarkable behavior since Theorem 3.4 only guarantees the asymptotical convergence to a zone bounded by 0.2. Finally, the control signals of the two neurocontrollers are displayed in Figure 9.

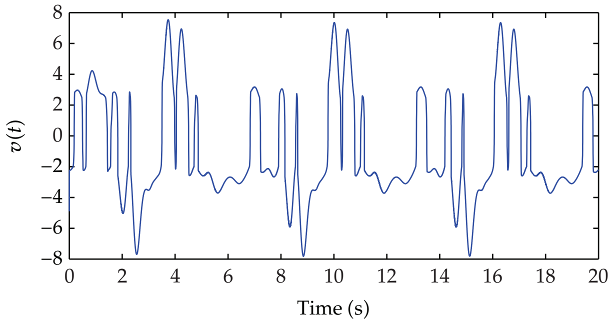

Figure 5: Control signal of the scheme I.

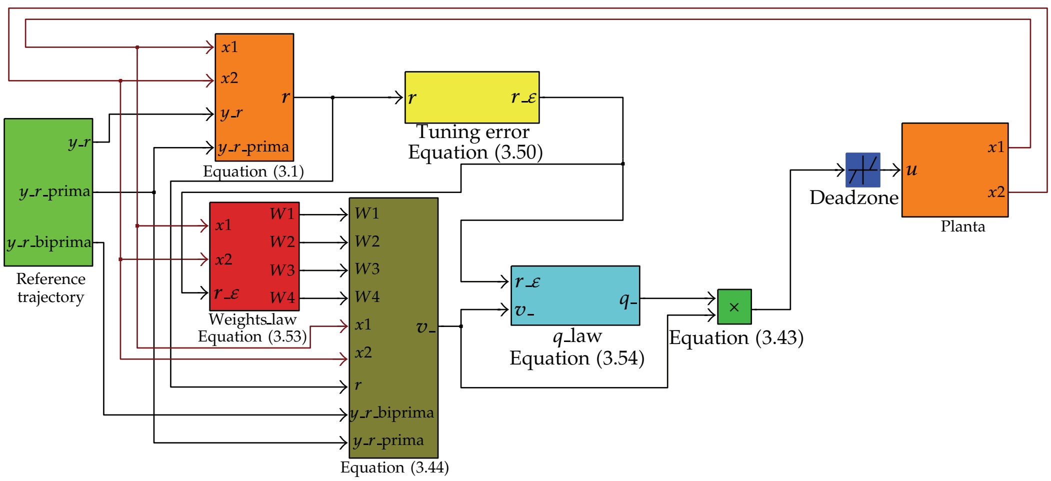

Figure 6: Simulink block diagram for the neurocontroller II.

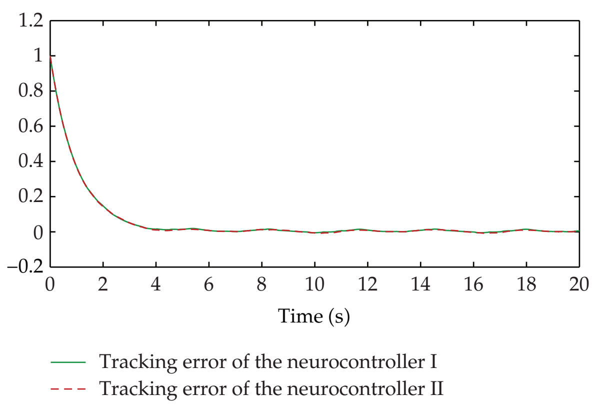

Figure 7: Tracking error evolution for the two neurocontrollers.

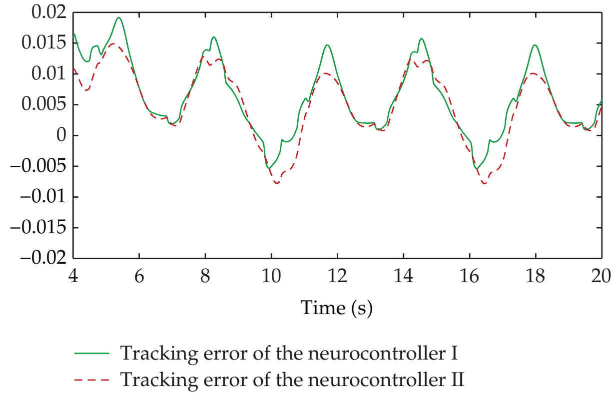

Figure 8: Tracking error evolution for the two neurocontrollers (zoom of Figure 7).

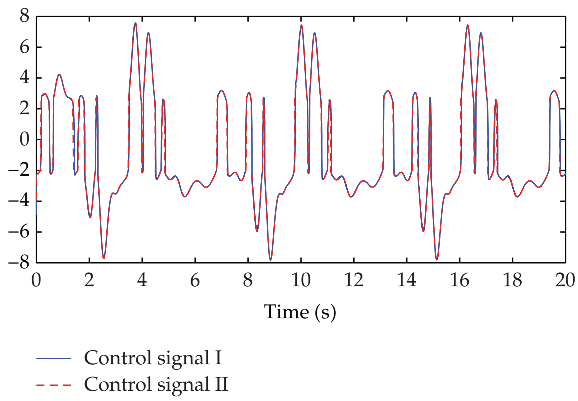

Figure 9: Control signals of the two schemes.

So far, the neurocontroller II has been used in a nonrigorous way. In order to guarantee its stability, a bound for $\zeta(t)$ must be first determined. By using the procedure presented in Section 3, W^*, bm can be estimated as $\overline{W^*} = [10.7869, -29.0274, -7.4315, 19.3033]$, $\overline{bm} = 0.2453$. With these parameters, the maximum value for $\zeta(t)$ is estimated as 7.5819 and a very conservative bound for $\zeta(t)$ can be established as $\overline{\zeta} = 16$. In view of the above and in accordance with the selected values for the neurocontroller II parameters and from Theorem 3.4, the stability of the closed-loop system can be guaranteed.

As the comparison between two empirically tuned techniques is necessarily limited, the main objective of this numerical example was rather to show that by using any of the two proposed controllers, a good performance can be obtained. Notwithstanding, it must be mentioned that the tuning process was easier for the case of Scheme II.

5. Conclusions

In this paper, two adaptive schemes based on universal approximation property of the neural networks are proposed in order to control an unknown SISO nonlinear system in Brunovsky canonical form with unknown deadzone input. The objective is to determine a control signal such that the output of the unknown plant follows a specified reference trajectory, and, at the same time, all closed-loop signals stay bounded in spite of the presence of unknown but bounded disturbances. The first scheme utilizes a smooth adaptive inverse of the deadzone while the second scheme considers the deadzone as a combination of a linear term and a disturbance-like term. For the first case, the exponential convergence of the tracking error is guaranteed by using a Lyapunov analyses. For the second case, only the asymptotical convergence can be guaranteed theoretically. However, a numerical example shows that a similarly satisfactory performance can be obtained by using any of the two proposed controllers. The designer should select between these two schemes in accordance with the particular features of his application and considering the following:

(i) Scheme I does not need the specific knowledge of a bound for the unknown dynamics and/or the disturbance term.

(ii) Scheme I can handle the case when $m_r \neq m_l$.

(iii) The tuning and implementation process is easier for Scheme II.

Finally, it must be mentioned that a compromise should be established between the accuracy of the tracking process and the smoothness of the control signal.

Acknowledgment

J. H. Pérez-Cruz would like to thank the financial support through a postdoctoral fellowship from the Mexican National Council for Science and Technology (CONACYT).

References

[1] B. Magyar, C. Hös, and G. Stépán, “Influence of control valve delay and dead zone on the stability of a simple hydraulic positioning system,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 349489, 15 pages, 2010.

[2] A. C. Valdiero, C. S. Ritter, C. F. Rios, and M. Rafikov, “Nonlinear mathematical modeling in pneumatic servo position applications,” *Mathematical Problems in Engineering*, vol. 2011, Article ID 472903, 16 pages, 2011.

[3] G. Tao and P. V. Kokotović, “Adaptive control of plants with unknown dead-zones,” *IEEE Transactions on Automatic Control*, vol. 39, no. 1, pp. 59–68, 1994.

[4] G. Tao and P. V. Kokotović, *Adaptive Control of Systems with Actuator and Sensor Nonlinearities*, John Wiley & Sons, New York, NY, USA, 1996.

[5] G. Tao and F. L. Lewis, *Adaptive Control of Nonsmooth Dynamic Systems*, Springer, 2003.

[6] Y.-J. Sun, “Composite tracking control for generalized practical synchronization of duffing-holmes systems with parameter mismatching, unknown external excitation, plant uncertainties, and uncertain deadzone nonlinearities,” *Abstract and Applied Analysis*, vol. 2012, Article ID 640568, 11 pages, 2012.

[7] H. Y. Cho and E.-W. Bai, “Convergence results for an adaptive dead zone inverse,” *International Journal of Adaptive Control and Signal Processing*, vol. 12, no. 5, pp. 451–466, 1998.

[8] X. S. Wang, H. Hong, and C. Y. Su, “Model reference adaptive control of continuous-time systems with an unknown input dead-zone,” *IEE Proceedings: Control Theory and Applications*, vol. 150, no. 3, pp. 261–266, 2003.

[9] J. Zhou and X. Z. Shen, “Robust adaptive control of nonlinear uncertain plants with unknown dead-zone,” *IET Control Theory & Applications*, vol. 1, no. 1, pp. 25–32, 2007.

[10] X.-S. Wang, C.-Y. Su, and H. Hong, “Robust adaptive control of a class of nonlinear systems with unknown dead-zone,” *Automatica*, vol. 40, no. 3, pp. 407–413, 2004.

[11] J. Zhou and C. Wen, *Adaptive Backstepping Control of Uncertain Systems*, Springer, Berlin, Germany, 2007.

[12] F. L. Lewis, J. Campos, and R. Selmic, *Neuro-fuzzy control of industrial systems with actuator nonlinearities*, Society for Industrial and Applied Mathematics, Philadelphia, Pa, USA, 2002.
[13] J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, New York, NY, USA, 1991.
[14] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems*, vol. 2, no. 4, pp. 303–314, 1989.
[15] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989.
[16] F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor & Francis, Boca Raton, Fla, USA, 1999.
[17] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design*, John Wiley & Sons, New York, NY, USA, 1995.
[18] P. A. Ioannou and J. Sun, *Robust Adaptive Control*, Prentice Hall, New York, NY, USA, 1996.
[19] I. Chairez, A. Poznyak, and T. Poznyak, "New sliding-mode learning law for dynamic neural network observer," *IEEE Transactions on Circuits and Systems II*, vol. 53, no. 12, pp. 1338–1342, 2006.
[20] A. Levant, "Robust exact differentiation via sliding mode technique," *Automatica*, vol. 34, no. 3, pp. 379–384, 1998.
[21] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control," *International Journal of Control*, vol. 76, no. 9-10, pp. 924–941, 2003.
[22] J. H. Pérez-Cruz, I. Chairez, A. Poznyak, and J. J. de Rubio, "Constrained neural control for the adaptive tracking of power profiles in a triga reactor," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 7, pp. 4575–4788, 2011.
[23] J. H. Pérez-Cruz, A. Y. Alanis, J. D. J. Rubio, and J. Pacheco, "System identification using multilayer differential neural networks: a new result," *Journal of Applied Mathematics*, vol. 2012, Article ID 529176, 20 pages, 2012.
[24] R. Carmona and W. Yu, "Robust adaptive control via neural linearization and compensation," vol. 2012, Article ID 867178, 9 pages, 2012.