Research Article

Adaptive Neural Network Motion Control of Manipulators with Experimental Evaluations

S. Puga-Guzmán,[1,2] **J. Moreno-Valenzuela,**[2] **and V. Santibáñez**[3]

[1] *Instituto Tecnológico de Tijuana, Boulevard. Industrial S/N, 22510 Tijuana, BC, Mexico*
[2] *Instituto Politécnico Nacional-CITEDI, Avenida del Parque 1310, Mesa de Otay, 22510 Tijuana, BC, Mexico*
[3] *Instituto Tecnológico de La Laguna, Boulevard Revolución y Cuahtémoc, 27000 Torreón, COAH, Mexico*

Correspondence should be addressed to J. Moreno-Valenzuela; moreno@citedi.mx

A nonlinear proportional-derivative controller plus adaptive neuronal network compensation is proposed. With the aim of estimating the desired torque, a two-layer neural network is used. Then, adaptation laws for the neural network weights are derived. Asymptotic convergence of the position and velocity tracking errors is proven, while the neural network weights are shown to be uniformly bounded. The proposed scheme has been experimentally validated in real time. These experimental evaluations were carried in two different mechanical systems: a horizontal two degrees-of-freedom robot and a vertical one degree-of-freedom arm which is affected by the gravitational force. In each one of the two experimental set-ups, the proposed scheme was implemented without and with adaptive neural network compensation. Experimental results confirmed the tracking accuracy of the proposed adaptive neural network-based controller.

1. Introduction

Robust control consists in designing control strategies by using little information of the system model and considering that the system may be affected by bounded disturbances. Robust controllers can be designed to satisfy either the regulation goal or the trajectory tracking objective. Thus, in the last years there has been mathematical and practical interest in studying robust control architectures. In this class of controllers, adaptive neural networks have been used in the design of robust controllers for electromechanical systems.

Neural networks can be used in the control of unknown systems without requirements for linearity in the system parameters. Neural networks exhibit the universal approximation property which allows approximating unknown linear and nonlinear functions [1].

In the following, we provide a literature review on application of neural networks for robot motion control. Selmic and Lewis [2] proposed a dynamical inversion compensation scheme by using a backstepping technique with neural networks, which was applied to mechanical systems. Kwan et al. [3] proposed a class of neural network

robust controllers showing global asymptotic stability of tracking errors and boundedness of neural network weights. Lewis et al. [1] provided in manner of survey a study of the application of neural networks in the compensation of actuator nonlinearities. In [4] a robust controller with a nonlinear two-layer neural network structure was proposed for control of four-axis SCARA robot manipulator. The error trajectories are proven to be uniformly ultimately bounded. Yu and Li [5] propose a PD-type controller plus neural network compensation of the uncertainties and velocity estimation is achieved by using a high-gain observer. Stability is proven using Lyapunov-based analysis. Wang et al. [6] proposed a neural network-based motion controller in task space. The controller is addressed as two-loop cascade control scheme. The inner loop implements a velocity servo loop at the robot joint level using a radial basis function network with a proportional-integral controller. In [7] an adaptive neural network algorithm is developed for rigid-link electrically driven robot systems. The controller is developed in a constructive form and a rigorous stability analysis is also provided. In Moreno-Armendáriz et al. [8], an indirect adaptive control using hierarchical fuzzy CMAC neuronal

network for the ball and plate system was introduced. The proposed controller was validated by means of numerical simulations and experiments.

Fateh and Alavi [9] introduced a new scheme for the impedance control of an active suspension system. The control was achieved through two interior loops which are force control of the actuator by feedback linearization and fuzzy control loop. The work proposed by Liu et al. in [10, 11] addressed the problem of control of manipulators with saturated input. In particular, new controllers based on fuzzy self-tuning of the proportional and derivative gains were proposed. Although the controllers in [9–11] are based in fuzzy logic, it has been proven in [12] that fuzzy logic systems and feedforward neural networks are equivalent in essence.

In Sun et al. [13], a robust tracking control for robot manipulators in the presence of uncertainties and disturbances was proposed. A neural network-based sliding mode adaptive control was designed to ensure trajectory tracking by the robot manipulator. In Hernandez et al. [14], a neural PD with a second order sliding mode compensation technique was introduced. The scheme is able to guarantee asymptotic convergence of the error trajectories. In [15], in order to deal with actuator and model nonlinearities, a neural network-based controller was addressed. The scheme was experimentally tested in real time showing the advantages of the neural network. In [16], neural network contouring control using a Zebra-Zero robotic manipulator was presented. The paper in [17] used two modified optimal controllers based on neural networks. The closed-loop system trajectories were studied in a rigorous form although validation was presented with simulations. Other novel approaches of neural networks are presented in [18, 19].

Experimental evidence that neural networks are efficient in the control of a mechanical systems has been provided in [8, 15], for example. However, existing literature reveals a gap in the experimental evaluation of new controllers since most of the published works only consider numerical simulations to assess the performance of the proposed controllers.

In the present work a different approach is taken. In the theoretical part of our work, a neural network is used to approach the desired torque as a function of the desired joint position trajectory. The inner and outer weights of the neural network are adapted on-line by using an update law coming from a Lyapunov-like analysis of the closed-loop system trajectories. In addition, an extensive real-time implementation study in two different experimental set-ups has also been carried out. We prove by means of the experimental tests that the neural network compensation is really effective to reduce the joint tracking error. The real-time experiments show that an excellent tracking accuracy can be obtained by using adaptive neural network compensation plus a "small amount" of nonlinear PD control compensation. In summary, the contribution of this paper is twofold:

(1) a new nonlinear PD controller plus adaptive neural network compensation,

(2) a real-time experimental study in two different experimental set-ups.

Our approach is based on the adaptation of the weights of a neural network that only depends on the desired signals of joint position, velocity, and acceleration. In other words, the neural network used in the controller approaches the desired torque. In addition to the adaptive neural network feedforward compensation, the proposed controller is equipped with nonlinear PD terms, which are motivated by the convergence analysis. We prove that by using the proposed controller; the position and velocity error trajectories converge to zero, while the adapted neural network signals remain uniformly bounded. It is noteworthy that the proposed neural network controller resembles the PD control with feedforward compensation for robot manipulators, whose global asymptotical convergence proof was reported in [20, 21].

The proposed controller is tested in real time in a horizontal two-degree-of-freedom direct-driven arm and in a vertical single-link arm, which is affected by the gravitational force. In both experimental set-ups, the new scheme is implemented with and without neural network adaptation. The experimental results show that the tracking performance is drastically improved by using the new controller with the adaptive neural network feedforward compensation.

The difference of this document with respect to our previous work in [22] is that here the study of closed-loop trajectories is rigorously presented and experimental validation of the proposed scheme is evaluated in a detailed form.

The present document is organized as follows. Mathematical preliminaries, the robot model, and the control goal are given in Section 2. The proposed controller is discussed in Section 3. Section 4 is devoted to the real-time experimental result, while some concluding observations are given in Section 5.

2. Mathematical Preliminaries, Robot Model, and Control Goal

The notations $\lambda_{\min}\{A\}$ and $\lambda_{\max}\{A\}$ denote the minimum and maximum eigenvalues of a symmetric positive definite matrix $A \in \mathbb{R}^{n \times n}$, respectively. $\| \mathbf{x} \| = \sqrt{\mathbf{x}^T \mathbf{x}}$ stands for the norm of vector $\mathbf{x} \in \mathbb{R}^n$. $\| B \| = \sqrt{\lambda_{\max}\{B^T B\}}$ stands for the induced norm of a matrix $B(\mathbf{x}) \in \mathbb{R}^{m \times n}$ for all $\mathbf{x} \in \mathbb{R}^n$.

Given $A \in \mathbb{R}^{n \times m}$ the Frobenius norm is defined [1, 23] by

$$\|A\|_F^2 = \text{Tr}\left(A^T A\right) = \sum_{i,j} a_{ij}^2. \tag{1}$$

Other properties are

$$\langle A, B \rangle_F = \text{Tr}\left(A^T B\right),$$
$$\left|\langle A, B \rangle_F\right| \leq \|A\|_F \|B\|_F. \tag{2}$$

2.1. Properties on Hyperbolic Functions. Some properties on hyperbolic functions will be used. See [24, 25], where the cited properties are used in the design and analysis of controllers

for mechanical systems. The tangent hyperbolic function is defined as

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \tag{3}$$

where $x \in \mathbb{R}$. The tangent hyperbolic function can be arranged in a vector in the following way:

$$\tanh(\mathbf{z}) = \left[\tanh(z_1), \ldots, \tanh(z_n)\right]^T, \tag{4}$$

and the following properties are accomplished by $\tanh(\mathbf{z})$.

(a) For all $\mathbf{z} \in \mathbb{R}^n$, the Euclidean norm of $\tanh(\mathbf{z})$ satisfies

$$
\begin{aligned}
\|\tanh(\mathbf{z})\| &\leq \|\mathbf{z}\|, \\
\|\tanh(\mathbf{z})\| &\leq \sqrt{n}.
\end{aligned} \tag{5}
$$

(b) The time derivative of $\tanh(\mathbf{z})$ is given by

$$\frac{d}{dt}\tanh(\mathbf{z}) = \mathrm{Sech}^2(\mathbf{z})\,\dot{\mathbf{z}}, \tag{6}$$

where $\mathrm{Sech}^2(\mathbf{z}) = \mathrm{diag}\{\mathrm{sech}^2(z_1), \ldots, \mathrm{sech}^2(z_n)\}$ and

$$\mathrm{Sech}(x) = \frac{2}{e^x + e^{-x}} = \frac{1}{\cosh(x)}. \tag{7}$$

(c) The maximum eigenvalue of the matrix $\mathrm{sech}^2(\mathbf{x})$ is one for all $\mathbf{z} \in \mathbb{R}^n$; that is,

$$\lambda_{\max}\left\{\mathrm{sech}^2(\mathbf{z})\right\} = 1. \tag{8}$$

(d) Finally, the property

$$\sqrt{\sum_{i=1}^n \ln(\cosh(z_i))} \geq \frac{1}{\sqrt{2}}\|\tanh(\mathbf{z})\|, \quad \forall \mathbf{z} \in \mathbb{R}^n, \tag{9}$$

holds.

2.2. Neural Networks.

Let us recall the universal approximation property of the neural networks [1, 3]. A function $\mathbf{f}(\mathbf{x}) : \mathbb{R}^{N+1} \to \mathbb{R}^n$ can be approximated by

$$\mathbf{f}(\mathbf{x}) = W^T \boldsymbol{\sigma}\left(V^T \mathbf{x}\right) + \boldsymbol{\epsilon}, \quad \forall \mathbf{x} \in \mathbb{R}^{N+1}, \tag{10}$$

where $\mathbf{x} \in \mathbb{R}^{N+1}$ is the vector of input signals to the neural network, $V \in \mathbb{R}^{(N+1) \times L}$ and $W \in \mathbb{R}^{L \times n}$ are the input and output ideal weights, respectively, L is the number of neurons in the hidden layer, $N + 1$ is the number of input signals to the neural network, $\boldsymbol{\sigma} \in \mathbb{R}^L$ is the activation function in the hidden layer, and $\boldsymbol{\epsilon} \in \mathbb{R}^n$ is the approximation error with

$$|\epsilon_i| \leq \phi, \quad i = 1, \ldots, n, \tag{11}$$

where $\phi > 0$.

The output of an activation function, $\sigma_i : \mathbb{R} \to \mathbb{R}$, $i = 1, \ldots, L$, is used to define the output signal of a neuron from a modified combination of its input signals by compressing the signal. The function σ_i is usually between the values $0 \leq \sigma_i \leq 1$ or valued such that $-1 \leq \sigma_i \leq 1$ is satisfied.

In this paper, we have used as activation function σ_i the hyperbolic tangent function. Therefore, by defining $\mathbf{z} = V^T \mathbf{x}$,

$$\boldsymbol{\sigma}(\mathbf{z}) = \left[\tanh(z_1), \ldots, \tanh(z_L)\right]^T. \tag{12}$$

2.3. Robot Dynamics.

The dynamics in joint space of a serial-chain n-link robot manipulator considering the presence of friction at the robot joints can be written as [26, 27]

$$M(\mathbf{q})\,\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + F_v \dot{\mathbf{q}} = \boldsymbol{\tau}, \tag{13}$$

where $\mathbf{q} \in \mathbb{R}^n$ is a vector of joint positions, $M(\mathbf{q}) \in \mathbb{R}^{n \times n}$ is the symmetric positive definite inertia matrix, $C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \in \mathbb{R}^n$ is the vector of centripetal and Coriolis torques, $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ is the vector of gravitational torques, $F_v \in \mathbb{R}^{n \times n}$ is a diagonal positive definite matrix containing the viscous friction coefficients of each joint, and $\boldsymbol{\tau} \in \mathbb{R}^n$ is the vector of torques input.

The dynamics of n-link robotic manipulator expressed in (13) has the following properties, which hold for rigid-link revolute joint manipulators [26–28].

Property 1. The inertia matrix $M(\mathbf{q})$ is symmetric and positive definite; that is,

$$\lambda_{\min}\{M(\mathbf{q})\}\|\mathbf{x}\|^2 \leq \mathbf{x}^T M(\mathbf{q})\mathbf{x} \leq \lambda_{\max}\{M(\mathbf{q})\}\|\mathbf{x}\|^2. \tag{14}$$

Property 2. Assuming that the robot has revolute joints, the vector $C(\mathbf{q}, \mathbf{x})\mathbf{y}$ satisfies the bound

$$\|C(\mathbf{q}, \mathbf{x})\mathbf{y}\| \leq k_{C1}\|\mathbf{x}\|\|\mathbf{y}\|, \quad \forall \mathbf{q}, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \tag{15}$$

where $k_{C1} > 0$.

Property 3. Assume that the centrifugal and Coriolis torque matrix $C(\mathbf{q}, \dot{\mathbf{q}})$ is computed by means of the so-called Christoffel symbols of the first kind. Then,

$$\mathbf{x}^T\left[\dot{M}(\mathbf{q}) - 2C(\mathbf{q}, \dot{\mathbf{q}})\right]\mathbf{x} = 0, \quad \forall \mathbf{x}, \mathbf{q}, \dot{\mathbf{q}}. \tag{16}$$

Besides,

$$\dot{M}(\mathbf{q}) = C(\mathbf{q}, \dot{\mathbf{q}}) + C(\mathbf{q}, \dot{\mathbf{q}})^T. \tag{17}$$

Property 4. The so-called residual dynamics [29, 30] is defined by

$$
\begin{aligned}
\mathbf{h}(t, \mathbf{e}, \dot{\mathbf{e}}) = &\left[M(\mathbf{q}_d) - M(\mathbf{q})\right]\ddot{\mathbf{q}}_d \\
&+ \left[C(\mathbf{q}_d, \dot{\mathbf{q}}_d) - C(\mathbf{q}, \dot{\mathbf{q}})\right]\dot{\mathbf{q}}_d \\
&+ \mathbf{g}(\mathbf{q}_d) - \mathbf{g}(\mathbf{q}),
\end{aligned} \tag{18}
$$

where $\mathbf{e}$ will be defined later and $\mathbf{q}_d(t)$ is the desired joint position trajectory assumed to be bounded together with its

first and second time derivative. The residual dynamics (18) satisfies the following inequality [27]:

$$\|\mathbf{h}(t, \mathbf{e}, \dot{\mathbf{e}})\| \leq k_{h1} \|\dot{\mathbf{e}}\| + k_{h2} \|\tanh(\gamma \mathbf{e})\|, \tag{19}$$

where k_{h1} and k_{h1} are sufficiently large strictly positive constants that depend on the robot model parameters and $\gamma > 0$.

2.4. Control Goal. Let us define the joint position tracking error as

$$\mathbf{e}(t) = \mathbf{q}_d(t) - \mathbf{q}(t), \tag{20}$$

where $\mathbf{q}_d(t) \in \mathbb{R}^n$ denotes the desired joint position trajectory. The estimated weight errors are defined as

$$\begin{aligned} \widetilde{W} &= W - \widehat{W}, \\ \widetilde{V} &= V - \widehat{V}, \end{aligned} \tag{21}$$

where $V \in \mathbb{R}^{[N+1] \times L}$ are the ideal input weights, $\widehat{V} \in \mathbb{R}^{[N+1] \times L}$ is an estimation of the input weights, $W \in \mathbb{R}^{L \times n}$ are the ideal output weights, and $\widehat{W} \in \mathbb{R}^{L \times n}$ is an estimation of the output weights.

The desired time-varying trajectory $\mathbf{q}_d(t)$ is assumed to be three times differentiable and bounded for all time $t \geq 0$ in the sense

$$\begin{aligned} \|\mathbf{q}_d(t)\| &\leq \mu_1, \\ \|\dot{\mathbf{q}}_d(t)\| &\leq \mu_2, \\ \|\ddot{\mathbf{q}}_d(t)\| &\leq \mu_3, \end{aligned} \tag{22}$$

where μ_1, μ_2, and μ_3 denote known positive constants.

The control problem consists in designing a controller $\boldsymbol{\tau}(t)$ and update laws for the estimated weights $\widehat{V}(t)$ and $\widehat{W}(t)$ such that the signals $\mathbf{e}(t)$, $\dot{\mathbf{e}}(t)$, $\widehat{W}(t)$, and $\widetilde{V}(t)$ are uniformly bounded. In addition, the limit

$$\lim_{t \to \infty} \begin{bmatrix} \mathbf{e}(t) \\ \dot{\mathbf{e}}(t) \end{bmatrix} = \mathbf{0} \tag{23}$$

should be satisfied.

3. Proposed Adaptive Neuronal Network Compensation Controller

The proposed controller has a structure of a nonlinear PD controller plus adaptive neural network feedforward compensation. The design of the controller departs from the assumption that the desired torque can be approached by a neural network and then it can be estimated on-line by means of proper adaptation laws for the input and output neural network weights.

3.1. Proposed Scheme. The development of the proposed approach is presented in a constructive form. First, let us consider the robot dynamics (13) evaluated along the desired position $\mathbf{q}_d \in \mathbb{R}^n$ such that the desired torque $\boldsymbol{\tau}_d \in \mathbb{R}^n$ can be founded by

$$M(\mathbf{q}_d)\ddot{\mathbf{q}}_d + C(\mathbf{q}_d, \dot{\mathbf{q}}_d)\dot{\mathbf{q}}_d + \mathbf{g}(\mathbf{q}_d) + F_v\dot{\mathbf{q}}_d = \boldsymbol{\tau}_d, \tag{24}$$

by combining (13) and (24) and using the tracking error defined in (20) the following equation is obtained:

$$M(\mathbf{q})\ddot{\mathbf{e}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{e}} + \mathbf{h}(t, \mathbf{e}, \dot{\mathbf{e}}) + F_v\dot{\mathbf{e}} = \boldsymbol{\tau}_d - \boldsymbol{\tau}, \tag{25}$$

where $\mathbf{h}(t, \mathbf{e}, \dot{\mathbf{e}}) \in \mathbb{R}^n$ is the so-called residual dynamics [29, 30], defined by (18).

By using the universal approximation property of the neural networks in (10), the desired torque in (24) can be approached as

$$\boldsymbol{\tau}_d = W^T \boldsymbol{\sigma}(V^T \mathbf{x}_d) + \boldsymbol{\epsilon}, \tag{26}$$

where

$$\mathbf{x}_d = \begin{bmatrix} \mathbf{q}_d^T & \dot{\mathbf{q}}_d^T & \ddot{\mathbf{q}}_d^T & 1 \end{bmatrix}^T \in \mathbb{R}^{N+1} \tag{27}$$

is the vector of input signals to the neural network. Notice that in this case $N = 3n$.

Now, we are in position to introduce the following tracking controller:

$$\boldsymbol{\tau} = \widehat{W}^T \boldsymbol{\sigma}(\widehat{V}^T \mathbf{x}_d) + K_p \tanh(\gamma \mathbf{e}) + K_d \dot{\mathbf{e}} + \Delta \operatorname{sign}(\mathbf{r}), \tag{28}$$

where K_p, K_d, and Δ are diagonal positive definite matrices, γ is a positive scalar, $\widehat{V}$ is the estimated input weight, $\widehat{W}$ is the estimated output weight,

$$\mathbf{r} = \dot{\mathbf{e}} + \alpha \tanh(\gamma \mathbf{e}), \tag{29}$$

$$\operatorname{sign}(\mathbf{r}) = \begin{bmatrix} \operatorname{sign}(r_1) & \cdots & \operatorname{sign}(r_n) \end{bmatrix}^T \in \mathbb{R}^n, \tag{30}$$

with

$$\operatorname{sign}(x) = \begin{cases} 1, & \text{for } x > 0, \\ 0, & \text{for } x = 0, \\ -1, & \text{for } x < 0. \end{cases} \tag{31}$$

The proposed update laws for the estimated input and output weights, denoted as $\widehat{V}$ and $\widehat{W}$, respectively, are

$$\dot{\widehat{V}} = R\mathbf{x}_d \mathbf{r}^T \widehat{W}\widehat{\boldsymbol{\sigma}}', \tag{32}$$

$$\dot{\widehat{W}} = F\widehat{\boldsymbol{\sigma}}\mathbf{r}^T - F\widehat{\boldsymbol{\sigma}}'\widehat{V}^T \mathbf{x}_d \mathbf{r}^T, \tag{33}$$

where $R \in \mathbb{R}^{[N+1] \times [N+1]}$ and $F \in \mathbb{R}^{L \times L}$ are positive definite matrices, $\widehat{\boldsymbol{\sigma}} = \boldsymbol{\sigma}(\widehat{V}^T \mathbf{x}_d)$ and

$$\widehat{\boldsymbol{\sigma}}' = \frac{\partial \boldsymbol{\sigma}(\mathbf{x})}{\partial \mathbf{x}}, \tag{34}$$

with $\widehat{V}^T \mathbf{x}_d$.

3.2. Closed-Loop System Derivation. By substituting (26) and (28) in (25), the equation

$$\begin{aligned} M(\mathbf{q})\ddot{\mathbf{e}} &+ C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{e}} + \mathbf{h}(t, \mathbf{e}, \dot{\mathbf{e}}) + F_v\dot{\mathbf{e}} \\ &= -K_p \tanh(\gamma \mathbf{e}) - K_d \dot{\mathbf{e}} \\ &\quad - \Delta \operatorname{sign}(\mathbf{r}) + W^T \boldsymbol{\sigma}(V^T \mathbf{x}_d) - \widehat{W}^T \boldsymbol{\sigma}(\widehat{V}^T \mathbf{x}_d) + \boldsymbol{\epsilon} \end{aligned} \tag{35}$$

is obtained. The weight error is defined as in (21) and

$$\widetilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma} - \widehat{\boldsymbol{\sigma}} = \boldsymbol{\sigma}\left(V^T \mathbf{x}_d\right) - \boldsymbol{\sigma}\left(\widehat{V}^T \mathbf{x}_d\right). \tag{36}$$

Multilayered neural networks are nonlinear in the weights V and Taylor's series can be used to approximate the activation function $\boldsymbol{\sigma}$. Thus,

$$\boldsymbol{\sigma}\left(V^T \mathbf{x}_d\right) = \boldsymbol{\sigma}\left(\widehat{V}^T \mathbf{x}_d\right) + \boldsymbol{\sigma}'\left(\widehat{V}^T \mathbf{x}_d\right) \widetilde{V}^T \mathbf{x}_d + O\left(\widetilde{V}^T \mathbf{x}_d\right)^2, \tag{37}$$

where $O(\widetilde{V}^T \mathbf{x}_d)^2$ represents the higher order terms. Approximation of activation function via Taylor's series has been used in, for example, [1, 3]. Therefore, substituting (21) and (37) in (35) and simplifying, we obtain the following expression:

$$
\begin{aligned}
M\ddot{\mathbf{e}} = & -\mathbf{h} - C\dot{\mathbf{e}} - K_p \tanh(\gamma \mathbf{e}) - \left[K_d + F_v\right]\dot{\mathbf{e}} \\
& - \Delta \operatorname{sign}(\mathbf{r}) + \widehat{W}^T \widehat{\boldsymbol{\sigma}}' \widetilde{V}^T \mathbf{x}_d - \widetilde{W}^T \widehat{\boldsymbol{\sigma}}' \widehat{V}^T \mathbf{x}_d \\
& + \widetilde{W}^T \boldsymbol{\sigma} + \boldsymbol{\omega}(t),
\end{aligned}
\tag{38}
$$

where $M = M(\mathbf{q})$, $\mathbf{h} = \mathbf{h}(\mathbf{e}, \dot{\mathbf{e}})$, $C = C(\mathbf{q}, \dot{\mathbf{q}})$, and

$$\boldsymbol{\omega}(t) = \widetilde{W}^T \widehat{\boldsymbol{\sigma}}' V^T \mathbf{x}_d + WO^2 + \boldsymbol{\epsilon}, \tag{39}$$

with $O^2 = O(\widetilde{V}^T \mathbf{x}_d)^2$.

Finally, the dynamics of $[\mathbf{e}^T \; \dot{\mathbf{e}}^T]^T \in \mathbb{R}^{2n}$ is given by

$$\frac{d}{dt}\mathbf{e} = \dot{\mathbf{e}},$$

$$
\begin{aligned}
\frac{d}{dt}\dot{\mathbf{e}} = M^{-1}\Big[& -\mathbf{h} - C\dot{\mathbf{e}} - K_p \tanh(\gamma \mathbf{e}) - \left[K_d + F_v\right]\dot{\mathbf{e}} \\
& - \Delta \operatorname{sign}(\mathbf{r}) + \widehat{W}^T \widehat{\boldsymbol{\sigma}}' \widetilde{V}^T \mathbf{x}_d - \widetilde{W}^T \widehat{\boldsymbol{\sigma}}' \widehat{V}^T \mathbf{x}_d \\
& + \widetilde{W}^T \widehat{\boldsymbol{\sigma}} + \boldsymbol{\omega}(t) \Big].
\end{aligned}
\tag{40}
$$

By using the definition of the weight errors (21), we can rewrite the proposed update laws (32)-(33) as

$$
\begin{aligned}
\frac{d}{dt}\widetilde{V} &= -R\mathbf{x}_d \mathbf{r}^T W \widehat{\boldsymbol{\sigma}}' + R\mathbf{x}_d \mathbf{r}^T \widetilde{W} \widehat{\boldsymbol{\sigma}}', \\
\frac{d}{dt}\widetilde{W} &= -F\widehat{\boldsymbol{\sigma}}\mathbf{r}^T + F\widehat{\boldsymbol{\sigma}}' V^T \mathbf{x}_d \mathbf{r}^T \\
&\quad - F\widehat{\boldsymbol{\sigma}}' \widetilde{V}^T \mathbf{x}_d \mathbf{r}^T.
\end{aligned}
\tag{41}
$$

The overall closed-loop system is given by (40) and (41).

3.3. Convergence Analysis. The assumption

$$k_\omega \geq \|\boldsymbol{\omega}(t)\|, \quad \forall t \geq 0, \tag{42}$$

with $k_\omega > 0$ and $\boldsymbol{\omega}(t)$ defined in (39), will be used in the next.

Concerning the trajectories of the closed-loop system (40)-(32), we have the following result.

Proposition 1. *One assumes that the desired trajectory $\mathbf{q}_d(t)$ is bounded as (22). Then, provided that $\lambda_{\min}\{K_p\}$, $\lambda_{\min}\{K_d\}$ and $\lambda_{\min}\{\Delta\}$ are sufficiently large, there always exist strictly positive constants $\alpha^*_{\max}$ and $\alpha^*_{\min}$ such that*

$$\alpha^*_{\min} < \alpha < \alpha^*_{\max} \tag{43}$$

guarantees that the trajectories $\mathbf{e}(t)$, $\dot{\mathbf{e}}(t)$, $\widetilde{W}(t)$, and $\widetilde{V}(t)$ of the overall closed-loop system (40)-(32) are uniformly bounded. In addition, the limit

$$\lim_{t \to \infty} \begin{bmatrix} \mathbf{e}(t) \\ \dot{\mathbf{e}}(t) \end{bmatrix} = \mathbf{0} \tag{44}$$

is satisfied.

Proof. Let us consider the function

$$
\begin{aligned}
U(t, \mathbf{e}, \dot{\mathbf{e}}) = & \frac{1}{2}\dot{\mathbf{e}}^T M\dot{\mathbf{e}} + \sum_{i=1}^n K_{pi}\gamma^{-1}\ln\left(\cosh(\gamma \mathbf{e}_i)\right) \\
& + \alpha \tanh(\gamma \mathbf{e})^T M\dot{\mathbf{e}} + \frac{1}{2}T_r\left(\widetilde{W}^T F^{-1}\widetilde{W}\right) \\
& + \frac{1}{2}T_r\left(\widetilde{V}^T R^{-1}\widetilde{V}\right),
\end{aligned}
\tag{45}
$$

which is positive definite in terms of the state space of the closed-loop system (40)-(32).

By using property (9), a lower bound on $U(t, \mathbf{e}, \dot{\mathbf{e}})$ can be computed as follows:

$$
\begin{aligned}
U(t, \mathbf{e}, \dot{\mathbf{e}}) \geq \; & \boldsymbol{\eta}^T \underbrace{\begin{bmatrix} \dfrac{1}{2}\lambda_{\min}\{M\} & -\dfrac{\alpha}{2}\lambda_{\max}\{M\} \\[2mm] -\dfrac{\alpha}{2}\lambda_{\max}\{M\} & \dfrac{\gamma^{-1}}{2}\lambda_{\min}\{K_p\} \end{bmatrix}}_{P} \boldsymbol{\eta} \\
& + \frac{1}{2}T_r\left(\widetilde{W}^T F^{-1}\widetilde{W}\right) + \frac{1}{2}T_r\left(\widetilde{V}^T R^{-1}\widetilde{V}\right),
\end{aligned}
\tag{46}
$$

with $\boldsymbol{\eta} = [\|\dot{\mathbf{e}}\| \; \|\tanh(\gamma \mathbf{e})\|]^T$.

If P is positive definite, then the function $U(t, \mathbf{e}, \dot{\mathbf{e}})$ is globally positive definite and radially unbounded. By using Sylvester's Theorem, the sufficient and necessary condition for P to be positive definite is

$$0 < \alpha < \frac{\sqrt{\gamma^{-1}\lambda_{\min}\{K_p\}\lambda_{\min}\{M\}}}{\lambda_{\max}\{M\}}. \tag{47}$$

Next step in the proof is to compute the time derivative of $U(t, \mathbf{e}, \dot{\mathbf{e}})$ along the closed-loop system trajectories (40) and (32). Thus, we have that

$$
\begin{aligned}
\dot{U}(t, \mathbf{e}, \dot{\mathbf{e}}) = & -\dot{\mathbf{e}}^T\left[K_d + F_v\right]\dot{\mathbf{e}} - \mathbf{r}^T \mathbf{h} + \mathbf{r}^T\left[\boldsymbol{\omega} - \Delta \operatorname{sign}(\mathbf{r})\right] \\
& + \alpha\gamma\dot{\mathbf{e}}^T M\operatorname{Sech}^2(\gamma \mathbf{e})\dot{\mathbf{e}} + \alpha \tanh(\gamma \mathbf{e})^T C^T \dot{\mathbf{e}} \\
& - \alpha \tanh(\gamma \mathbf{e})^T\left[K_d + F_v\right]\dot{\mathbf{e}} \\
& - \alpha \tanh(\gamma \mathbf{e})^T K_p \tanh(\gamma \mathbf{e}),
\end{aligned}
\tag{48}
$$

which was obtained thanks to the property (6), the robot model properties (16)-(17), and the facts

$$\text{Tr}\left(\widetilde{W}^T \widehat{\boldsymbol{\sigma}} \mathbf{r}^T\right) = \text{Tr}\left(\mathbf{r}^T \widetilde{W}^T \widehat{\boldsymbol{\sigma}}\right),$$

$$\text{Tr}\left(\widetilde{W}^T \widehat{\boldsymbol{\sigma}}' \widehat{V}^T \mathbf{x}_d \mathbf{r}^T\right) = \text{Tr}\left(\mathbf{r}^T \widetilde{W}^T \widehat{\boldsymbol{\sigma}}' \widehat{V}^T \mathbf{x}_d\right), \quad (49)$$

$$\text{Tr}\left(\widetilde{V}^T \mathbf{x}_d \mathbf{r}^T \widehat{W}^T \widehat{\boldsymbol{\sigma}}'\right) = \text{Tr}\left(\mathbf{r}^T \widehat{W}^T \widehat{\boldsymbol{\sigma}}' \widetilde{V}^T \mathbf{x}_d\right).$$

An upper bound on each term of $\dot{U}(t, \mathbf{e}, \dot{\mathbf{e}})$ is computed as follows:

$$-\dot{\mathbf{e}}^T \left[K_d + F_v\right] \dot{\mathbf{e}} \le -\lambda_{\min}\left\{K_d + F_v\right\} \|\dot{\mathbf{e}}\|,$$

$$-\mathbf{r}^T \mathbf{h} \le \alpha k_{h2} \|\tanh(\gamma \mathbf{e})\|^2$$

$$+ \left[\alpha k_{h1} + k_{h2}\right] \|\tanh(\gamma \mathbf{e})\| \|\dot{\mathbf{e}}\| + k_{h1} \|\dot{\mathbf{e}}\|^2,$$

$$\mathbf{r}^T \left[\boldsymbol{\omega} - \Delta \operatorname{sign}(\mathbf{r})\right] \le -\left[\lambda_{\min}\{\Delta\} - k_\omega\right] \sum_{i=1}^n |r_i|,$$

$$\alpha \gamma \dot{\mathbf{e}}^T M \operatorname{Sech}^2(\gamma \mathbf{e}) \dot{\mathbf{e}} \le \alpha \gamma \lambda_{\max}\{M\} \|\dot{\mathbf{e}}\|^2,$$

$$\alpha \tanh(\gamma \mathbf{e})^T C^T \dot{\mathbf{e}} \le \alpha k_{C1} \mu_2 \|\tanh(\gamma \mathbf{e})\| \|\dot{\mathbf{e}}\| \quad (50)$$

$$+ \alpha k_{C1} \sqrt{n} \|\dot{\mathbf{e}}\|^2,$$

$$-\alpha \tanh(\gamma \mathbf{e})^T \left[K_d + F_v\right] \dot{\mathbf{e}}$$

$$\le \alpha \lambda_{\max}\left\{K_d + F_v\right\} \|\tanh(\gamma \mathbf{e})\| \|\dot{\mathbf{e}}\|,$$

$$-\alpha \tanh(\gamma \mathbf{e})^T K_p \tanh(\gamma \mathbf{e})$$

$$\le -\alpha \lambda_{\min}\left\{K_p\right\} \|\tanh(\gamma \mathbf{e})\|^2,$$

where the property of the residual dynamics $\mathbf{h}$ in (19), assumption (42), property (8), property (15), and the fact that $\|\dot{\mathbf{q}}\| \le \mu_2 + \|\dot{\mathbf{e}}\|$, with μ_2 defined in (22), were used.

With the computed bounds on each term of $\dot{U}(t, \mathbf{e}, \dot{\mathbf{e}})$, we can write

$$\dot{U}(t, \mathbf{e}, \dot{\mathbf{e}}) \le -\mathbf{p}_1^T Q_1 \mathbf{p}_1 - \left[\lambda_{\min}\{\Delta\} - k_\omega\right] \sum_{i=1}^n |r_i|, \quad (51)$$

where

$$\mathbf{p}_1 = \begin{bmatrix} \|\tanh(\gamma \mathbf{e})\| \\ \|\dot{\mathbf{e}}\| \end{bmatrix}, \quad (52)$$

$$Q_1 = \begin{bmatrix} \alpha a & -\dfrac{1}{2}\left[\alpha d + e\right] \\ -\dfrac{1}{2}\left[\alpha d + e\right] & b - \alpha c \end{bmatrix}, \quad (53)$$

with

$$a = \lambda_{\min}\left\{K_p\right\} - k_{h2},$$

$$b = \lambda_{\min}\left\{K_d + F_v\right\} - k_{h1},$$

$$c = \gamma \lambda_{\max}\{M\} + k_{C1}\sqrt{n}, \quad (54)$$

$$d = k_{h2} + k_{C1}\mu_2 + \lambda_{\max}\left\{K_d + F_v\right\},$$

$$e = k_{h1}.$$

It is easy to observe that if Q_1 in (53) is positive definite and if

$$\lambda_{\min}\{\Delta\} > k_\omega, \quad (55)$$

with k_ω defined in (42), then the function $\dot{U}(t, \mathbf{e}, \dot{\mathbf{e}})$ is negative semidefinite. Besides, notice that the matrix Q_1 in (53) can be rewritten as

$$Q_1 = Q_a + Q_b, \quad (56)$$

where

$$Q_a = \begin{bmatrix} \dfrac{1}{2}\alpha a & -\dfrac{1}{2}\alpha d \\ -\dfrac{1}{2}\alpha d & \dfrac{1}{2}\left[b - 2\alpha c\right] \end{bmatrix},$$

$$\quad (57)$$

$$Q_b = \begin{bmatrix} \dfrac{1}{2}\alpha a & -\dfrac{1}{2}e \\ -\dfrac{1}{2}e & \dfrac{1}{2}b \end{bmatrix}.$$

If Q_a and Q_b are positive definite then Q_1 is also positive definite. By applying Sylvester's criterion, the matrix Q_a is positive definite if

$$a > 0 \implies \lambda_{\min}\left\{K_p\right\} > k_{h2},$$

$$\alpha < \frac{b}{2c} \implies \lambda_{\min}\left\{K_d + F_v\right\} > k_{h1}, \quad (58)$$

$$\alpha < \frac{ab}{2ab + d^2},$$

while Q_b is positive definite if

$$a > 0 \implies \lambda_{\min}\left\{K_p\right\} > k_{h2},$$

$$b > 0 \implies \lambda_{\min}\left\{K_d + F_v\right\} > k_{h1}, \quad (59)$$

$$\alpha > \frac{e^2}{ab}.$$

Therefore, the selection of large enough control gains Δ, K_p and K_d guarantee the existence of α satisfying (43) so that the function $U(t, \mathbf{e}, \dot{\mathbf{e}})$ in (45) is positive definite and radially unbounded, and $\dot{U}(t, \mathbf{e}, \dot{\mathbf{e}})$ in (48) is a negative semidefinite function. Hence the trajectories $\mathbf{e}(t)$, $\dot{\mathbf{e}}(t)$, $\widetilde{W}(t)$, and $\widetilde{V}(t)$ of the overall closed-loop system (40)-(32) are uniformly bounded.

By integrating both sides of (51) it is possible to prove that

$$\int_0^t \|\mathbf{p}_1(\rho)\|^2 d\rho \le \frac{U(0, \mathbf{e}(0), \dot{\mathbf{e}}(0))}{\lambda_{\min}\{Q_1\}}, \quad (60)$$

where $\mathbf{p}_1(t)$ is defined in (52). Then, by invoking Barbalat's lemma [31], the limit (44) is assured. □
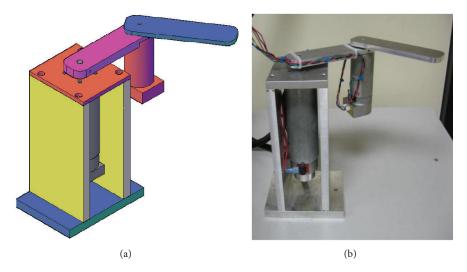
(a)

(b)

FIGURE 1: Experimental horizontal two-degree-of-freedom robot manipulator.
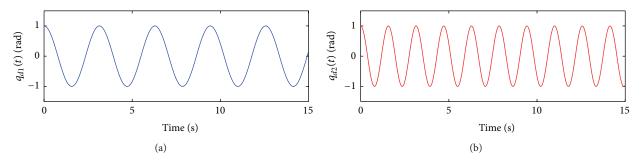

(a)

(b)

FIGURE 2: Desired joint trajectories $q_{d1}(t)$ and $q_{d2}(t)$ used in the real-time evaluations.

4. Experimental Results

Two sets of experiments in different experimental set-ups have been carried out in order to assess the performance of the proposed adaptive neural network controller (28) and (32)-(33). The used experimental systems are

(i) a horizontal two-degree-of-freedom arm,

(ii) a vertical one-degree-of-freedom arm, which is affected by the gravitational force.

In each set of experiments the applied torque (28) is implemented with and without adaptive neural network compensation. The purpose of the experiments is to show the benefit of the adaptive neural network feedforward compensation.

4.1. Planar Two-Degree-of-Freedom Robot. To carry out experiments, a planar two-degree-of-freedom direct-driven arm has been used. See Figure 1 for a CAD drawing and picture. The system is composed by two DC *Pittman* motors operated in current mode with two *Advanced Motion Controls* servo amplifiers. A *Sensoray 626* I/O card is used to read encoder signals with quadrature included and to transfer control commands through the D/A channels. A PC running *Windows XP*, *Matlab*, *Simulink,* and *Real-Time Windows*

Target is used to execute controllers in real time at 1 [kHz] sampling rate.

Two experiments have been conducted corresponding to the implementation of a nonlinear PD controller which does not consider any adaptation and the proposed adaptive neural network-based controller in (28).

In reference to Tables 1 and 2, and the model (13), the elements of the experimental robot model are

$$M(\mathbf{q}) = \begin{bmatrix} \theta_1 + 2\theta_2 \cos(q_2) & \theta_3 + \theta_2 \cos(q_2) \\ \theta_3 + \theta_2 \cos(q_2) & \theta_3 \end{bmatrix},$$

$$C(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} -\theta_2 \sin(q_2)\dot{q}_2 & -\theta_2 \sin(q_2)[\dot{q}_1 + \dot{q}_2] \\ \theta_2 \sin(q_2)\dot{q}_1 & 0 \end{bmatrix}, \quad (61)$$

$$F_v = \text{diag}\{\theta_4, \theta_5\},$$

$$\mathbf{g}(\mathbf{q}) = \mathbf{0}.$$

It is noteworthy that the robot is moving in the horizontal plane, whereby the vector of gravitational forces is null. Table 2 shows a numerical estimation of the parameters $\theta_i \in \mathbb{R}$, which was obtained by using the filtered dynamic model and the classical least-squares identification method;
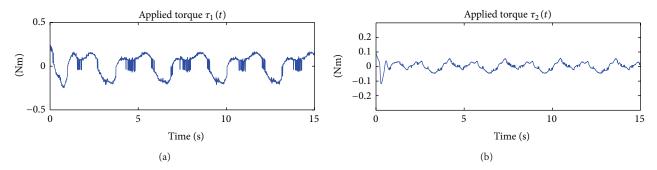
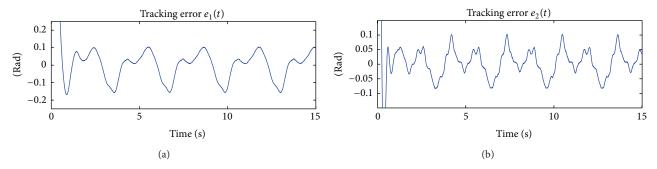FIGURE 3: Applied torque by using the PD + Δ sign (**r**) scheme which does not consider any adaptation.

FIGURE 4: Tracking errors $e_1(t)$ and $e_2(t)$ by using the PD + Δ sign (**r**) scheme which does not consider any adaptation.

see [32, 33]. The links of the arm are made of aluminum and both have a length of 0.15 [m].

4.1.1. Desired Trajectory.

The desired joint velocity trajectory chosen to achieve the experimental tests is given by

$$\mathbf{q}_d(t) = \begin{bmatrix} 1.0 \cos{(2.0t)} \\ 1.0 \cos{(4.0t)} \end{bmatrix}. \tag{62}$$

It is clear that the desired joint trajectory $\mathbf{q}_d(t)$ satisfies the bounding assumption (22). See Figure 2 to observe the time evolution of the desired trajectory (62).

4.1.2. Results.

By removing the adaptive neural network compensation in the proposed controller (28), the following nonlinear PD-type controller is obtained:

$$\boldsymbol{\tau} = K_p \tanh{(\gamma \mathbf{e})} + K_d \dot{\mathbf{e}} + \Delta \operatorname{sign}(\mathbf{r}), \tag{63}$$

with **r** defined in (29). Hereafter, the controller (63) will be denoted as PD + Δ sign (**r**).

The PD + Δ sign (**r**) controller in (63) has been implemented in real time with the following control gains:

$$K_p = \operatorname{diag}\{0.2, 0.1\},$$

$$K_d = \operatorname{diag}\{0.1, 0.01\}, \tag{64}$$

$$\Delta = \operatorname{diag}\{0.05, 0.005\},$$

and with

$$\alpha = 1.0, \qquad \gamma = 5.0. \tag{65}$$

TABLE 1: Physical parameters of the experimental robot arm.

Description	Notation	Units
Length of link 1	l_1	m
Length of link 2	l_2	m
Distance to the center of mass of link 1	l_{c1}	m
Distance to the center of mass of link 2	l_{c2}	m
Mass of link 1	m_1	kg
Mass of link 2	m_2	kg
Inertia rel. to center of mass (link 1)	I_1	kg-m^2
Inertia rel. to center of mass (link 2)	I_2	kg-m^2

The results of implementing in real time the PD + Δ sign (**r**) controller in (63) can be appreciated in Figure 3, which shows the time evolution of the applied torques $\tau_1(t)$ and $\tau_2(t)$, and in Figure 4, that depicts the time evolution of the tracking error $e_1(t)$ and $e_2(t)$.

On the other hand, the new control scheme in (28) will be referred to as ANN + PD + Δ sign (**r**). This controller was implemented by using $L = 20$, which is the number of neurons. The neural network requires the input vector

$$\mathbf{x}_d = \begin{bmatrix} q_{d1} & \dot{q}_{d1} & \ddot{q}_{d1} & q_{d2} & \dot{q}_{d2} & \ddot{q}_{d2} & 1 \end{bmatrix}^T \in \mathbb{R}^7. \tag{66}$$

The vector $\mathbf{x}_d(t)$ describes the signals that are used in the inner layer of the neural network.

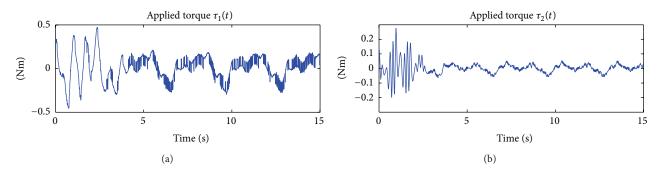In order to establish a fair comparison scenery, the control gains (64) and (65), which were used in the PD + Δ sign (**r**)

(a)

(b)

FIGURE 5: Applied torque by using the ANN + PD + Δ sign ($\mathbf{r}$) scheme which considers adaptation.
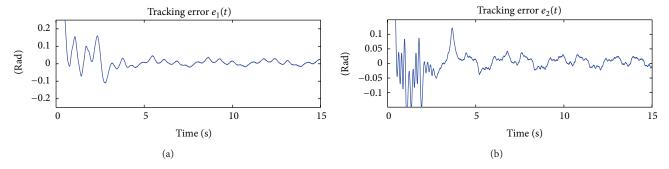

(a)

(b)

FIGURE 6: Tracking errors $e_1(t)$ and $e_2(t)$ by using the ANN + PD + Δ sign ($\mathbf{r}$) scheme which considers adaptation.

TABLE 2: Estimated parameters of the experimental robot arm; see (61) for reference.

Parameter	Definition	Value	Unit
θ_1	$m_1 l_{c1}^2 + m_2 \left[l_1^2 + l_{c2}^2\right] + I_1 + I_2$	0.0363	Kg m^2
θ_2	$m_2 l_1 l_{c2}$	0.0028	Kg m^2
θ_3	$m_2 l_{c2}^2 + I_2$	0.0023	Kg m^2
θ_4	f_{v1}	0.0084	Nm sec
θ_5	f_{v1}	0.0024	Nm sec

controller in (63), have been used in the real-time implementation of the proposed controller ANN + PD + Δ sign ($\mathbf{r}$) in (28). Besides, the gains

$$R = 0.2I_7, \qquad F = 0.5I_{20}, \qquad (67)$$

with I_n meaning the identity matrix of dimension n, were used the adaptation update laws (32) and (33), respectively.

The results of implementing the proposed controller ANN + PD + Δ sign ($\mathbf{r}$) in (28) is shown in Figure 5, where the time evolution of the applied torques $\tau_1(t)$ and $\tau_2(t)$ are represented, and in Figure 6, where the obtained tracking errors $e_1(t)$ and $e_2(t)$ are described.

4.1.3. Observations. Figures 4 and 6 show the tracking error $\mathbf{e}(t) \in \mathbb{R}^2$ for the controller PD + Δ sign ($\mathbf{r}$) in (63), which does not consider adaptation, and the new scheme ANN + PD + Δ sign ($\mathbf{r}$) in (28), which is based on the adaptation of neural network. From these figures the performance of the controllers can be assessed.

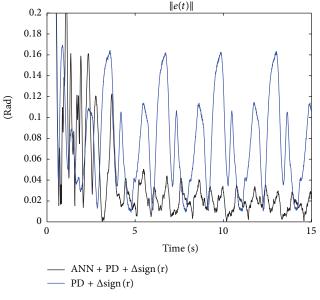—— ANN + PD + Δsign (r)
—— PD + Δsign (r)

FIGURE 7: Time evolution of the norm of the tracking error $\|\mathbf{e}(t)\|$.

For both implementations, one can consider a steady state behavior after 10 [sec]. In order to assess the performance of the controllers, the maximum peak-to-peak tracking error for each joint and for each implementation has been computed. Table 3 shows the performance of the two controllers, where the notation PTPV stands for "peak-to-peak value." We can see that by using the new controller the tracking performance
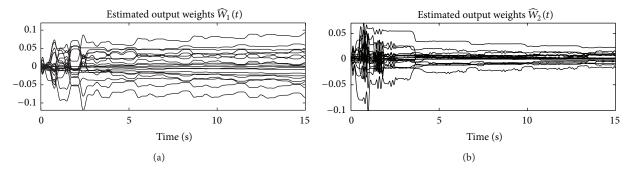
FIGURE 8: Time evolution of the output weights $\widehat{W}_1(t) \in \mathbb{R}^{10\times 1}$ and $\widehat{W}_2(t) \in \mathbb{R}^{10\times 1}$.

TABLE 3: Performance of the two controllers: maximum peak-to-peak value (PTPV) of the tracking errors after the settling time.

Index [rad]	PD + Δsign(r)	ANN + PD + Δsign(r)
$\max_{t\geq 10}\{$PTPV of $e_1(t)\}$ (rad)	0.260	0.046
$\max_{t\geq 10}\{$PTPV of $e_2(t)\}$ (rad)	0.187	0.047

TABLE 4: Identified parameters of the vertical one degree-of-freedom robot.

Parameter	Unit	Estimation
Lumped inertia M	A(s^2/rad)	0.0292
Lumped viscous friction coefficient B	A(s/rad)	0.0298
Lumped gravitational load N	A	2.2387
Inductance L	H	0.0031
Resistance R	Ω	0.9322
Back electromotive force coefficient k_B	N(m/A)	0.0246

is drastically improved. In particular, the percentage of improvement for joint 1 is 82.3% and for joint 2 is 74.9%.

On other hand, Figure 7 shows the norm of the tracking error $\| \mathbf{e}(t) \|$, which is another form to compare the tracking performance of a controller with respect to the other. In this figure, the improvement in the tracking of the desired trajectory by using the ANN + PD + Δ sign ($\mathbf{r}$) scheme in (28) is clearly observed. After 10 [sec], the peak value of the $\| \mathbf{e}(t) \|$ by using the PD + Δ sign ($\mathbf{r}$) scheme in (63) is 0.161 [rad] and by using the ANN + PD + Δ sign ($\mathbf{r}$) controller in (28) is 0.035 [rad]. Thus, the improvement using the new scheme is 78.3%.

Finally, the output weights $\widehat{W}_1(t) \in \mathbb{R}^{10\times 1}$ and $\widehat{W}_2(t) \in \mathbb{R}^{10\times 1}$ for joints 1 and 2, respectively, obtained in the real-time implementation of the new controller (28) are observed in Figure 8. A small value of such a weights is enough to improve the tracking performance drastically.

4.2. Vertical One Degree-of-Freedom Robot. This experimental system consists in an *Advanced Motion Controls* servo amplifier operated in voltage mode and a *Nema 34* brushed direct current motor which has attached a pendulum as shown in Figure 9. Like the two-degree-of-freedom robot, a data acquisition board and Matlab were used to implement the controllers in real time at 1 [kHz] sampling rate.

The model of this system corresponds to a pendulum with viscous friction which is actuated by a armature-controlled direct current motor, which can be written as [34]

$$M\ddot{q} + B\dot{q} + N\sin\left(q\right) = I,$$
$$L\frac{dI}{dt} + RI + k_B\dot{q} = v, \tag{68}$$

where M, B, N, L, R, and k_B are strictly positive constants; I denotes the armature current, and v is the voltage input.

By using the least-squares identification method we have estimated the numerical value of the constant parameters, which are shown in Table 4. Let us notice that the motor shaft has attached a pendulum whose mass is 0.285 [Kg] and length is 0.1803 [m] (equivalent to 7.1 [in]) from the rotation axis to the arm tip.

The tested controllers corresponds to

$$v = k_p \tanh\left(\gamma e\right) + k_d\dot{q} + \delta\,\mathrm{sign}\left(r\right), \tag{69}$$

$$v = \widehat{W}^T\boldsymbol{\sigma}\left(\widehat{V}^T\mathbf{x}_d\right) + k_p \tanh\left(\gamma e\right) + k_d\dot{q} + \delta\,\mathrm{sign}\left(r\right). \tag{70}$$

In analogy with experimental tests carried out in the horizontal two-degree-of-freedom robot, the controller in (69) will be denoted as PD + δ sign (r), while the scheme (70) will be referred as ANN + PD + δ sign (r). In both controllers $r = \dot{e} + \alpha \tanh(\gamma e)$. The adaptation signals $\widehat{V}(t)$ and $\widehat{W}(t)$ in (70) are computed by using the expressions (32) and (32), respectively.

The idea is to compare the performance and robustness of the controllers (69) and (70) in a system affected by the gravitational force and the electrical dynamics of the motor.

In this case, the model of the system of the robot was assumed to be unknown, whereby the tested controllers were tuned by trial and error until an acceptable performance was obtained.

4.2.1. Desired Trajectory. The proposed desired trajectory for this set of experiments was

$$\mathrm{q}_d\left(t\right) = 1.5 + 2\sin\left(3t\right)\ [\mathrm{rad}]. \tag{71}$$

FIGURE 9: Experimental one-degree-of-freedom arm (pendulum) subject to gravitational force.
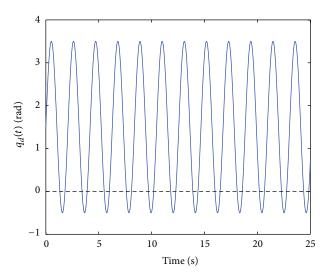

FIGURE 10: Desired joint trajectory $q_d(t)$ the real-time evaluations.

Notice that with trajectory the effect of the gravitational force is maximized. Figure 10 shows the time history of the desired joint trajectory $q_d(t)$ in (71).

4.2.2. Results. Like the case of the experiments with the two-degree-of-freedom robot, we have selected the control gains so that most of the control action is contributed by the adaptive neural network part of the control.

Specifically, we selected the gains

$$k_p = 1, \qquad k_d = 0.2,$$
$$\alpha = 1.5, \qquad \gamma = 5. \tag{72}$$

The adaptation of the parameters is given by using (32) and (33) with $\mathbf{x}_d = [1 \ q_d \ \dot{q}_d \ \ddot{q}_d]^T \in \mathbb{R}^4$ and the gains

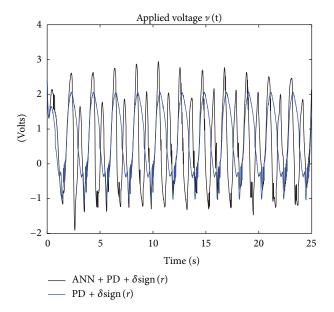$$R = 0.2I_4, \qquad F = 0.3I_{10}. \tag{73}$$

FIGURE 11: Experiments with the one degree-of-freedom robot: applied voltage $v(t)$ for the PD + δ sign (r) and the ANN + PD + δ sign (r) controllers.
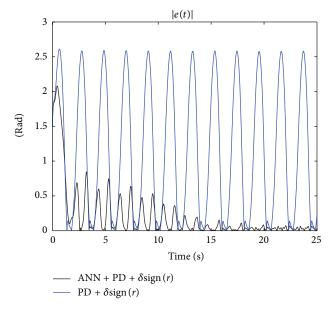

FIGURE 12: Experiments with the one degree-of-freedom robot: absolute value of tracking error $|e(t)|$ for the PD + δ sign (r) and the ANN + PD + δ sign (r) controllers.

The proposed adaptive NN controller ANN + PD + δ sign (r) was implemented by using $L = 10$ neurons.

The results of the implementation of the controllers PD + δ sign (r) in (69) and ANN + PD + δ sign (r) in (70) are given in Figures 11 and 12, which show the time evolution of the applied voltage $v(t)$ and the absolute value of tracking error $|e(t)|$, respectively. The advantage of the proposed controller in ANN + PD + δ sign (r) is clearly observed from Figure 12. Notice that with the new controller (70), which
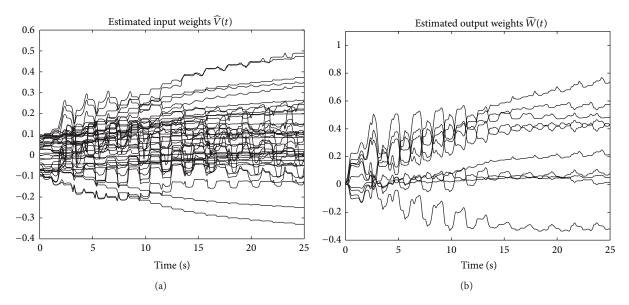
FIGURE 13: Experiments with the one degree-of-freedom robot: estimated input and output weights for the ANN + PD + δ sign (r) controller.

includes the adaptive neural network, the absolute value of the tracking error $|e(t)|$ is reduced to values less than 0.08 [rad] after 20 [sec]. Finally, Figure 13 depicts the estimated input and output weights, $\widehat{V}(t)$ and $\widehat{W}(t)$, respectively, which remain bounded for all time.

5. Conclusions

This paper introduced a new adaptive neural network control algorithm for the tracking control of robot manipulators. The neural network of this controller uses only the desired joint position, velocity, and acceleration. The main theoretical result consisted in showing that the position and velocity error converge asymptotically while the input and output weights of the neural network remain bounded.

The experimental tests showed the benefit of using an adaptive neural network plus nonlinear PD control action. The new controller showed robustness to model uncertainties and strong nonlinear effects such as gravitational effects and Coulomb friction.

Further research considers the application of the proposed ideas in PI joint velocity control [35], anticontrol of chaos [36], output feedback tracking control [37], and underactuated systems [38].

Conflict of Interests

The authors declares that there is no conflict of interests regarding the publication of this paper.
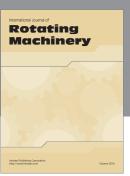
Acknowledgments

References

[1] F. L. Lewis, J. Campos, and R. Selmic, *Neuro-Fuzzy Control of Industrial Systems with Actuator Non-Linearities*, SIAM, Philadelphia, Pa, USA, 2002.

[2] R. R. Selmic and F. L. Lewis, "Backlash compensation in nonlinear systems using dynamic inversion by neural networks," *Asian Journal of Control*, vol. 2, no. 2, pp. 76–87, 2000.

[3] C. Kwan, D. M. Dawson, and F. L. Lewis, "Robust adaptive control of robots using neural network: global stability," *Asian Journal of Control*, vol. 3, no. 2, pp. 111–121, 2001.

[4] I.-C. Ha and M.-C. Han, "A robust control with a neural network structure for uncertain robot manipulator," *KSME International Journal*, vol. 18, no. 11, pp. 1916–1922, 2004.

[5] W. Yu and X. Li, "PD control of robot with velocity estimation and uncertainties compensation," *International Journal of Robotics and Automation*, vol. 21, no. 1, pp. 1–9, 2006.

[6] L. Wang, T. Chai, and Z. Fang, "Neural-network-based two-loop control of robotic manipulators including actuator dynamics in task space," *Journal of Control Theory and Applications*, vol. 7, no. 2, pp. 112–118, 2009.

[7] S. N. Huang, K. K. Tan, and T. H. Lee, "Adaptive neural network algorithm for control design of rigid-link electrically driven robots," *Neurocomputing*, vol. 71, no. 4–6, pp. 885–894, 2008.

[8] M. A. Moreno-Armendáriz, C. A. Pérez-Olvera, F. O. Rodríguez, and E. Rubio, "Indirect hierarchical FCMAC control for the ball and plate system," *Neurocomputing*, vol. 73, no. 13, pp. 2454–2463, 2010.

[9] M. M. Fateh and S. S. Alavi, "Impedance control of an active suspension system," *Mechatronics*, vol. 19, no. 1, pp. 134–140, 2009.

[10] H.-S. Liu, S.-Q. Zhu, and Z.-W. Chen, "Saturated output feedback tracking control for robot manipulators via fuzzy self-tuning," *Journal of Zhejiang University C*, vol. 11, no. 12, pp. 956–966, 2010.

[11] H. Liu, K. Hao, and X. Lai, "Fuzzy saturated output feedback tracking control for robot manipulators: a singular perturbation

theory based approach," *International Journal of Advanced Robotic Systems*, vol. 8, no. 4, pp. 43–53, 2011.

[12] H.-X. Li and C. L. P. Chen, "The equivalence between fuzzy logic systems and feedforward neural networks," *IEEE Transactions on Neural Networks*, vol. 11, no. 2, pp. 356–365, 2000.

[13] T. Sun, H. Pei, Y. Pan, H. Zhou, and C. Zhang, "Neural network-based sliding mode adaptive control for robot manipulators," *Neurocomputing*, vol. 74, no. 14-15, pp. 2377–2384, 2011.

[14] D. Hernández, W. Yu, and M. A. Moreno-Armendáriz, "Neural PD control with second-order sliding mode compensation for robot manipulators," in *Proceedings of the International Joint Conference on Neural Network (IJCNN '11)*, pp. 2395–2402, San Jose, Calif, USA, August 2011.

[15] A. Izadbakhsh, A. A. Kalat, M. M. Fateh, and M. R. Rafiei, "A robust anti-windup control design for electrically driven robots—theory and experiment," *International Journal of Control, Automation and Systems*, vol. 9, no. 5, pp. 1005–1012, 2011.

[16] L. Wang, T. Chai, and C. Yang, "Neural-network-based contouring control for robotic manipulators in operational space," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 4, pp. 1073–1080, 2011.

[17] J. J. Rubio, "Modified optimal control with a backpropagation network for robotic arms," *IET Control Theory and Applications*, vol. 6, no. 14, pp. 2216–2225, 2012.

[18] J. H. Pérez-Cruz, J. J. Rubio, E. Ruiz-Velázquez, and G. Solıs-Perales, "Tracking control based on recurrent neural networks for nonlinear systems with multiple inputs and unknown deadzone," *Abstract and Applied Analysis*, vol. 2012, Article ID 471281, 18 pages, 2012.

[19] J. H. Pérez Cruz, E. Ruiz Velázquez, J. J. Rubio, and C. A. de Alba Padilla, "Robust adaptive neurocontrol of siso nonlinear systems preceded by unknow deadzone," *Mathematical Problems in Engineering*, vol. 2012, Article ID 342739, 23 pages, 2012.

[20] V. Santibáñez and R. Kelly, "Global asymptotic stability of bounded output feedback tracking control for robot manipulators," in *Proceedings of the 40th IEEE Conference on Decision and Control*, vol. 2, pp. 1378–1379, 1999.

[21] V. Santibáñez and R. Kelly, "PD control with feedforward compensation for robot manipulators: analysisysis and experimentation," *Robotica*, vol. 19, no. 1, pp. 11–19, 2001.

[22] S. Puga-Guzmán, J. Moreno-Valenzuela, and V. Santibáñez, "On robot motion control via adaptive neural networks," in *Proceedings of the 9th International Conference on Electrical Engineering, Computing Science and Automatic Control*, pp. 1–6, 2012.

[23] C. Meyer, *Matrix Analysis and Applied Linear Algebra*, SIAM, Philadelphia, Pa, USA, 2000.

[24] F. Zhang, D. M. Dawson, M. S. De Queiroz, and W. E. Dixon, "Global adaptive output feedback tracking control of robot manipulators," *IEEE Transactions on Automatic Control*, vol. 45, no. 6, pp. 1203–1208, 2000.

[25] J. Moreno-Valenzuela, V. Santibáñez, and R. Campa, "On output feedback tracking control of robot manipulators with bounded torque input," *International Journal of Control, Automation and Systems*, vol. 6, no. 1, pp. 76–85, 2008.

[26] L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators*, Springer, London, UK, 2000.

[27] R. Kelly, V. Santibáñez, and A. Loría, *Control of Robot Manipulators in Joint Space*, Springer, London, UK, 2005.

[28] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, New York, NY, USA, 1993.
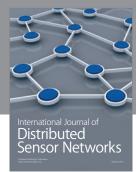
[29] S. Arimoto, "Fundamental problems of robot control: part I, innovations in the realm of robot servo-loops," *Robotica*, vol. 13, no. 1, pp. 19–27, 1995.

[30] S. Arimoto, "Fundamental problems of robot control: part II a nonlinear circuit theory towards an understanding of dexterous motions," *Robotica*, vol. 13, no. 2, pp. 111–122, 1995.

[31] H. Khalil, *Nonlinear Systems*, Prentice Hall, Upper Saddle River, NJ, USA, 2002.

[32] P. Logothetis and J. Kieffer, "On the identification of the robot dynamics without acceleration measurements," Internal Report, Faculty of Engineering and Information Technology, Australian National University, 1996, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.55.8716.

[33] M. Gautier and P. Poignet, "Extended Kalman filtering and weighted least squares dynamic identification of robot," *Control Engineering Practice*, vol. 9, no. 12, pp. 1361–1372, 2001.

[34] D. M. Dawson, J. Hu, and T. Burg, *Nonlinear Control of Electrical Maquinery*, Marcel Dekker, New York, NY, USA, 1998.

[35] J. Moreno-Valenzuela and V. Santibáñez, "Robust saturated PI joint velocity control for robot manipulators," *Asian Journal of Control*, vol. 15, no. 1, pp. 64–79, 2013.

[36] J. Moreno-Valenzuela, "Adaptive anti control of chaos for robot manipulators with experimental evaluations," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 1, pp. 1–11, 2013.

[37] A. Yarza, V. Santibáñez, and J. Moreno-Valenzuela, "An adaptive output feedback motion tracking controller for robot manipulators: uniform global asymptotic stability and experimentation," *International Journal of Applied Mathematics and Computer Science*, vol. 23, no. 3, pp. 599–611, 2013.

[38] O. García-Alarcón, S. Puga-Guzmán, and J. Moreno-Valenzuela, "On parameter identification of the Furuta pendulum," *Procedia Engineering*, vol. 35, pp. 77–84, 2012.

Journal of
Engineering

The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

International Journal of
Distributed
Sensor Networks

Advances in
Civil Engineering

Journal of
Control Science
and Engineering

Journal of
Robotics

Journal of
Electrical and Computer
Engineering

Advances in
OptoElectronics

VLSI Design

International Journal of
Navigation and
Observation

Modelling &
Simulation
in Engineering

International Journal of
Aerospace
Engineering

Hindawi

Submit your manuscripts at
http://www.hindawi.com

International Journal of
Chemical Engineering

International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components

Shock and Vibration

Advances in
Acoustics and Vibration