Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 451049, 12 pages
http://dx.doi.org/10.1155/2015/451049

Research Article

Decentralized Identification and Control in Real-Time of a Robot Manipulator via Recurrent Wavelet First-Order Neural Network

Luis A. Vázquez,[1] Francisco Jurado,[1] and Alma Y. Alanís[2]

[1] *Tecnológico Nacional de México, Instituto Tecnológico de La Laguna, Boulevard Revolución y Calzada Cuauhtémoc, 27000 Torreón, COAH, Mexico*
[2] *Centro Universitario de Ciencias Exactas e Ingenierías (CUCEI), Universidad de Guadalajara, Boulevard Marcelino García Barragán No. 1421, 44430 Guadalajara, JAL, Mexico*

Correspondence should be addressed to Luis A. Vázquez; lvazquez@itlalaguna.edu.mx

Received 27 October 2014; Revised 7 March 2015; Accepted 10 March 2015

Academic Editor: Fernando Torres

A decentralized recurrent wavelet first-order neural network (RWFONN) structure is presented. The use of a wavelet Morlet activation function allows proposing a neural structure in continuous time of a single layer and a single neuron in order to identify online in a series-parallel configuration, using the filtered error (FE) training algorithm, the dynamics behavior of each joint for a two-degree-of-freedom (DOF) vertical robot manipulator, whose parameters such as friction and inertia are unknown. Based on the RWFONN subsystem, a decentralized neural controller is designed via backstepping approach. The performance of the decentralized wavelet neural controller is validated via real-time results.

1. Introduction

The control of robotic manipulators is a mature yet fruitful field for research, development, and manufacturing. The industrial robots are basically positioning systems; therefore, the trajectory tracking control problem for this kind of robot manipulators has been a challenging problem to be solved in the last years. In this respect, the field of artificial neural networks (ANNs) has been playing an interesting role and poses a great challenge to its implementation in real-time. Some works have been dealing with the design of neural control schemes in discrete time where the extended Kalman filtering has been used as training algorithm for both recurrent high-order neural networks (RHONNs) [1–3] and high-order neural networks (HONNs) [4, 5]. The neural control schemes in continuous time have been proposed using the filtered error (FE) algorithm as training law for a RHONN [6–8]. In all of the works referenced above, the identification and control were carried out online at the same time.

The wavelet neural networks (WNNs) structure arises from combining wavelets concept with the ANNs approach in order to achieve a better identification performance [9–13]. Unlike sigmoidal activation functions used in conventional ANNs, these are replaced by a wavelet activation function in the WNN. WNNs have proved to be better than ANNs in the sense that their structure can provide more potential to enrich the mapping relationship between inputs and outputs [9]. WNNs simultaneously possess the advantages of ANN learning ability and wavelet decomposition ability. WNNs-based control systems have been adopted widely for control of complex dynamical systems owing to their fast learning properties and good generalization capability [10, 11]. Moreover, recurrent wavelet neural networks (RWNNs), which combine properties such as dynamic response of recurrent neural networks (RNNs) and the fast convergence of a WNN, have been proposed to identify and control nonlinear systems [14–18]. An intelligent control system using recurrent wavelet-based Elman neural network for position control of a permanent-magnet synchronous motor servo drive was proposed in [19]. In [20, 21], a self-recurrent wavelet neural network structure was proposed in order to identify a synchronous generator and the nonlinearities introduced in the system due to actuator saturation. In [22],

two types of Haar wavelet neural network, feedforward and RWNN, were used to model discrete-time nonlinear systems. Some researches have been carried out in real-time with interesting results, such as in [23], where an adaptive RWNN uncertainty observer was proposed to estimate the required lumped uncertainty and the backstepping approach was used to control the motion of a linear synchronous motor drive system. A novel global PID control scheme for nonlinear MIMO systems was proposed and synthesized for a robot manipulator in [24]. Inverse dynamics identification was based on a radial basis neural network with daughter RASP1 wavelets activation functions in cascade with an infinite impulse response filter in the output to prune irrelevant signals and nodes. In [25], an intelligent control system based on a four-layer RWNN, trained via gradient descent method and which includes Gaussian wavelet functions, and using the sliding mode approach was proposed to achieve trajectory tracking for an UAV.

The use of WNNs and RWNNs is not a new theme in the literature; however, the implementation of these structures involves the use of multiple layers with multiple neurons. The use of offline training algorithms and a little bit of applications in real-time have generated a major motivation in the development of this work. Signal and images decomposition using a space of basis functions having local support in both the time and frequency domains has proven their usefulness in signal and images processing. Such motivations across various fields have led to the development of wavelets. Some view wavelets as a new basis for representing functions. Where the Fourier series maps a one-dimensional function of a continuous variable into a one-dimensional sequence of coefficients, the wavelet expansion maps it into a two-dimensional array of coefficients that allows localizing the signal in both time and frequency simultaneously. Wavelet algorithms are defined to process data at different scales of resolution in both the time and frequency domains and are useful tools for multichannel signal processing, for example, estimation or filtering, and multiresolution signal analysis (MRA). The MRA shows how orthonormal wavelet bases can be used as a tool to describe mathematically the increment of information needed to go from a coarse approximation to a higher resolution of approximation. For all of the above, we propose a continuous-time decentralized wavelet neural identification and control scheme based on the structure of a RHONN model where sigmoidal activation functions and high-order connections between them are replaced by a wavelet (Morlet) activation function. The resulting structure has been called recurrent wavelet first-order neural network (RWFONN), from which the design of the controller is carried out using the backstepping approach. The training for the RWFONN is performed online in a series-parallel configuration using the FE algorithm, shaping a new neural structure of a single layer and a single neuron, where its capacity for identification of more complex dynamics increases with the implementation of a wavelet Morlet. The performance of the proposed scheme for trajectory tracking is validated via experimental results when it is applied to a vertical robot manipulator of two DOF with unknown parameters.

This paper is organized as follows. The RWFONN model, its approximation properties, and the FE algorithm are described in Section 2. Section 3 shows the RWFONN model proposed as well as the decentralized neural controller design. In Section 4 are presented real-time results, and final comments conclude the paper.

2. Recurrent Wavelet First-Order Neural Network Model

Consider the RHONN model [26] where the state of each neuron is governed by a differential equation of the form

$$\dot{x}_j^i = -a_j^i x_j^i + b_j^i \sum_{k=1}^{L} w_{jk}^i \prod_{j\in I_k} y_j^{d_j(k)}, \tag{1}$$

where x_j^i is the state of the ith neuron, a_j^i and b_j^i are positive real constants, w_{jk}^i is the kth adjustable synaptic weight connecting the jth state to the ith neuron, L represents the total number of weights used to identify the plant behavior, and y_j is the activation function for each one of the connections. Each y_j is either an external input or the state of a neuron through a sigmoidal function; that is, $y_j = s(x_j^i)$, where $s(\cdot)$ is a sigmoidal nonlinearity. In a recurrent second-order neural network, the total input to the neuron not only is a linear combination of the components y_j, but also may be the product of two or more elements represented by triplets $y_1 y_2 y_3$, quadruplets, and so forth. $\{I_1, I_2, \ldots, I_L\}$ is a collection of L nonordered subsets of $\{1, 2, \ldots, i+j\}$ and $d_j(k)$ are nonnegative integers. This class of neural networks form a RHONN.

The input vector for each neuron is given by

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_j \\ y_{j+1} \\ \vdots \\ y_{i+j} \end{pmatrix} = \begin{pmatrix} s\left(x^1\right) \\ \vdots \\ s\left(x^i\right) \\ u^1 \\ \vdots \\ u^i \end{pmatrix}, \tag{2}$$

where $\mathbf{u} = \begin{bmatrix} u^1 & u^2 & \cdots & u^i \end{bmatrix}^T$ is the vector of external control inputs to the network.

Introducing a L-dimensional vector $\mathbf{z}$, defined as

$$\mathbf{z} = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_L \end{pmatrix} = \begin{pmatrix} \prod_{j\in I_1} y_j^{d_j(1)} \\ \prod_{j\in I_2} y_j^{d_j(2)} \\ \vdots \\ \prod_{j\in I_L} y_j^{d_j(L)} \end{pmatrix}, \tag{3}$$

the RHONN model (1) can be rewritten as

$$\dot{\mathbf{x}}_j^i = -a_j^i \mathbf{x}_j^i + b_j^i \sum_{k=1}^{L} \mathbf{w}_{jk}^i \mathbf{z}_{jk}^i. \tag{4}$$

Replacing now the vector $\mathbf{z}$ by a wavelet vector $\boldsymbol{\psi}$, considering that higher-order terms y_j will not be used, the RHONN model (1) can be rewritten as

$$\dot{\mathbf{x}}_j^i = -a_j^i \mathbf{x}_j^i + b_j^i \sum_{k=1}^{L} \mathbf{w}_{jk}^i \boldsymbol{\psi}_{jk}^i. \tag{5}$$

The adjustable parameter vector is defined by $\mathbf{w}_{jk}^i = b_j^i \left[w_{j1}^i \;\; w_{j2}^i \;\; \cdots \;\; w_{jL}^i \right]^T$, so (5) becomes

$$\dot{\mathbf{x}}_j^i = -a_j^i \mathbf{x}_j^i + \left(\mathbf{w}_{jk}^i\right)^T \boldsymbol{\psi}_{jk}^i, \tag{6}$$

where the vectors $\mathbf{w}_{jk}^i$, for $k = 1, 2, \ldots, L$, represent the adjustable weights of the network, while the coefficients a_j^i, for $i = 1, 2, \ldots, n$, are part of the underlying network architecture and are fixed during training. The structure in form (6) here is called RWFONN.

This RWFONN, that is, a neuron with a single connection of first-order, is the most simple structure for a RHONN and is an expansion of the first-order Hopfield [27] and Cohen-Grossberg [28] models. The sigmoidal activation function $s(\cdot)$ is replaced by the real version of the modified wavelet Morlet [29, 30] of form $\psi(x) = e^{-x^2/\beta} \cos(\lambda x)$, with parameters β and λ representing expansion and dilation, respectively. Properties about wavelet functions are described in [31–35].

2.1. Approximation Properties of the RWFONN. In the following, the problem of approximating a general nonlinear dynamical system by a RWFONN is described. The input-output behavior of the system to be approximated is given by

$$\dot{\boldsymbol{\chi}}_j^i = F\left(\boldsymbol{\chi}_j^i, u^i\right), \tag{7}$$

where $u^i \in \mathscr{R}^i$ is the input to the system, $\boldsymbol{\chi}_j^i \in \mathscr{R}^j$ is the state of the system, and $F : \mathscr{R}^{i+j} \mapsto \mathscr{R}^j$ is a smooth vector field defined on a compact set $\mathscr{D} \subset \mathscr{R}^{i+j}$, where i and j are constants. The approximation problem consists in determining and using wavelet activation function for continuous time, if there exist weights $\mathbf{w}_{jk}^i$ such that (6) approximates the input-output behavior of an arbitrary dynamical system of form (7). Assume that $F(\cdot)$ is continuous and satisfies a local Lipschitz condition such that (7) has a unique solution and $(\boldsymbol{\chi}_j^i(t), u^i(t)) \in \mathscr{D}$ for all t in some time interval $J_T = \{t : 0 \le t \le T\}$, where J_T represents the time period over which the approximation is performed. Based on the above assumptions, the next theorem, which is strictly an existence result and does not provide any constructive method in order to obtain the correct weights $\mathbf{w}^{*i}_{jk}$, proves that if a sufficiently large number of weights are allowed in (6), then it is possible to approximate any dynamical system to any degree of accuracy.

Theorem 1. *Suppose that system (7) and the RWFONN model (6) are initially at the same state $\boldsymbol{\chi}_j^i(0) = \mathbf{x}_j^i(0)$. Then, for any $\varepsilon > 0$ and any finite $T > 0$, there exists an integer L and a vector $\mathbf{w}^{*i}_{jk} \in \mathscr{R}^L$ such that the state $\mathbf{x}_j^i(t)$ of the RWFONN model (6), with L number of weights whose values $\mathbf{w}_{jk}^i = \mathbf{w}^{*i}_{jk}$, satisfies*

$$\sup_{0 \le t \le T} \left| \mathbf{x}_j^i(t) - \boldsymbol{\chi}_j^i(t) \right| \le \varepsilon. \tag{8}$$

Proof. The proof proceeds along the same lines as the proof in [36]. The dynamic behavior of the RWFONN model is described by

$$\dot{\mathbf{x}}_j^i = -a_j^i \mathbf{x}_j^i + \left(\mathbf{w}_{jk}^i\right)^T \boldsymbol{\psi}_{jk}^i\left(x_j^i, u^i\right). \tag{9}$$

Assuming that each a_j^i is positive, then the bounded-input bounded-output (BIBO) stability for each neuron x_j^i is guaranteed. By adding and subtracting $-a_j^i \boldsymbol{\chi}_j^i$, system (7) can be rewritten as

$$\dot{\boldsymbol{\chi}}_j^i = -a_j^i \boldsymbol{\chi}_j^i + G\left(\boldsymbol{\chi}_j^i, u^i\right), \tag{10}$$

where $G(\boldsymbol{\chi}_j^i, u^i) = F(\boldsymbol{\chi}_j^i, u^i) + a_j^i \boldsymbol{\chi}_j^i$. Since $\mathbf{x}_j^i(0) = \boldsymbol{\chi}_j^i(0)$, the identification error $\boldsymbol{\xi}_j^i = \mathbf{x}_j^i - \boldsymbol{\chi}_j^i$ satisfies

$$\dot{\boldsymbol{\xi}}_j^i = -a_j^i \boldsymbol{\xi}_j^i + \left(\mathbf{w}_{jk}^i\right)^T \boldsymbol{\psi}_{jk}^i\left(x_j^i, u^i\right) - G\left(\boldsymbol{\chi}_j^i, u^i\right) \tag{11}$$

with $\boldsymbol{\xi}_j^i(0) = 0$.

By assumption, $(\boldsymbol{\chi}_j^i(t), u^i(t)) \in \mathscr{D}$ $\forall t \in [0, T]$, where $\mathscr{D}$ is a compact subset of $\mathscr{R}^{i+j}$.

Let

$$\mathscr{D}_\varepsilon = \left\{ \left(\boldsymbol{\chi}_j^i, u^i\right) \in \mathscr{R}^{i+j} : \left| \left(\boldsymbol{\chi}_j^i, u^i\right) - \left(\boldsymbol{\chi}_{jy}^i, u_y^i\right) \right| \le \varepsilon, \; \left(\boldsymbol{\chi}_{jy}^i, u_y^i\right) \in \mathscr{D} \right\}. \tag{12}$$

It can be seen that $\mathscr{D}_\varepsilon$ is also a compact subset of $\mathscr{R}^{i+j}$ and $\mathscr{D} \subset \mathscr{D}_\varepsilon$, where ε is the required degree of approximation. Since $\boldsymbol{\psi}_{jk}^i$ is a continuous function, it satisfies a Lipschitz condition

$$\left| \boldsymbol{\psi}_{jk}^i\left(\mathbf{x}_{j1}^i, u^i\right) - \boldsymbol{\psi}_{jk}^i\left(\mathbf{x}_{j2}^i, u^i\right) \right| \le l \left| \mathbf{x}_{j1}^i - \mathbf{x}_{j2}^i \right| \tag{13}$$

in the compact domain $\mathscr{D}_\varepsilon$.

In what follows, it is shown that the function $(\mathbf{w}_{jk}^i)^T \boldsymbol{\psi}_{jk}^i(x_j^i, u^i)$ satisfies the conditions of the Stone-Weierstrass theorem and it can therefore approximate, over a compact domain, any continuous function. The preprocessing of the input via a continuous invertible function does not affect the ability of a network to approximate continuous functions. Therefore, it can be shown that if L is large enough, that is, the number of weights is large enough, then there exist weight values $\mathbf{w}_{jk}^i = \mathbf{w}^{*i}_{jk}$ such that $(\mathbf{w}^{*i}_{jk})^T \boldsymbol{\psi}_{jk}^i(\mathbf{x}_j^i, u^i)$ can approximate $G(\boldsymbol{\chi}_j^i, u^i)$, in

a compact domain, to any degree of accuracy for all $(\mathbf{x}_j^i, u^i)$. Hence, there exists $\mathbf{w}_{jk}^i = \mathbf{w}^{*i}_{jk}$ such that

$$\sup_{(\chi_j^i, u^i)\in\mathscr{D}_\varepsilon} \left| \left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\chi_j^i, u^i\right) - G\left(\chi_j^i, u^i\right)\right| \le \eta, \tag{14}$$

where η is a constant.

The solution to the differential equation (11) can be expressed as

$$\begin{aligned}\xi_j^i &= \int_0^t \xi_j^{i\,-a_j^i(t-\tau)} \left[\left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\mathbf{x}_j^i(\tau), u^i(\tau)\right) - G\left(\chi_j^i(\tau), u^i(\tau)\right)\right] d\tau \\ &= \int_0^t \xi_j^{i\,-a_j^i(t-\tau)} \left[\left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\mathbf{x}_j^i(\tau), u^i(\tau)\right) - \left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\chi_j^i(\tau), u^i(\tau)\right)\right] d\tau \\ &\quad + \int_0^t \xi_j^{i\,-a_j^i(t-\tau)} \left[\left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\chi_j^i(\tau), u^i(\tau)\right) - G\left(\chi_j^i(\tau), u^i(\tau)\right)\right] d\tau.\end{aligned} \tag{15}$$

Since each a_j^i is a positive real constant, there exists a positive constant α such that $\|\xi_j^{i\,-a_j^i t}\| \le \xi_j^{i\,-\alpha t}$ and $\alpha < L$, where $L = l\|\mathbf{w}^{*i}_{jk}\|$. Based on all of the above, let η be chosen as

$$\eta = \frac{(L-\alpha)\,\varepsilon}{2\left(\xi_j^{i\,(L-\alpha)T} - 1\right)} > 0. \tag{16}$$

Now, consider that $(\mathbf{x}_j^i(t), u^i(t)) \in \mathscr{D}_\varepsilon \quad \forall t \in [0, T]$. From (15), taking norms on both sides and using (13), (14), and (16), the following inequalities hold for $t \in [0, T]$:

$$\begin{aligned}\left|\xi_j^i(t)\right| &\le \int_0^t \left\|\xi_j^{i\,-a_j^i(t-\tau)}\right\| \cdot \left|\left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\mathbf{x}_j^i(\tau), u^i(\tau)\right) - \left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\chi_j^i(\tau), u^i(\tau)\right)\right| d\tau \\ &\quad + \int_0^t \left\|\xi_j^{i\,-a_j^i(t-\tau)}\right\| \cdot \left|\left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\chi_j^i(\tau), u^i(\tau)\right) - G\left(\chi_j^i(\tau), u^i(\tau)\right)\right| d\tau \\ &\le \int_0^t \xi_j^{i\,-\alpha(t-\tau)} L \left|\xi_j^i(\tau)\right| d\tau + \int_0^t \delta \xi_j^{i\,-\alpha(t-\tau)} d\tau.\end{aligned} \tag{17}$$

Using the Bellman-Gronwall lemma, it can be easily shown that

$$\begin{aligned}\left|\xi_j^i(t)\right| &\le \frac{\eta}{L-\alpha}\left(\xi_j^{i\,(L-\alpha)t} - 1\right), \\ \left|\xi_j^i(t)\right| &\le \frac{\varepsilon}{2}.\end{aligned} \tag{18}$$

Suppose that $(\mathbf{x}_j^i, u^i)$ does not belong to $\mathscr{D}_\varepsilon \quad \forall t \in [0, T]$. Then, by the continuity of $\mathbf{x}_j^i(t)$, there exists a T^*, where $0 \le T^* \le T$, such that $(\mathbf{x}_j^i(T^*), u^i(T^*)) \in \partial\mathscr{D}_\varepsilon$ where $\partial\mathscr{D}_\varepsilon$ denotes the boundary of $\mathscr{D}_\varepsilon$. Furthermore, carrying out the same analysis for $t \in [0, T^*]$ it results $|\mathbf{x}_j^i(t) - \chi_j^i(t)| \le \varepsilon/2$. Hence, (18) holds for all $t \in [0, T]$. □

2.2. Filtered Error Training Algorithm. Under the assumption that the unknown system is exactly modeled by a RWFONN architecture of form (6), the weight adjustment law and the FE training algorithm for this RWFONN are next summarized. Based on the assumptions of no modeling error, there exist unknown weight vectors $\mathbf{w}^{*i}_{jk}$ such that each state χ_j^i of the unknown dynamical system (7) satisfies

$$\dot{\chi}_j^i = -a_j^i \chi_j^i + \left(\mathbf{w}^{*i}_{jk}\right)^T \psi_{jk}^i\left(\chi_j^i, u^i\right) \quad \chi_j^i(0) = \chi_{j0}^i, \tag{19}$$

where χ_{j0}^i is the initial state of the system. As is standard in systems identification procedures, here it is assumed that the input $u^i(t)$ and the state $\chi_j^i(t)$ remain bounded for all $t \ge 0$. Based on the definition for $\psi_{jk}^i(\chi_j^i, u^i)$ given by (3), this implies that $\psi_{jk}^i(\chi_j^i, u^i)$ is also bounded. In the sequel, unless there exists confusion, the arguments of the vector field ψ will be omitted. Next, the approach for estimating the unknown parameters $\mathbf{w}^{*i}_{jk}$ of the RWFONN model (19) is described.

Considering (19) as the differential equation describing the dynamics of the unknown system, the identifier structure is chosen with the same form as in (6), where $\mathbf{w}_{jk}^i$ is the estimate of the unknown weight vector $\mathbf{w}^{*i}_{jk}$. From (6) and (19), the identification error $\xi_j^i = \mathbf{x}_j^i - \chi_j^i$ satisfies

$$\begin{aligned}\dot{\xi}_j^i &= \dot{\mathbf{x}}_j^i - \dot{\chi}_j^i \\ &= -a_j^i\left(\mathbf{x}_j^i - \chi_j^i\right) + \left(\left(\mathbf{w}_{jk}^i\right)^T - \left(\mathbf{w}^{*i}_{jk}\right)^T\right)\psi_{jk}^i,\end{aligned} \tag{20}$$

which can be rewritten as

$$\dot{\xi}_j^i = -a_j^i \xi_j^i + \left(\tilde{\mathbf{w}}_{jk}^i\right)^T \psi_{jk}^i, \tag{21}$$

where $(\tilde{\mathbf{w}}_{jk}^i)^T = (\mathbf{w}_{jk}^i)^T - (\mathbf{w}^{*i}_{jk})^T$ denotes the parametric error [35]. The weights $\mathbf{w}_{jk}^i$ are adjusted according to the learning law

$$\dot{\mathbf{w}}_{jk}^i = -\Gamma_{jk}^i \psi_{jk}^i \xi_j^i, \tag{22}$$

where the adaptive gain $\Gamma_{jk}^i \in \mathscr{R}^{L\times L}$ is a positive definite matrix. Stability and convergence properties for the weight

adjustment law given above are analyzed in [37]. The following theorem establishes that this identification scheme has convergence properties with the gradient method for adjusting the weights.

Theorem 2 (see [36]). *Consider the filtered error RWFONN model given by (21) whose weights are adjusted according to (22). Then,*

(1) $\boldsymbol{\xi}_j^i, \widetilde{\mathbf{w}}_{jk}^i \in \mathscr{L}_\infty$ *(i.e.,* $\boldsymbol{\xi}_j^i$ *and* $\widetilde{\mathbf{w}}_{jk}^i$ *are uniformly bounded);*

(2) $\lim_{t\to\infty}\boldsymbol{\xi}_j^i(t) = 0$.

3. Neural Backstepping Controller Design

Consider now the mathematical model of an i–DOF robot manipulator given by

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\dot{\mathbf{q}}), \tag{23}$$

where $\boldsymbol{\tau}$ is a $i \times 1$ input vector that represents the torques applied to each joint; $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}} \in \mathscr{R}^j$ are the states of the system corresponding to position, velocity, and acceleration for each joint, respectively; $\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} \in \mathscr{R}^{j\times i}$ represents the contribution of the inertial forces to the dynamical equation; hence the matrix $\mathbf{M}$ represents the inertia matrix of the manipulator; $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} \in \mathscr{R}^j$ represents the Coriolis forces, $\mathbf{g}(\mathbf{q}) \in \mathscr{R}^j$ represents the gravitational forces, and $\mathbf{f}(\dot{\mathbf{q}}) \in \mathscr{R}^j$ is a vector that combines both viscous and Coulomb friction terms, that is, the so-called friction vector [38]. In this work, as a continuity of [35], a decentralized RWFONN trained via FE algorithm (21) with weights adjustment law (22) is proposed for identification and control of a two-DOF vertical robot manipulator. From (6), the decentralized RWFONN model is given as follows:

$$\begin{aligned}\dot{x}_1^i &= -a_1^i x_1^i + w_{11}^i \psi_{11}^i\left(q^i\right) + x_2^i,\\ \dot{x}_2^i &= -a_2^i x_2^i + w_{21}^i \psi_{21}^i\left(\dot{q}^i\right) + u^i,\end{aligned} \tag{24}$$

with $k = 1$ for the $w_{jk}^i\psi_{jk}^i(\cdot)$ term, where $i = 1, 2$ denotes the number of the joints; $j = 1, 2$ is for the number of states of the ith RWFONN model; q^i represents the measurable local angular position and $\dot{q}^i$ is for the calculated local angular velocity; u^i is the control input. It must be noticed that this decentralized neural scheme is in the form of a strict-feedback system as that proposed in [8]; then, the use of the backstepping approach is still being suitable for the design of the neural controller. For each ith joint the identification error between the neural identifier and the joint variable is defined as $\xi_1^i = x_1^i - q^i$ for the angular position and $\xi_2^i = x_2^i - \dot{q}^i$ for the angular velocity. To update online the synaptic weights, the adaptive learning laws are given by $\dot{w}_{11}^i = -\Gamma_{11}^i\psi_{11}^i(q^i)\xi_1^i$ and $\dot{w}_{21}^i = -\Gamma_{21}^i\psi_{21}^i(\dot{q}^i)\xi_2^i$, with $\Gamma_{jk}^i > 0$ as the adaptive gain and

$$\begin{aligned}\psi_{11}^i\left(q^i\right) &= e^{-(q^i)^2/\beta_1}\cos\left(\lambda_1 q^i\right) + e^{-(q^i)^2/\beta_1}\cos\left(\lambda_2 q^i\right),\\ \psi_{21}^i\left(\dot{q}^i\right) &= e^{-(\dot{q}^i)^2/\beta_2}\cos\left(\lambda_3 \dot{q}^i\right),\end{aligned} \tag{25}$$

with $\beta_1 = 2$, $\lambda_1 = 0.1$, and $\lambda_2 = 0.01$ for ψ_{11}^i and $\beta_2 = \lambda_3 = 2$ for ψ_{21}^i.

Next, our objective is to design a feedback control law $u(t)$ to force the system output to follow a desired trajectory. The decentralized neural control scheme is based on the following.

Denoting ξ^i as the identification error and the trajectory tracking error between the states of the neural network for position and the desired trajectory as $\epsilon^i = |x_1^i - q_d^i|$, the output tracking error is rewritten as

$$\widetilde{q}^i = \xi^i + \epsilon^i, \tag{26}$$

where $\widetilde{q}^i = |q^i - q_d^i|$. Consequently, the error dynamics is given as

$$\dot{\widetilde{q}}^i = \dot{\xi}^i + \dot{\epsilon}^i. \tag{27}$$

Considering (21), (24), and

$$\dot{\epsilon}^i = \dot{x}_1^i - \dot{q}_d^i, \tag{28}$$

the error dynamics is then described by

$$\begin{aligned}\dot{\widetilde{q}}^i = &-a_1^i\xi_1^i + \left(\widetilde{w}_{1k}^i\right)\psi_{1k}^i - a_1^i x_1^i\\ &+ w_{11}^i\psi_{11}^i\left(q^i\right) + x_2^i - \dot{q}_d^i.\end{aligned} \tag{29}$$

The decentralized wavelet neural controller design can be formulated as follows.

Theorem 3. *For a neural identifier in a strict-feedback form (24) the dynamics of (26) and the neural network tracking error (28) with control law (41) and positive values for* ζ_1^i *and* ζ_2^i *have an asymptotically stable equilibrium point and the output tracking error (26) tends to zero.*

Proof. Assuming that the system is Lipschitz, we proceed by proposing the augmented Lyapunov-like function

$$V_1^i\left(\epsilon^i, \xi^i, \widetilde{w}^i\right) = V^i\left(\xi^i, \widetilde{w}^i\right) + \frac{1}{2}\left(\epsilon^i\right)^2 \tag{30}$$

with

$$V^i\left(\xi^i, \widetilde{w}^i\right) = \frac{1}{2}\sum_{i=1}^{n}\left(\left(\xi_1^i\right)^2 + \left(\Gamma_{1k}^i\right)^{-1}\left(\widetilde{w}_{1k}^i\right)^2\right) \tag{31}$$

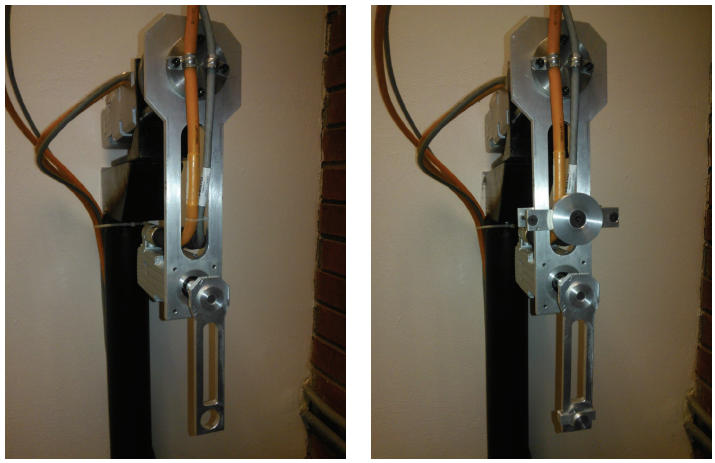
taken from [36].

Considering (21)-(22), the time derivative of (31) is then given by [36]

$$\dot{V}^i(\cdot) = -\sum_{i=1}^{n} a_1^i\left(\xi_1^i\right)^2 \le 0. \tag{32}$$

The time derivative of $V_1^i(\cdot)$ along the solution of (28) is

$$\dot{V}_1^i(\cdot) = \dot{V}^i(\cdot) + \epsilon^i\left[-a_1^i x_1^i + w_{11}^i\psi_{11}^i + x_2^i - \dot{q}_d^i\right]. \tag{33}$$



(a) (b)

Figure 1: Two-DOF vertical robotic arm. From left to right: without payload on each link, with payload in both links.

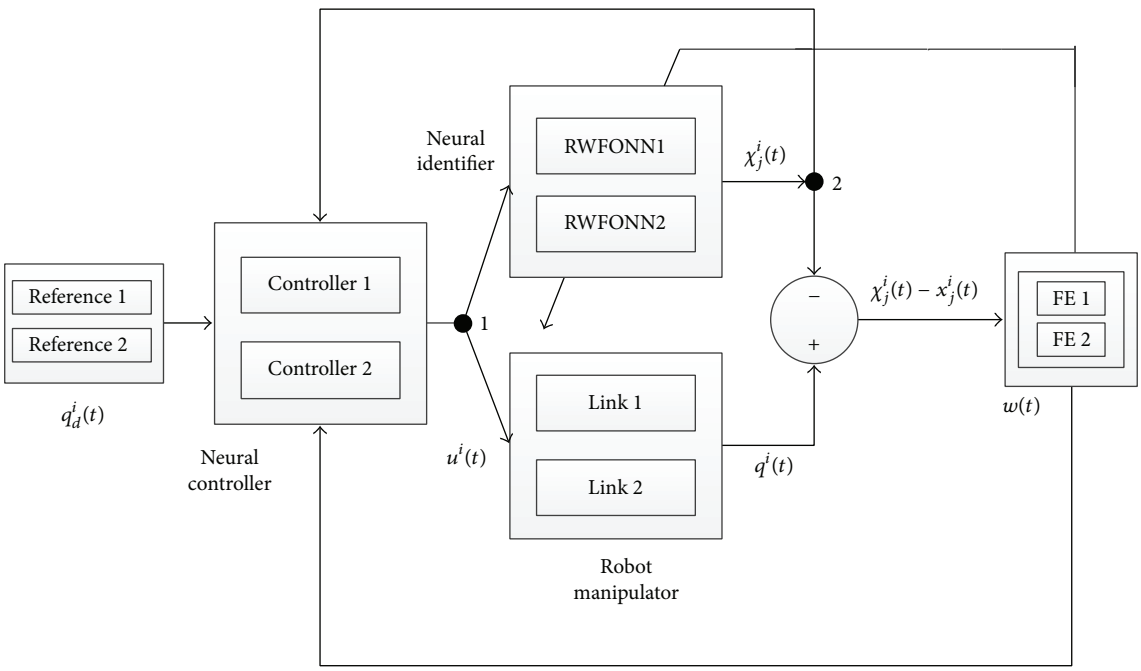


Figure 2: Block diagram for the decentralized neural control scheme proposed.

In order to guarantee that (33) is negative definite, the desired dynamics for x_2^i is proposed as

$$x_{2d}^i = -\zeta_1^i \epsilon^i + a_1^i x_1^i - w_{11}^i \psi_{11}^i + \dot{q}_d^i; \tag{34}$$

thus,

$$\begin{aligned}\dot{V}_1^i(\cdot) &= \dot{V}^i(\cdot) + \epsilon^i \left[-a_1^i x_1^i + w_{11}^i \psi_{11}^i - \zeta_1^i \epsilon^i \right. \\ &\quad \left. + a_1^i x_1^i - w_{11}^i \psi_{11}^i + \dot{q}_d^i - \dot{q}_d^i\right] \\ &= \dot{V}^i(\cdot) - \zeta_1^i \left(\epsilon^i\right)^2 \\ &= -\sum_{i=1}^{n} a_1^i \left(\xi_1^i\right)^2 - \zeta_1^i \left(\epsilon^i\right)^2\end{aligned} \tag{35}$$

with $\zeta_1^i > 0$ a real value.

Now, proposing the augmented Lyapunov-like function

$$V_2^i\left(\epsilon^i, \xi^i, \tilde{w}^i, \tilde{q}_2^i\right) = V^i(\cdot) + V_1^i(\cdot) + \frac{1}{2}\left(\tilde{q}_2^i\right)^2 \tag{36}$$

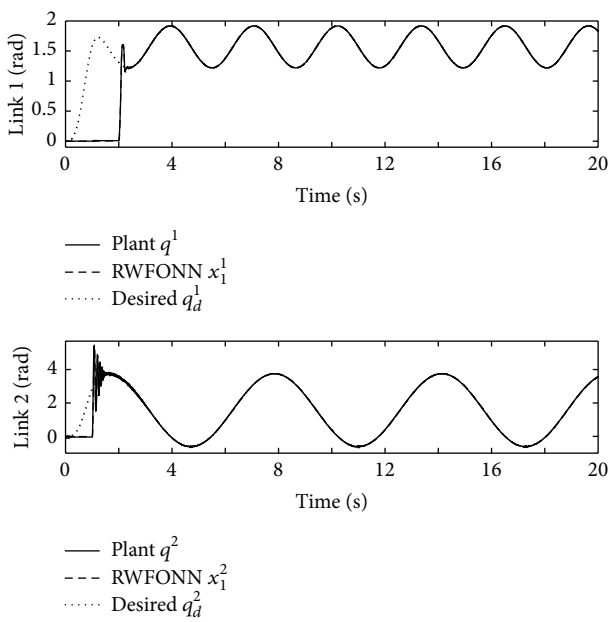


Figure 3: Identification and trajectory tracking performed by the decentralized RWFONN control scheme.

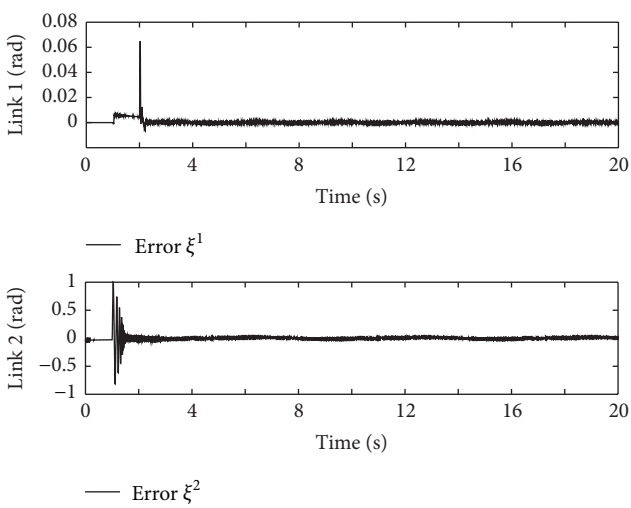


Figure 4: Identification error between robot manipulator and RWFONN.

and introducing the error

$$\tilde{q}_2^i = x_2^i - x_{2d}^i, \tag{37}$$

the time derivative of (36) is given by

$$\dot{V}_2^i(\cdot) = \dot{V}^i(\cdot) + \dot{V}_1^i(\cdot) + \tilde{q}_2^i \dot{\tilde{q}}_2^i. \tag{38}$$

From (37) and (24), $\dot{\tilde{q}}_2^i$ is rewritten as

$$\dot{\tilde{q}}_2^i = -a_2^i x_2^i + w_{21}^i \psi_{21}^i\left(q^i\right) + u^i - \dot{x}_{2d}^i. \tag{39}$$

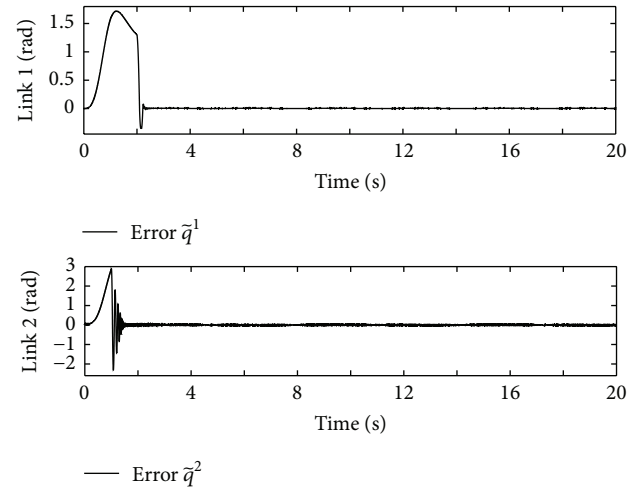


Figure 5: Tracking errors between desired trajectory and robot manipulator.

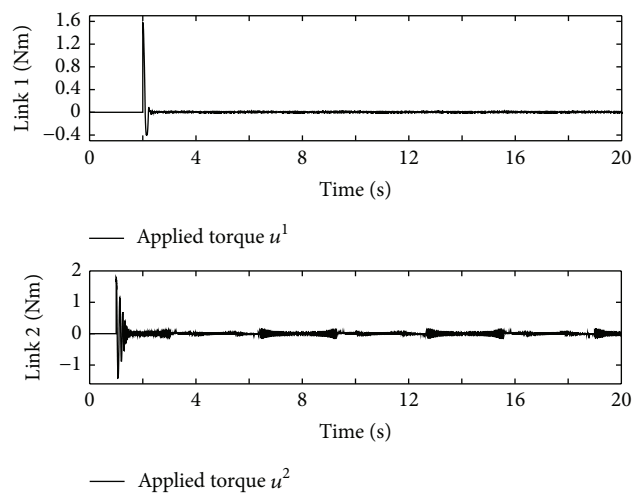


Figure 6: Applied torques for each joint via decentralized RWFONN controller.

From all of the above, (38) takes the form

$$\begin{aligned}\dot{V}_2^i(\cdot) = \dot{V}^i(\cdot) - \zeta_1^i\left(\epsilon^i\right)^2 + \tilde{q}_2^i \\ \cdot\left[-a_2^i x_2^i + w_{21}^i \psi_{21}^i\left(q^i\right) + u^i - \dot{x}_{2d}^i + \epsilon^i\right].\end{aligned} \tag{40}$$

In order for (40) to be negative definite, the control law is proposed as

$$u^i = a_2^i x_2^i - w_{21}^i \psi_{21}^i\left(q^i\right) + \dot{x}_{2d}^i - \epsilon^i - \zeta_2^i \tilde{q}_2^i. \tag{41}$$

From (34), considering (28) and (24), we have

$$\begin{aligned}\dot{x}_{2d}^i = \left(a_1^i - \zeta_1^i\right)\left(-a_1^i x_1^i + w_{11}^i \psi_{11}^i + x_2^i\right) \\ - \dot{w}_{11}^i \psi_{11}^i - w_{11}^i \frac{\partial\left(\psi_{11}^i\right)}{\partial q}\dot{q} + \ddot{q}_d^i + \zeta_1^i \dot{q}_d^i.\end{aligned} \tag{42}$$

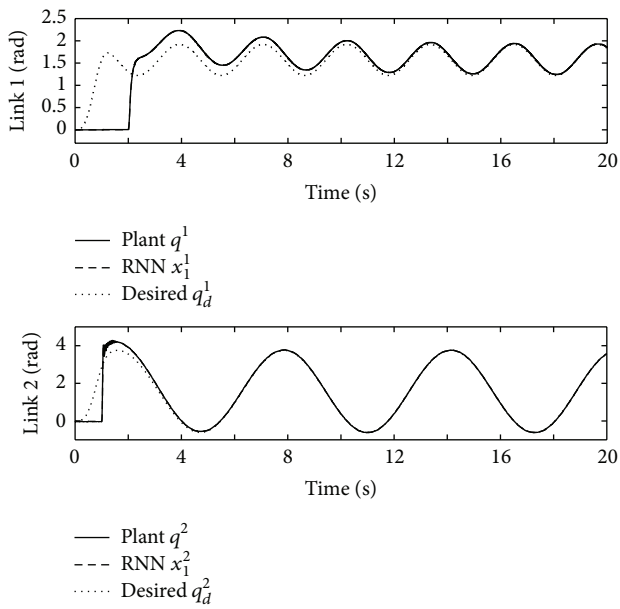


Figure 7: Identification and trajectory tracking performed by the first-order RNN control scheme.

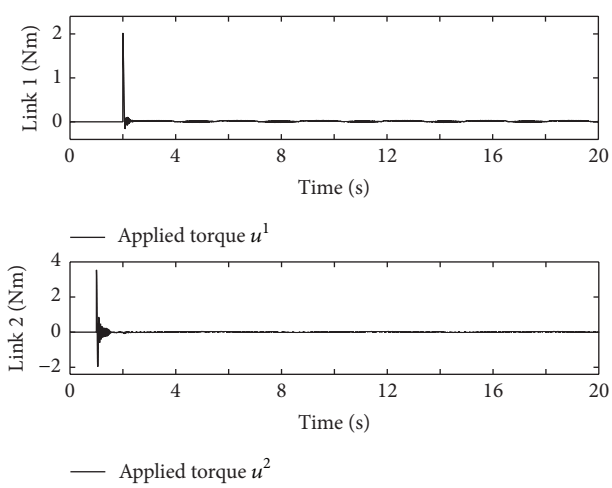


Figure 8: Applied torques for each joint of the robot manipulator via first-order RNN controller.

Accordingly,

$$\begin{aligned}\dot{V}_2^i(\cdot) &= \dot{V}^i(\cdot) - \zeta_1^i\left(\epsilon^i\right)^2 + \tilde{q}_2^i \\ &\quad \cdot\left[-a_2^i x_2^i + w_{21}^i \psi_{21}^i\left(q^i\right) + a_2^i x_2^i - w_{21}^i \psi_{21}^i\left(q^i\right)\right. \\ &\quad \left. + \dot{x}_{2d}^i - \epsilon^i - \zeta_2^i \tilde{q}_2^i - \dot{x}_{2d}^i + \epsilon^i\right] \\ &= \dot{V}^i(\cdot) - \zeta_1^i\left(\epsilon^i\right)^2 - \zeta_2^i\left(\tilde{q}_2^i\right)^2 \\ &= -\sum_{i=1}^{n} a_1^i\left(\xi_1^i\right)^2 - \zeta_1^i\left(\epsilon^i\right)^2 - \zeta_2^i\left(\tilde{q}_2^i\right)^2\end{aligned} \tag{43}$$

with $\zeta_2^i > 0$ a real value. □

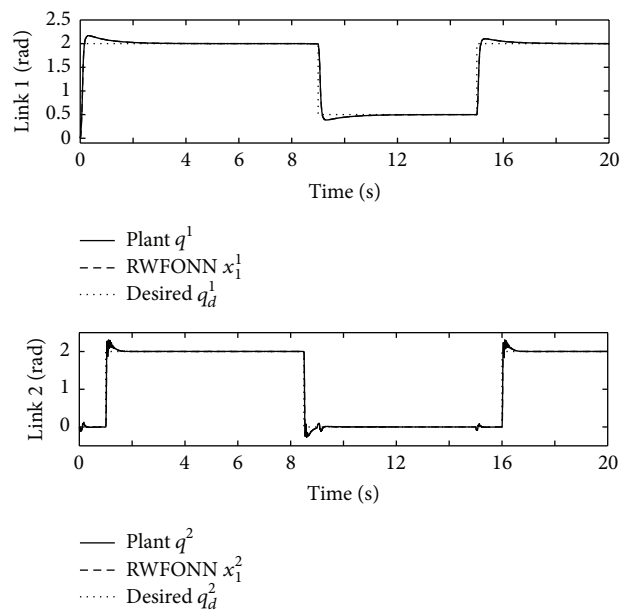


Figure 9: Identification and trajectory tracking for a step signal via decentralized RWFONN control.

Table 1: Admissible signals by the servomotors.

	$\tau_{\max}$ (Nm)	DAB signal (Vdc)	Gearbox ratio	τ_{applied} (Nm)
u^1	2	$\lvert 10\rvert$	16 : 1	32
u^2	2	$\lvert 10\rvert$	1 : 1	2

4. Real-Time Results

4.1. Robot Description. In order to evaluate in real-time the performance of the decentralized neural backstepping control scheme proposed, it is implemented on a 2-DOF robot manipulator, of our own design and unknown parameters shown in Figure 1, whose displacements are involved on the vertical plane [38]. The robot manipulator consists of two rigid links; brushless direct-drive servos are used to drive the joints, the first one with a 16 : 1 gear reduction for link 1 and direct connection of the last one for link 2. The robot arm is constituted by the MTR-70-3S-AA and MTR-55-3S-AA servomotors, manufactured by FESTO Pneumatic AG, for link 1 and link 2, respectively. Incremental encoders embedded on the servomotors deliver information, via RS-422 protocol, related with the angular displacements. Both servomotors exhibit a resolution of 4096 pulses/rev, that is, an accuracy of 0.00153398 rad/pulse. The angular velocities are computed via numerical differentiation of the angular position's signal. According to the actuator's manufacturer, the direct-drive servomotors are able to supply torques within the following bound:

$$|u| \le \tau_{\max} = 2\,\text{Nm}. \tag{44}$$

Considering that servo drives are controlled by means of a data acquisition board (DAB) using two analog channels of ±10 Vdc, with a sampling time of 0.0005 s, the voltage-torque relationship has been established as it is shown in Table 1.

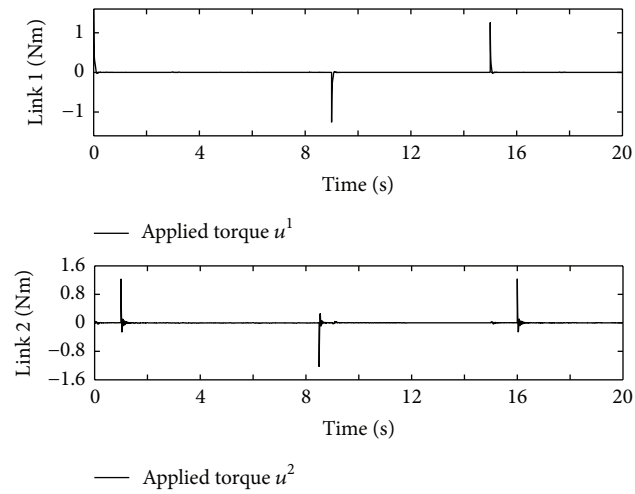


Figure 10: Exerted torques by the decentralized RWFONN controller when following the step signal.

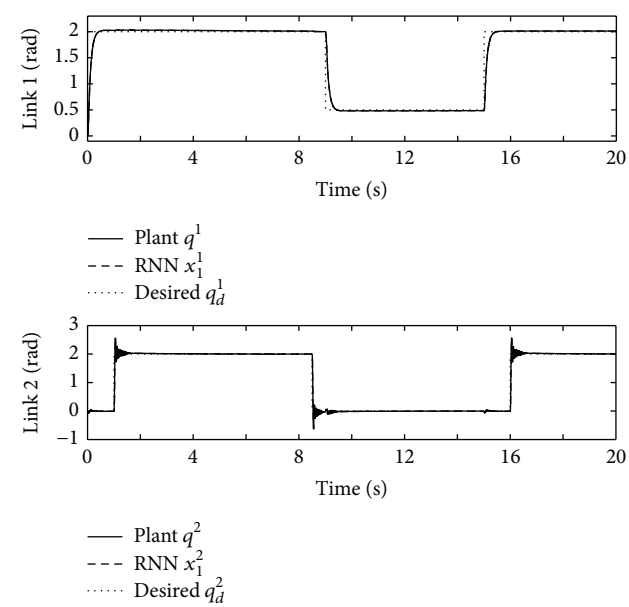


Figure 11: Identification and trajectory tracking for a step signal via RNN controller.

4.2. Tracking Neural Control. For validating the experimental results we are using the following bounded desired trajectories:

$$q_{1d} = r_1\left(1 - e^{d_1 t^3}\right) + c_1\left(1 - e^{d_1 t^3}\right)\sin\left(\omega_1 t\right),$$
$$q_{2d} = r_2\left(1 - e^{d_2 t^3}\right) + c_2\left(1 - e^{d_2 t^3}\right)\sin\left(\omega_2 t\right), \quad (45)$$

where $r_1 = \pi/2$, $c_1 = 2\pi/18$, $d_1 = -1.8$, and $\omega_1 = 2$ (rad/s) are parameters of the desired position trajectory for the first joint, whereas $r_2 = \pi/2$, $c_2 = 25\pi/36$, $d_1 = -1.8$, and $\omega_2 = 1$ (rad/s) are parameters of the desired position trajectory for the second joint. These trajectories have the following special characteristics: (a) they include a step-like end of small magnitude so that it is possible to show the transient response of the controller, (b) they incorporate a sinusoidal term to evaluate the provision to relatively rapid periodic signals, where the nonlinearities of the robot dynamics are really important, and (c) they present a term that gently increases to keep the robot in a state of operation without saturating the actuators [38].

The decentralized neural backstepping control scheme proposed is depicted in Figure 2.

It should be noticed that for a robot manipulator with more than 2 DOF, a decentralized RWFONN controller must be implemented for each joint added.

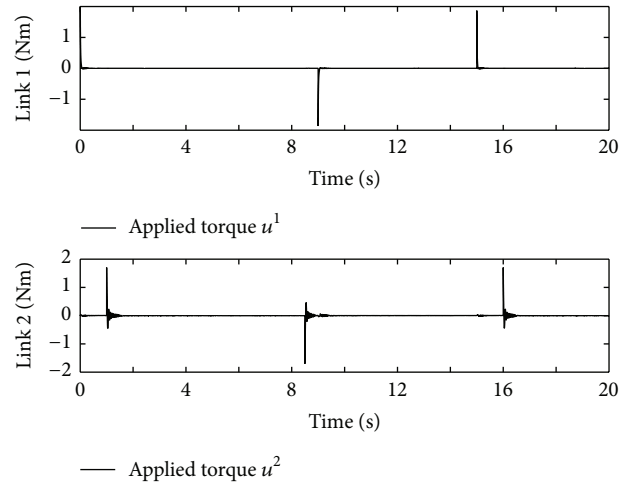


Figure 12: Applied torques by the first-order RNN controller when following a step signal.

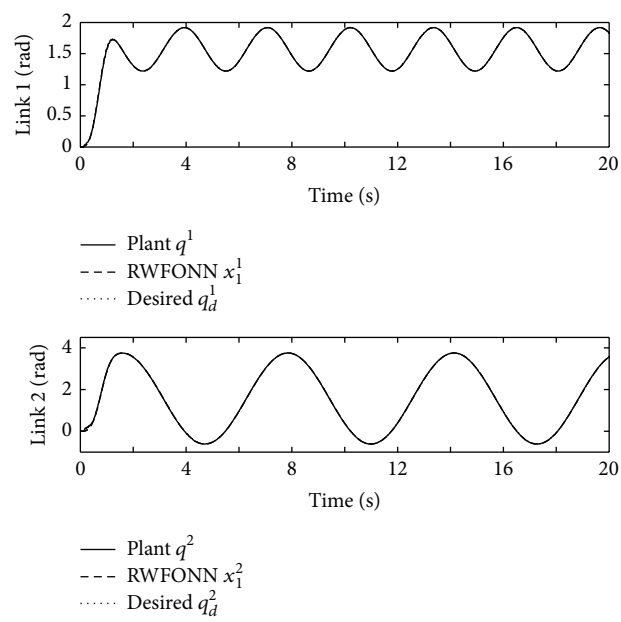


Figure 13: Identification and trajectory tracking performed by the decentralized RWFONN controller with payload added to both joints of the robot manipulator.

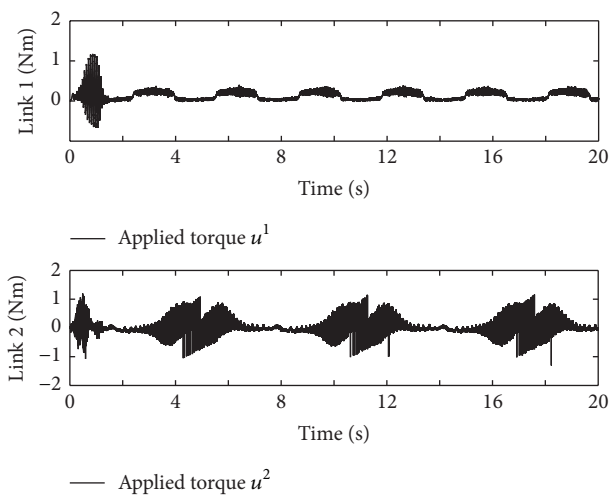


Figure 14: Exerted torques by the decentralized RWFONN control scheme when payloads are added to each joint of the robot manipulator.

4.3. Real-Time Experiments. To evaluate the efficiency and speed of the identification RWFONN scheme and the performance and robustness of the decentralized neural backstepping control scheme, we propose the next experiments under the following criteria:

(i) for joint 1: zero initial conditions for both plant and RWFONN, with a delay of 2 seconds for the controller; for joint 2: zero initial conditions for both plant and RWFONN with a delay of 1 second for the controller;

(ii) for joints 1 and 2: trajectory tracking of a step signal, with zero initial conditions for both plant and RWFONN;

(iii) for joint 1: zero initial conditions for both plant and RWFONN, adding a payload of 0.325 kg; for joint 2: zero initial conditions for both plant and RWFONN, adding a payload of 0.100 kg.

Figure 3 illustrates real-time results about identification of the plant behavior and trajectory tracking performed by the RWFONN. Figure 4 shows the identification error, that is, the difference between the plant dynamics and the RWFONN. From Figure 5, it can be shown that tracking errors tend to occur at a small region near zero. In Figure 6, the applied torques to each joint are shown in their corresponding $\tau_{\max}$ values, according to Table 1. It can be seen that both control signals are always inside the prescribed limits given by the actuators manufacturer. Figure 7 illustrates real-time results about identification of the plant behavior and trajectory tracking performed in a decentralized way by a first-order RNN, with sigmoidal functions, via a control law synthesized by a recursive method as was proposed in [6]. The corresponding applied torques to each joint from the first-order RNN control scheme are shown in Figure 8. From Figures 9–12, trajectory tracking for a step signal performed by both RWFONN and first-order RNN, as well as their corresponding exerted torques, is presented. From Figures 7–12, it has been shown that the RWFONN control scheme performs much better than the first-order RNN controller when showing a good speed for convergence to the signals to track and a less control effort. From Figures 13 and 14, real-time results about the performance of the decentralized neural control scheme here proposed are shown for the case when payloads are added to each link.

5. Conclusion

The real-time results validate the efficiency of the continuous-time decentralized wavelet neural control scheme proposed for trajectory tracking. Furthermore, it is shown that effects due to friction and gravitational forces, which can be seen as physical interconnections between the subsystems of the whole system, are absorbed by the local neural controllers. Moreover, the decentralized wavelet neural controllers exhibit a good performance in spite of forced delays and more even when payloads on each link of the robot manipulator are added.

Conflict of Interests

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

Acknowledgments

The authors thank CONACYT, Tecnológico Nacional de México and Universidad de Guadalajara.

References

[1] F. O. Tellez, A. G. Loukianov, E. N. Sanchez, and E. J. Bayro-Corrochano, "Decentralized neural identification and control for uncertain nonlinear systems: application to planar robot," *Journal of the Franklin Institute*, vol. 347, no. 6, pp. 1015–1034, 2010.

[2] C. E. Castañeda and P. Esquivel, "Decentralized neural identifier and control for nonlinear systems based on extended Kalman filter," *Neural Networks*, vol. 31, pp. 81–87, 2012.

[3] R. Garcia-Hernandez, E. N. Sanchez, E. Bayro-Corrochano, V. Santibañez, and J. A. Ruz-Hernandez, "Real-time decentralized neural block control: application to a two DOF robot manipulator," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 3, pp. 1075–1085, 2011.

[4] R. García Hernandez, E. N. Sanchez, E. Bayro-Corrochano, M. A. Llama, and J. A. RuzHernandez, " Real-time decentralized neural backstepping control: application for a two DOF robot manipulator," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 4, 2011.

[5] R. García Hernandez, J. A. Ruz Hernandez, E. N. Sanchez, and M. Saad, "Real-time decentralized neural control for a five DOF redundant robot," *Intelligent Automation & Soft Computing*, vol. 19, no. 1, pp. 23–37, 2013.

[6] E. N. Sánchez, A. Gaytan, and M. Saad, "Decentralized neural identification and control for robotics manipulators," in *Proceedings of the International Symposium on Intelligent Control*, pp. 1614–1615, October 2006.

[7] S. Álvarez, C. E. Castañeda, and F. Jurado, "Neural identification and control using high-order sliding modes," in *Proceedings of the Latin American Control Conference (CLCA '12) and Red Peruana de Control y Automatización (REPCA '12)*, pp. 23–26, Lima, Peru, 2012.

[8] F. Jurado, M. A. Flores, and C. E. Castañeda, "Continuous-time neural control for a 2 DOF vertical robot manipulator," in *Proceedings of the 8th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE '11)*, October 2011.

[9] D. W. C. Ho, P.-A. Zhang, and J. Xu, "Fuzzy wavelet networks for function learning," *IEEE Transactions on Fuzzy Systems*, vol. 9, no. 1, pp. 200–211, 2001.

[10] R.-J. Wai and H.-H. Chang, "Backstepping wavelet neural network control for indirect field-oriented induction motor drive," *IEEE Transactions on Neural Networks*, vol. 15, no. 2, pp. 367–382, 2004.

[11] C. C. Liao, C. H. Chen, Y. F. Peng, and S. C. Wu, "A combined backstepping and wavelet neural network control approach for mechanical system," in *Proceedings of the 9th Asian Control Conference (ASCC '13)*, Istambul, Turkey, June 2013.

[12] H. A. Yousef, M. E. Elkhatib, and O. A. Sebakhy, "Wavelet network-based motion control of DC motors," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1522–1527, 2010.

[13] C.-M. Lin, M.-H. Lin, and R.-G. Yeh, "Synchronization of unified chaotic system via adaptive wavelet cerebellar model articulation controller," *Neural Computing and Applications*, vol. 23, no. 3-4, pp. 965–973, 2013.

[14] C.-J. Lin and C.-C. Chin, "Prediction and identification using wavelet-based recurrent fuzzy neural networks," *IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics*, vol. 34, no. 5, pp. 2144–2154, 2004.

[15] S. J. Yoo, J. B. Park, and Y. H. Choi, "Adaptive dynamic surface control of flexible-joint robots using self-recurrent wavelet neural networks," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 36, no. 6, pp. 1342–1355, 2006.

[16] M. Dehghani, M. Ahmadi, A. Khayatian, and M. Eghtesad, "Wavelet based neural network solution for forward kinematics problem of HEXA parallel robot," in *Proceedings of the 12th International Conference on Intelligent Engineering Systems (INES '08)*, pp. 63–70, Miami, Fla, USA, February 2008.

[17] C.-H. Lu, "Design and application of stable predictive controller using recurrent wavelet neural networks," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 9, pp. 3733–3742, 2009.

[18] A. Ghadirian and M. Zekri, "Design of an on-line recurrent wavelet network controller for a class of nonlinear systems," in *Proceedings of the 2nd International Conference on Control, Instrumentation and Automation (ICCIA '11)*, pp. 373–378, December 2011.

[19] F. F. M. El-Sousy, "Intelligent optimal recurrent wavelet Elman neural network control system for permanent-magnet synchronous motor servo drive," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 1986–2003, 2013.

[20] S. Ganjefar and M. Alizadeh, "On-line identification of synchronous generator using self recurrent wavelet neural networks via adaptive learning rates," in *Proceedings of the 5th International Power Engineering and Optimization Conference (PEOCO '11)*, pp. 243–248, Shah Alam, Malaysia, June 2011.

[21] M. Sharma and A. Verma, "Adaptive tracking control for a class of uncertain non-affine delayed systems subjected to input constraints using self recurrent wavelet neural network," in *Proceedings of the International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom '10)*, pp. 60–65, IEEE, Kottayam, India, October 2010.

[22] J. J. Cordova, W. Yu, and X. Li, "Haar wavelet neural networks for nonlinear system identification," in *Proceedings of the IEEE International Symposium on Intelligent Control (ISIC '12)*, Dubrovnik, Croatia, October 2012.

[23] C. H. Lin and W.-J. Wong, "Robust backstepping RWNN control for a linear synchronous motor drive," in *Proceedings of the IEEE 10th International Conference on Power Electronics and Drive Systems (PEDS '13)*, pp. 695–700, Kitakyushu, Japan, April 2013.

[24] F. A. Diaz-Lopez, L. E. R. Velasco, O. A. D. Ramirez, and V. Parra-Vega, "Multiresolution wavenet PID control for global regulation of robots," in *Proceedings of the 9th Asian Control Conference (ASCC '13)*, pp. 1–6, IEEE, Istanbul, Turkey, June 2013.

[25] C.-M. Lin, C.-F. Tai, and C.-C. Chung, "Intelligent control system design for UAV using a recurrent wavelet neural network," *Neural Computing and Applications*, vol. 24, no. 2, pp. 487–496, 2014.

[26] G. A. Rovithakis and M. A. Christodoulou, *Adaptive Control with Recurrent High-Order Neural Networks*, Springer, 2000.

[27] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 81, no. 10, pp. 3088–3092, 1984.

[28] M. A. Cohen and S. Grossberg, "Absolute stability of global pattern formation and parallel memory storage by competitive neural networks," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 13, no. 5, pp. 815–826, 1983.

[29] P. S. Addison, J. N. Watson, and T. Feng, "Low-oscillation complex wavelets," *Journal of Sound and Vibration*, vol. 254, no. 4, pp. 733–762, 2002.

[30] X. Mi, H. Ren, Z. Ouyang, W. Wei, and K. Ma, "The use of the Mexican hat and the Morlet wavelets for detection of ecological patterns," *Plant Ecology*, vol. 179, no. 1, pp. 1–19, 2005.

[31] K. Ch. Chui and G. Chen, *Kalman Filtering with Real-Time Applications*, Springer, Berlin, Germany, 1987.

[32] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[33] I. Daubechies, *Ten Lectures on Wavelets*, CBMS–NSF Regional Conference series in Applied Mathematics, Rulgers University, AT&T Bell Labs, SIAM, New Brunswick, NJ, USA, 1992.

[34] C. S. Burrus, R. A. Gopinath, and H. Guo, *Introduction to Wavelets and Wavelet Transforms: A Primer*, Prentice Hall, Upper Saddle River, NJ, USA, 1998.

[35] L. A. Vazquez and F. Jurado, "Continuous-time decentralized wavelet neural control for a 2 DOF robot manipulator," in *Proceedings of the 11th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE '14)*, pp. 1–6, Campeche, Mexico, September-October 2014.

[36] E. B. Kosmatopoulos, M. M. Polycarpou, M. A. Christodoulou, and P. A. Ioannou, "High-order neural network structures for identification of dynamical systems," *IEEE Transactions on Neural Networks*, vol. 6, no. 2, pp. 422–431, 1995.

[37] P. A. Ioannou and A. M. Datta, "Robust adaptive control: a unified approach," *Proceedings of the IEEE*, vol. 79, no. 12, pp. 1736–1768, 1991.

[38] R. Kelly, V. Santibañez, and A. Loría, *Control of Robot Manipulators in Joint Space*, Springer, London, UK, 1995.