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

Luis A. Vázquez, Francisco Jurado, and Alma Y. Alanís

1Tecnológico Nacional de México, Instituto Tecnológico de La Laguna, Boulevard Revolución y Calzada Cuauhtémoc, 27000 Torreón, COAH, Mexico
2Centro 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@itlaguna.edu.mx

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

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

\[ \dot{x}_i^j = -a_i^j x_i^j + b_i^j \sum_{k=1}^{L} w_{jk} \prod_{j \in I} y_{j, d(j)}, \]

(1)

where \( x_i^j \) is the state of the ith neuron, \( a_i^j \) and \( b_i^j \) are positive real constants, \( w_{jk} \) is the kth adjustable synaptic weight connecting the ith state to the jth 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 \( d(j) \) are nonnegative integers. This class of neural networks form a RHONN.

The input vector for each neuron is given by

\[ y = \begin{bmatrix} y_1 \\ \vdots \\ y_j \\ y_{j+1} \\ \vdots \\ y_{i+j} \end{bmatrix} = \begin{bmatrix} s(x_1^i) \\ \vdots \\ s(x_j^i) \\ u_1 \\ \vdots \\ u_i \end{bmatrix}, \]

(2)

where \( u = [u_1 \ u_2 \ \ldots \ u_i]^T \) is the vector of external control inputs to the network.

Introducing a \( L \)-dimensional vector \( z \), defined as

\[ z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_L \end{bmatrix} = \begin{bmatrix} \prod_{j \in I_1} y_{j, d(1)} \\ \prod_{j \in I_2} y_{j, d(2)} \\ \vdots \\ \prod_{j \in I_L} y_{j, d(L)} \end{bmatrix}, \]

(3)
the RHONN model (1) can be rewritten as

$$\dot{x}_j = -a_j x_j + b_j \sum_{k=1}^{L} w_{jk} x_k^i.$$  (4)

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

$$\dot{x}_j = -a_j x_j + b_j \sum_{k=1}^{L} w_{jk} \psi_{jk}.$$  (5)

The adjustable parameter vector is defined by $w_{jk} = b_j [w_{j1}, w_{j2}, \ldots, w_{jL}]^T$, so (5) becomes

$$\dot{x}_j = -a_j x_j + (w_{jk})^T \psi_{jk},$$  (6)

where the vectors $w_{jk}$, for $k = 1, 2, \ldots, L$, represent the adjustable weights of the network, while the coefficients $a_j$, 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 $\beta$ and $\lambda$ representing expansion and dilation, respectively. Properties about wavelet vector fields 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

$$\dot{x}_j = F(x_j, u_j),$$  (7)

where $u_j \in \mathbb{R}^i$ is the input to the system, $x_j \in \mathbb{R}^i$ is the state of the system, and $F : \mathbb{R}^{i+j} \mapsto \mathbb{R}^i$ is a smooth vector field defined on a compact set $\mathcal{D} \subset \mathbb{R}^{i+j}$, where $i$ and $j$ are constants. The approximation problem consists in determining and using wavelet activation function for continuous time, if there exist weights $w_{jk}$ 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 $(x_j(t), u_j(t)) \in \mathcal{D}$ for all $t$ in some time interval $T = [0, T]$, where $T$ 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 $w_{jk}$, 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 $x_j(0) = \dot{x}_j(0)$. Then, for any $\varepsilon > 0$ and any finite $T > 0$, there exists an integer $L$ and a vector $w_{jk} \in \mathbb{R}^L$ such that the state $x_j(t)$ of the RWFONN model (6), with $L$ number of weights whose values $w_{jk} = w_{jk}^*$, satisfies

$$\sup_{t \leq T} \left| \dot{x}_j(t) - \dot{x}_j^*(t) \right| \leq \varepsilon.$$  (8)

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

$$\dot{x}_j = -a_j x_j + (w_{jk})^T \psi_{jk}(x_j, u_j).$$  (9)

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

$$\dot{x}_j = -a_j x_j + G(x_j, u_j),$$  (10)

where $G(x_j, u_j) = F(x_j, u_j) + a_j x_j$. Since $x_j(0) = x_j(0)$, the identification error $\xi_j = x_j - x_j^*$ satisfies

$$\dot{\xi}_j = -a_j \xi_j + (w_{jk})^T \psi_{jk}(x_j, u_j) - G(x_j, u_j)$$  (11)

with $\xi_j(0) = 0$.

By assumption, $(x_j(t), u_j(t)) \in \mathcal{D}$ for $t \in [0, T]$, where $\mathcal{D}$ is a compact subset of $\mathbb{R}^{i+j}$. Let

$$\mathcal{D}_\varepsilon = \{ (x_j, u_j) \in \mathbb{R}^{i+j} : \| (x_j, u_j) - (x_j^*, u_j^*) \| \leq \varepsilon, (x_j, u_j) \in \mathcal{D} \}.$$  (12)

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

$$\| \psi_{jk}(x_j, u_j) - \psi_{jk}(x_j^*, u_j^*) \| \leq l \| x_j - x_j^* \|$$  (13)

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

In what follows, it is shown that the function $(w_{jk})^T \psi_{jk}(x_j, u_j)$ 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 $w_{jk} = w_{jk}^*$ such that $(w_{jk}^*)^T \psi_{jk}(x_j, u_j)$ can approximate $G(x_j, u_j)$, in
a compact domain, to any degree of accuracy for all \((x^j, u^j)\). Hence, there exists \(w^j_{jk} = w^{*i}_{jk}\) such that
\[
\sup_{(x^j, u^j) \in \mathcal{D}_x} \left| \left( \begin{array}{c} w^{*i}_{jk} \end{array} \right)^T \Psi_{jk} \left( x^j, u^j \right) - G \left( x^j, u^j \right) \right| \leq \eta, \tag{14}
\]
where \(\eta\) is a constant.

The solution to the differential equation (11) can be expressed as
\[
\xi_j^i = \int_0^t \xi_j^{i, a(t-t)} \left[ \left( \begin{array}{c} w^{*i}_{jk} \end{array} \right)^T \Psi_{jk} \left( x^j(r), u^j(r) \right) - G \left( x^j(r), u^j(r) \right) \right] d\tau \tag{15}
\]
\[
+ \int_0^t \xi_j^{i, a(t-t)} \left[ \left( \begin{array}{c} w^{*i}_{jk} \end{array} \right)^T \Psi_{jk} \left( x^j(r), u^j(r) \right) - G \left( x^j(r), u^j(r) \right) \right] d\tau.
\]

Since each \(a_j^i\) is a positive real constant, there exists a positive constant \(\alpha\) such that \(\|\xi_j^i\| \leq \xi_j^{\infty - \alpha}\) and \(\alpha < L\), where \(L = \|w^{*i}_{jk}\|\). Based on all of the above, let \(\eta\) be chosen as
\[
\eta = \frac{(L - \alpha)\varepsilon}{2L} > 0. \tag{16}
\]

Now, consider that \((x^j(t), u^j(t)) \in \mathcal{D}_x\) \(\forall t \in [0, T]\). From (15), taking norms on both sides and using (13), (14), and (16), the following inequalities hold for \(t \in [0, T]\):
\[
\|\xi_j^i(t)\| \leq \int_0^t \|\xi_j^{i, a(t-t)}\| \left[ \left( \begin{array}{c} w^{*i}_{jk} \end{array} \right)^T \Psi_{jk} \left( x^j(r), u^j(r) \right) - G \left( x^j(r), u^j(r) \right) \right] d\tau \tag{17}
\]
\[
+ \int_0^t \|\xi_j^{i, a(t-t)}\| \left[ \left( \begin{array}{c} w^{*i}_{jk} \end{array} \right)^T \Psi_{jk} \left( x^j(r), u^j(r) \right) - G \left( x^j(r), u^j(r) \right) \right] d\tau.
\]

Using the Bellman-Gronwall lemma, it can be easily shown that
\[
\|\xi_j^i(t)\| \leq \frac{\eta}{L - \alpha} \left( \xi_j^{i, (L-\alpha)T} - 1 \right), \tag{18}
\]
\[
\|\xi_j^i(t)\| \leq \frac{\varepsilon}{2}.
\]

Suppose that \((x_j^0, u^j)\) does not belong to \(\mathcal{D}_x\) \(\forall t \in [0, T]\). Then, by the continuity of \(x^j(t)\), there exists a \(T^*\), such that \(0 \leq T^* \leq T\), such that \((x^j(T^*), u^j(T^*)) \in \mathcal{D}_x\) where \(\mathcal{D}_x\) denotes the boundary of \(\mathcal{D}_x\). Furthermore, carrying out the same analysis for \(t \in [0, T^*]\) it results \(\|\xi_j^i(t) - x^j(t)\| \leq \varepsilon/2\). 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 \(w^{*i}_{jk}\) such that each state \(x_j^i\) of the unknown dynamical system (7) satisfies
\[
\dot{x}_j^i = -a_j^i x_j^i + \left( \begin{array}{c} w^{*i}_{jk} \end{array} \right)^T \Psi_{jk} \left( x_j^i, u^j \right) \tag{19}
\]
where \(x_j^i(t) = x_{j0}^i\) is the initial state of the system. As is standard in systems identification procedures, here it is assumed that the input \(u^j(t)\) and the state \(x_j^i(t)\) remain bounded for all \(t \geq 0\). Based on the definition for \(\Psi_{jk}(x_j^i, u^j)\) given by (3), this implies that \(\Psi_{jk}(x_j^i, u^j)\) is also bounded. In the sequel, unless there exists confusion, the arguments of the vector field \(\Psi\) will be omitted. Next, the approach for estimating the unknown parameters \(w^{*i}_{jk}\) 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 \(w_{jk}\) is the estimate of the unknown weight vector \(w^{*i}_{jk}\). From (6) and (19), the identification error \(\xi_j^i = x_j^i - x_j^i\) satisfies
\[
\dot{\xi}_j^i = -a_j^i \left( x_j^i - x_j^i \right) + \left( \begin{array}{c} w_{jk} \end{array} \right)^T \Psi_{jk}(x_j^i, u^j), \tag{20}
\]
which can be rewritten as
\[
\dot{\xi}_j^i = -a_j^i \xi_j^i + \left( \begin{array}{c} w_{jk} \end{array} \right)^T \Psi_{jk}, \tag{21}
\]
where \(\left( \begin{array}{c} w_{jk} \end{array} \right)^T = \left( \begin{array}{c} w_{jk} \end{array} \right)^T - \left( \begin{array}{c} w^{*i}_{jk} \end{array} \right)^T\) denotes the parametric error [35]. The weights \(w_{jk}\) are adjusted according to the learning law
\[
w_{jk} = -\Gamma_{jk} \Psi_{jk} \xi_j^i, \tag{22}
\]
where the adaptive gain \(\Gamma_{jk} \in \mathcal{D}_w^{l \times l}\) 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 RWFO NN model given by (21) whose weights are adjusted according to (22). Then,

1. \( \xi_j, \tilde{w}_j^i \in L_{\infty} \) (i.e., \( \xi_j \) and \( \tilde{w}_j^i \) are uniformly bounded);
2. \( \lim_{t \to \infty} \xi_j(t) = 0 \).

3. Neural Backstepping Controller Design

Consider now the mathematical model of an \( i \)-DOF robot manipulator given by

\[
\tau = M(q) \dot{q} + C(q, \dot{q}) q + g(q) + f(q),
\]

where \( \tau \) is a \( 1 \times 1 \) input vector that represents the torques applied to each joint; \( q, \dot{q}, \) and \( \ddot{q} \in \mathbb{R}^i \) are the states of the system corresponding to position, velocity, and acceleration for each joint, respectively; \( M(q) \dot{q} \in \mathbb{R}^{3 \times 3} \) represents the contribution of the inertial forces to the dynamical equation; hence the matrix \( M \) represents the inertia matrix of the manipulator; \( C(q, \dot{q}) q \) \( \in \mathbb{R}^i \) represents the Coriolis forces, \( g(q) \in \mathbb{R}^i \) represents the gravitational forces, and \( f(q) \in \mathbb{R}^j \) 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 RWFO NN 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 RWFO NN model is given as follows:

\[
\begin{align*}
\dot{x}_1^i &= -a_1^i x_1^i + w_{11}^i \psi_{11}^i(q^i) + x_2^i, \\
\dot{x}_2^i &= -a_2^i x_2^i + w_{21}^i \psi_{21}^i(q^i) + u^i,
\end{align*}
\]

with \( k = 1 \) for the \( w_{ij}^i \psi_{jk}^i() \) term, where \( i = 1, 2 \) denotes the number of the joints; \( j = 1, 2 \) is for the number of states of the \( j \)th RWFO NN 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 \( j \)th joint the identification error between the neural identifier and the joint variable is defined as \( \xi_j = x_j - q_j \) for the angular position and \( \dot{\xi}_j = x_j - \dot{q}_j \) for the angular velocity. To update online the synaptic weights, the adaptive learning laws are given by \( \dot{w}_{11}^i = -\Gamma_{11}^i \psi_{11}^i(q^i) \xi_j^i \) and \( \dot{w}_{21}^i = -\Gamma_{21}^i \psi_{21}^i(q^i) \xi_j^i \), with \( \Gamma_{jk} > 0 \) as the adaptive gain and

\[
\begin{align*}
\psi_{11}^i(q^i) &= e^{-\alpha_1^i |q^i|} \cos(\lambda_1 |q^i|) + e^{-\alpha_2^i |q^i|} \cos(\lambda_2 |q^i|), \\
\psi_{21}^i(q^i) &= e^{-\alpha_1^i |q^i|} \cos(\lambda_3 |q^i|),
\end{align*}
\]

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

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

Denoting \( \xi_j^i \) as the identification error and the trajectory tracking error between the states of the neural network for position and the desired trajectory as \( \epsilon = |x_j - q^i_0| \), the output tracking error is rewritten as

\[
\ddot{q} = \xi + \epsilon,
\]

where \( \ddot{q} \) is \( |q_j - \dot{q}^i_0| \). Consequently, the error dynamics is given as

\[
\dot{\xi} = \xi + \epsilon.
\]

Considering (21), (24), and

\[
\epsilon = x_j - \dot{q}^i_0,
\]

the error dynamics is then described by

\[
\ddot{q} = -a_1^i \epsilon + (\delta_{1k}) \psi_{1k}^i - a_1^i x_j + w_{11}^i \psi_{11}^i(q^i) + x_j - \dot{q}^i_0
\]

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 \( \xi^i_1 \) and \( \xi^i_2 \), 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

\[
V_1^i(\epsilon^i, \dot{\xi}^i, \ddot{w}) = V^i(\xi^i, \ddot{w}) + \frac{1}{2} \epsilon^2
\]

with

\[
V^i(\xi^i, \ddot{w}) = \frac{1}{2} \sum_{j=1}^{n} \left( (\xi_j^i)^2 + (\Gamma_{1k}^{-1} \delta_{1k})^2 \right)
\]

taken from [36].

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

\[
\dot{V}_1^i(\cdot) = -\sum_{j=1}^{n} a_1^i (\xi_j^i)^2 \leq 0.
\]

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

\[
\dot{V}_1^i(\cdot) = \dot{V}_1^i(\cdot) + \epsilon^i [ -a_1^i x_j + w_{11}^i \psi_{11}^i + x_j - \dot{q}^i_0 ].
\]
In order to guarantee that (33) is negative definite, the desired dynamics for $x_2^i$ is proposed as

$$x_{2d}^i = -\zeta_1^i \varepsilon^i + a_1^i x_1^i - w_1^i \psi_1^i + \dot{\xi}_1^i \varepsilon^i$$

(34)

thus,

$$\dot{V}_1^i (\cdot) = \dot{V}^i (\cdot) + \varepsilon^i \left[ -a_1^i x_1^i + w_1^i \psi_1^i - \xi_1^i \varepsilon^i \\
+ d_1^i x_1^i - w_1^i \psi_1^i + \phi_d^i - \dot{q}_d^i \right]$$

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

Now, proposing the augmented Lyapunov-like function

$$V_2^i (\varepsilon^i, \xi_1^i, \bar{w}^i, \phi_d^i) = V^i (\cdot) + V_1^i (\cdot) + \frac{1}{2} (\phi_d^i)^2$$

(36)
and introducing the error
\[ \ddot{q}_2^i = x_2^i - x_{2d}^i, \] (37)
the time derivative of (36) is given by
\[ V_2^i (\cdot) = \dot{V}_i (\cdot) + \dot{V}_i (\cdot) + \ddot{q}_2^i. \] (38)
From (37) and (24), \( \ddot{q}_2^i \) is rewritten as
\[ \ddot{q}_2^i = -a_2^i x_2^i + w_{21}^i \psi_{21}^i (q^i) + u^i - \dot{x}_{2d}^i. \] (39)

From all of the above, (38) takes the form
\[ V_2^i (\cdot) = \dot{V}_i (\cdot) + \ddot{q}_2^i \cdot \left[ -a_2^i x_2^i + w_{21}^i \psi_{21}^i (q^i) + u^i - \dot{x}_{2d}^i + e^i. \right] \] (40)

In order for (40) to be negative definite, the control law is proposed as
\[ u^i = a_2^i x_2^i - w_{21}^i \psi_{21}^i (q^i) + \dot{x}_{2d}^i - e_i - \dot{q}_2^i. \] (41)
From (34), considering (28) and (24), we have
\[ x_{2d}^i = (a_i^i - \zeta_i^i) \left( -a_1^i x_1^i + w_1^i \psi_1^i + x_2^i \right) - w_{11}^i \psi_{11}^i \frac{\partial (\psi_{11}^i)}{\partial q} \dot{q} + \dot{q}_d^i + \zeta_i^d. \] (42)
Accordingly,

\[ V_i^2 (\cdot) = V_i (\cdot) - \zeta_i (e')^2 - \bar{q}_i^2 \]

\[ = V_i (\cdot) - \zeta_i (e')^2 - \zeta_i (q')^2 - \zeta_i (q')^2 - \zeta_i (q')^2 \]

\[ = -\sum_{i=1}^{n} a_i q_i^2 (\xi_i)^2 - \zeta_i (e')^2 - \zeta_i (q')^2 \]

\[ = -\sum_{i=1}^{n} a_i (\xi_i)^2 - \zeta_i (e')^2 - \zeta_i (q')^2 \]

with \( \zeta_2 > 0 \) a real value.

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 in 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:

\[ |u| \leq \tau_{\text{max}} = 2 \text{ Nm.} \]
4.2. Tracking Neural Control. For validating the experimental results we are using the following bounded desired trajectories:

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

(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 \( r_2 = \pi / 2, c_2 = 25\pi / 36, d_2 = -1.8 \), 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.
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 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


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