International Journal of Control, Automation, and Systems (2011) 9(5):895-904
DOI 10.1007/s12555-011-0510-y http://www.springer.com/12555

Task-space Neuro-Sliding Mode Control of Robot Manipulators under Jacobian Uncertainties

Rodolfo García-Rodríguez and Vicente Parra-Vega

Abstract: Cartesian robot control is an appealing scheme because it avoids the computation of inverse kinematics, in contrast to joint robot control approach. For tracking, high computational load is typically required to obtain Cartesian robot dynamics. In this paper, an alternative approach for Cartesian tracking is proposed under assumption that robot dynamics is unknown and the Jacobian are uncertain. A neuro-sliding second order mode controller delivers a low dimensional neural network, which roughly estimates inverse robot dynamics, and an inner smooth control loop guarantees exponential tracking. Experimental results are presented to confirm the performance in a real time environment.

Keywords: Cartesian sliding mode control, motion control, tracking robots, uncertain kinematics.

1. INTRODUCTION

To implement a joint robot control, the desired joint references are computed from desired Cartesian coordinates using inverse mapping and its derivatives up to second order. However, the high computational load and the ill-posed nature of the inverse kinematics mapping are the main disadvantages of this scheme. To circumvent the computation of the inverse kinematics and reduce the computational load, the Cartesian control stands as a useful control strategy. Therefore, due to several robotic tasks are coded in operational coordinates -generally in Cartesian coordinates; the Cartesian control allows an efficient and intuitive design of the task. Basically, two types of Cartesian controllers have been proposed: inverse Jacobian based and transpose Jacobian based. Nevertheless, an exact knowledge of the Jacobian is required in these controllers, that is, the exact knowledge of link lengths of the robot manipulator and the payload variations are needed. Recently, based on the seminal work of Miyazaki and Masutani [23] several Cartesian controllers have been proposed under the assumptions that the Jacobian is uncertain and the robot trajectories are free of singularities. Posteriorly, considering that the Jacobian matrix can be parametrized linearly, a passivity based approach which guarantee stability in the closed loop is presented in [3-7,31].

Now, if we are interested that having the end effector of the robot manipulator follow a desired trajectory, Cartesian robot dynamics knowledge is required. However, Cartesian robot dynamics demands even more computational power than computing the inverse kinematics. Therefore, non-model based control strategies which guarantee convergence of the Cartesian tracking errors is desirable. Based on the Cartesian dynamics and the Jacobian transpose, a Cartesian tracking controller is proposed in [24]. Additionally, to compensate the noise, unmodeled dynamics and to avoid the use of high gains an additional term is included in the controller. By other hand, in [8,9,19] adaptive Jacobian tracking controllers have been reported, assuming that the knowledge of robot dynamics and kinematics are uncertainties. Lyapunov stability is used to guarantee the convergence of the position and velocity tracking errors, considering that the Jacobian matrix is parametrized linearly.

At the same time, in order to compensate the nonlinearity and the uncertainties of the robot dynamics, the neural networks have been used into the control algorithms and pattern recognition, among others; to approximated smoothly vector fields with a certain desired accuracy, where the accuracy is controlled by the architecture of the neural network. In [13-17] the neural networks learn the inverse dynamics of the manipulators based on gradient descent method or adaptive control. However, a high computational cost is required because a great number of neurons in each layer are used. Additionally, to provide robustness some approaches add a high frequency input in the controller which represents the principal disadvantage in the practical applications. In [22] is proposed a neuro-visual servoing control for a planar robot manipulator assuming that link lengths of the robot manipulator are uncertain. In order to avoid the drift in the parameter estimated and some possible overshoot in the estimated of the gravitational vector a neural network is used. That is, it is not necessary to known the gravity vector and boundedness of the Jacobian matrix against a perturbation. Additionally, in [32] an adaptive neural network based controller is

Manuscript received September 28, 2010; revised May 11, 2011; accepted June 3, 2011. Recommended by Editorial Board member Youngjin Choi under the direction of Editor Jae-Bok Song.
Rodolfo García-Rodríguez is with the Facultad de Ingeniería y Ciencias Aplicadas, Universidad de los Andes, Av. San Carlos de Apoquindo 2200, Las Condes, Santiago, Chile (e-mail: rgarcia1@miunades.cl).
Vicente Parra-Vega is with the Robotics and Advanced, Manufacturing Division, CINVESTAV, Campus Saltillo, México (e-mail: vparra@cinvestav.mx).

proposed to approximate robot kinematics and dynamics of a pair of robotic fingers for manipulating an object. Specifically, a Gaussian radial basis function neural network (RBFNN) is used to approximate Jacobian matrices in order to guarantee boundedness of the position errors. Moreover, in [20] an adaptive neural controller is proposed to avoid the regression matrix computation. A parametric uncertainty in the Jacobian matrix and the external perturbations are compensated using a radial basis function neural network. In order to avoid the linearity- in-parameters, in [21] is presented a tracking of redundant robot manipulators assuming that parametric uncertainty is approximated by neural networks with 60 neurons in the hidden layer and an additional term is included to compensate external error and the approximated errors.

In this paper is proposed an alternative approach to solve the Cartesian tracking of non-redundant robot manipulator assuming a parametric uncertainty. This approach is based on the assumption that the initial conditions and desired trajectories belong to the workspace Ω that defines a hyperspace free of singular configurations, which is an standard assumption for joint robot control. The control strategy is based on a second order sliding (SOSM) surface and a low dimensional neural network, to guarantee Cartesian tracking errors convergence with a smooth control effort under parametric uncertainties. The key of this approach is to design a Cartesian manifold which is invariant to parametric robot kinematics, such that, exponential convergence is guaranteed despite Jacobian uncertainties. This is carried out, by designing a second order sliding mode (SOSM) control which is piecewise continuous in contrast to the classical first order sliding mode control. That is, because of the fact that the sliding mode condition is relegated to the first order time derivative of the sliding surface, the possibility of chattering in the closed loop is eliminated. Compared with other approaches [20,22,32] the low dimensional neural network is used to approximate the robot parametric uncertainty. Additionally, the estimated Jacobian matrix is proposed by the user, considering that the exact Jacobian matrix of the robot manipulator is a function of the articular joints. Finally, given that the SOSM satisfy the sliding mode condition, the system is robust against structured unmodelled dynamics if the disturbances enter into the system via the input space. Stability and boundedness of all closed loop error signals is proved in the sense of Lyapunov while exponential convergence is established using SOSM control arguments. Representative real-time experimental results validate the proposed approach.

2. DYNAMICAL MODEL OF THE MANIPULATOR

Consider the following dynamical model of the robot manipulator obtained with the Euler-Lagrange (E-L) modeling formalism

$$H(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau, \quad (1)$$

where $q \in \Re^n$, is a n-dimensional vector of the joint angular positions, $H(q) \in \Re^{n\times n}$ is the inertia matrix, $C(q,\dot{q}) \in \Re^{n\times n}$ is the centrifugal Coriolis matrix, $g(q) \in \Re^n$ is the vector of gravity forces, and $\tau \in \Re^n$ stands for the torque inputs[1].

Some important structural properties of E-L systems (1) are very useful in our ulterior analysis.

Property 1: The inertia matrix $H(q)$ is symmetric, positive definite, and both $H(q)$ and $H^{-1}(q)$ are uniformly bounded as a function of $q \in \Re^n$ [25].

Property 2: The matrix $C(q,\dot{q})$ and the time derivative of the inertia matrix $\dot{H}(q)$ satisfy

$$\dot{q}^T(\dot{H}(q) - 2C(q,\dot{q}))\dot{q} = 0 \quad \forall q,\dot{q} \in \Re^n. \quad (2)$$

Property 3: The general form of the E-L systems (1) can be parameterized linearly as follows, [28],

$$H(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = Y\Theta, \quad (3)$$

where $Y = Y(q,\dot{q},\dot{q},\ddot{q}) \in \Re^{n\times p}$ is known as the regressor and $\Theta \in \Re^p$ is a vector of constant unknown parameters.

Property 4: Equation (1) is passive from the torque input τ to velocity output $\dot{q}$ with storage function $H_0(q,\dot{q})$; that is

$$\dot{q}^T\tau \geq \frac{dH_0}{dt},$$

where $H_0(q,\dot{q})$ is the total energy of the system described in (1), such that

$$H_0(q,\dot{q}) = K(q,\dot{q}) + P(q)$$

with $K(q,\dot{q}) = \frac{1}{2}\dot{q}^T H(q)\dot{q}$ being the kinetic energy and $P(q)$ the potential energy of the system, assuming that $min_q P(q) = 0$.

Property 3 becomes, in terms of a nominal reference $(\dot{q}_r, \ddot{q}_r) \in \Re^{2n}$ as

$$H(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + g(q) = Y_r\Theta, \quad (4)$$

where the regressor, $Y_r = Y_r(q,\dot{q},\dot{q}_r,\ddot{q}_r)$ is an $n\times p$ matrix and Θ is an p×1 vector. Adding and subtracting (4) into (1), the open-loop error equation becomes

$$H(q)\dot{S}_q + C(q,\dot{q})S_q = \tau - Y_r\Theta, \quad (5)$$

where

$$S_q := \dot{q} - \dot{q}_r \quad (6)$$

defines a joint error manifold. This representation has

[1]Notice that we have not considered noise nor quantization sensor errors, typically of real world applications.

been very useful in previous works to design a variety of control strategies [1,2,17,19], in particular for sliding mode control. At this stage the control problem is the design of a Cartesian controller τ that guarantees tracking without resorting on $Y_r\Theta$ using smooth desired Cartesian trajectories, rather than desired joint coordinates $(q_d^T, \dot{q}_d^T, \ddot{q}_d^T)^T \in \Re^{3n}$.

To proceed, we need convenient open loop error dynamics parameterized in Cartesian errors, that is, we need to build an explicit representation of (6) in terms of Cartesian coordinates, then design the neuro-SOSM controller.

3. CARTESIAN ERROR MANIFOLD UNDER JACOBIAN UNCERTAINTIES

Let $X \in \Re^m$ be the end-effector position vector with respect to a fixed reference inertial frame. The relation between joint space and task space is described by forward kinematics as

$$X = f(q), \tag{7}$$

where X stands for the Cartesian coordinates, $f(.): \Re^n \to \Re^n$ is the forward kinematics map, generally a non-linear transformation.[2] Given that differential kinematics establishes a mapping between joint velocities and end-effector velocities, we have that

$$\dot{q} = J^{-1}(q)\dot{X}, \tag{8}$$

where $J(q)^{-1} \in \Re^{n\times m}$ represent the inverse Jacobian matrix. Based on (8), the nominal reference $\dot{q}_r$ can be defined as

$$\dot{q}_r = J^{-1}(q)\dot{X}_r. \tag{9}$$

Notice now that the joint nominal reference $\dot{q}_r$ in (9) appears as function of Cartesian nominal reference $\dot{X}_r$, i.e., the Cartesian nominal reference is mapped into joint nominal reference.

3.1. Cartesian adaptive control

Let $\dot{X}_{r1}$ be defined as

$$\dot{X}_{r1} = \dot{X}_{d1} - \alpha_1 \Delta X_1, \tag{10}$$

where the subscript d denotes desired trajectories, α_1 is a positive-definite diagonal matrix and $\Delta X_1 = X - X_d$. Now, substituting (8) and (9) with $\dot{q}_r = J^{-1}(q)\dot{X}_{r1}$ in (6), we obtain

$$\begin{aligned} S_q &= J^{-1}(q)(\dot{X} - \dot{X}_{r1}) \\ &= J^{-1}(q)S_{x1}, \end{aligned} \tag{11}$$

where $S_{x1} = \dot{X} - \dot{X}_{r1} = \Delta\dot{X}_1 + \alpha_1 \Delta X_1$ is called Cartesian error manifold. Therefore, the joint error manifold S_q in (11) is defined as a function of the Cartesian error manifold S_{x1}. It is noteworthy that (9) and (11) allow us to input desired Cartesian coordinates directly, as function of inverse Jacobian.

Assuming parametric uncertainties in robot dynamics, we can design the Cartesian version of the classical adaptive control for robots manipulators proposed by [28].

Theorem 1 (Cartesian Adaptive Control): Assuming that initial conditions and desired trajectories are defined in a singularities-free robot workspace. Consider known kinematic parameters and unknown dynamic ones, then

$$\tau = -K_d S_q + Y_r \hat{\Theta},$$
$$\dot{\hat{\Theta}} = -\Gamma Y_r^T S_q,$$

where $K_d = K_d^T > 0 \in \Re^{n\times n}$, $\Gamma = \Gamma^T > 0 \in \Re^{p\times p}$. Then ΔX_1 and $\Delta \dot{X}_1$ tends to zero asymptotically.

Proof: Considering a Lyapunov function $V = \frac{1}{2}S_q^T H(q) S_q + \frac{1}{2}\Delta\Theta^T \Gamma^{-1}\Delta\Theta$ then $\dot{V} = -S_q K_d S_q \le 0$. Since $\dot{V} \le 0$, we can state that V is also bounded. Therefore, S_q and $\Delta\Theta$ are bounded. This implies that $\hat{\Theta}$ and $J^{-1}(q)S_{x1}$ is bounded if $J^{-1}(q)$ is well posed for all t. From the definition of S_{x1} we have that $\Delta\dot{X}_1$, ΔX_1 are also bounded. Since $\Delta\dot{X}_1$, ΔX_1, $\Delta\Theta$ and S_q are bounded we have that $\dot{S}_q$ is bounded.This shows that $\ddot{V}$ is bounded. Hence, $\dot{V}$ is uniformly continuous. Using the Barbalat's lemma [28], we have that $\dot{V} \to 0$ at $t \to \infty$. This implies that ΔX_1 and $\Delta\dot{X}_1$ tends to zero as t tends to infinity. Then, tracking errors ΔX_1 and $\Delta\dot{X}_1$ are asymptotically stable [17].

Remark 1: Notice that the controller exhibits a PD structure plus passivity-based inverse dynamic compensation, i.e.,

$$\begin{aligned} \tau &= -K_d S_q + Y_r \hat{\Theta} \\ &= \underbrace{-K_p(t)\Delta X_1 + K_v(t)\Delta\dot{X}_1}_{PD} + Y_r\hat{\Theta}, \end{aligned}$$

where state dependant feedback gains are defined as

$$K_p(t) = K_d \alpha_1 J^{-1}(q),$$
$$K_v(t) = K_d J^{-1}(q)$$

with α_1 as is defined previously.

In this controller, we can recognize that the exact knowledge of $J^{-1}(q)$ and the robot structure -the regressor Y_r; are required. In addition, the assumption that $J^{-1}(q)$ is well posed for all t is very restrictive.

[2]We consider nonredundant robots thus $m = n$.

Because it is not possible to verify a priori the assumption on $rank(J^{-1}(q))=n$ for all t since $q(t)$ may exhibit a transient response so that $J(q)$ losses rank, thus the stability domain is very small.

In order to eliminate these drawbacks, a neural network to approximate continuous functions will be proposed to ensure the boundedness of all closed loop signals while tracking is ensured by means of tailoring a modified Cartesian nominal reference $\dot{X}_r$ to introduce a second order sliding mode.

Now, the modified Cartesian nominal reference $\dot{X}_r$ is defined as follows

$$\begin{aligned}\dot{X}_r &= \dot{X}_d - \alpha\Delta X + S_d - K_i\sigma,\\ \dot{\sigma} &= sgn(S_e),\end{aligned} \tag{12}$$

where α is a positive-definite diagonal matrix, $\Delta X = X - X_d$, K_i is positive-definite diagonal matrix and function $sgn(*)$ stands for the signum function of (*). Now, substituting (12) in (9) and (8) in (6) we have that

$$\begin{aligned}S_q &= J^{-1}(q)(\dot{X} - \dot{X}_r),\\ S_q &= J^{-1}(q)(\dot{X} - \dot{X}_d + \alpha\Delta X - S_d\\ &\quad + K_i \int_{t_0}^{t} sgn(S_e(\tau))d\tau),\\ S_q &\triangleq J^{-1}(q)\left[S_e + K_i \int_{t_0}^{t} sgn(S_e(\tau))d\tau\right],\end{aligned} \tag{13}$$

where

$$\begin{aligned}S_e &= S_x - S_d,\\ S_x &= \Delta\dot{X} + \alpha\Delta X,\\ S_d &= S_x(t_0)exp^{-\kappa(t-t_0)}, \quad \kappa > 0.\end{aligned}$$

Remark 2: Notice that the sliding mode condition is induced by the $sgn(S_e)$ term in (13) and exponential convergence of tracking error is established i.e., the discontinuity associated to the sliding mode present in $S_q=0$ is relegated to the first order time derivative of $\dot{S}_q = 0$. Then, the possibility of chattering in the closed loop dynamics is eliminated. In addition, it allows us to avoid the use of the boundary layer or a continuous approximation of function sgn(*) [12], i.e., dynamic sliding mode is imposed on the evolution of the sliding surface S_q.

To complete the parametrization of (4), it is necessary to obtain the time derivative of (9) using the Cartesian nominal reference defined in (12), so that

$$\begin{aligned}\ddot{q}_r &= J^{-1}(q)\ddot{X}_r + \dot{J}^{-1}(q)\dot{X}_r\\ &= J^{-1}(q)(\dot{S}_e + K_i sign(S_e)) + \dot{J}^{-1}(q)\dot{X}_r,\end{aligned}$$

where the first term of the right side is discontinuous. Since neural networks can not approximate discontinuous signals, we need to avoid discontinuous signals in the function $Y_r\Theta$. To solve this, $\ddot{q}_r$ is decompose into continuous and discontinuous terms as follows

$$\ddot{q}_r = \ddot{q}_{cont} + J^{-1}(q)K_i Z, \tag{14}$$

where

$$\begin{aligned}\ddot{q}_{cont} &= J^{-1}(q)\ddot{X}_{cont} + \dot{J}^{-1}(q)\dot{X}_r\\ Z &= \tanh(\lambda S_q) - sign(S_e)\end{aligned} \tag{15}$$

for the $\tanh(x) = [\tanh(x_1),\ldots,\tanh(x_k)]^T$ as the continuous hyperbolic tangent function of $X \in \Re^k$, $\lambda = \lambda^T \in \Re^{n\times n} > 0$.

Notice that $\ddot{X}_{cont} = \dot{S}_e - K_i \tanh(\lambda S_e)$ is continuous and Z in (15) yields a bounded discontinuous signal and it has the following properties: $Z \geq -1$, $Z \leq 1$, $Z_{S_e\to 0^-} = -1$, $Z_{S_e\to 0^+} = +1$ and $Z_{S_e\to\pm\infty} = 0$.

Now, the parametrization of (4) using (9) and (14) becomes

$$H(q)\ddot{q}_{cont} + C(q,\dot{q})\dot{q}_r + g(q) = Y_{cont}\Theta - \tau_d, \tag{16}$$

where the regressor $Y_{cont} = Y_r(q,\dot{q},\dot{q}_r,\ddot{q}_{cont})$ is continuous due to $(\dot{q}_r,\ddot{q}_{cont}) \in C^1$, and $\tau_d = H(q)J^{-1}(q)K_i Z$ denotes a bounded endogenous disturbances subject to matching conditions. Finally, the open-loop error dynamics [26,27] is obtained adding and subtracting (16) into (1) such that

$$H(q)\dot{S}_q = -C(q,\dot{q})S_q + \tau - Y_{cont}\Theta - \tau_d. \tag{17}$$

Remark 3: If $Y_{cont}\Theta$ and τ_d are available, the controller $\tau = -K_d S_q + Y_{cont}\Theta + \tau_d$ would guarantee asymptotic tracking, for a Lyapunov function $V = \frac{1}{2}S_q^T H(q)S_q$. However if $Y_{cont}\Theta$ and τ_d are not available and assuming that the Jacobian is not exactly known, the nominal reference (9) cannot be used, that is $\dot{q}_r = J^{-1}(q)\dot{X}_r$ is not available. Then, an uncalibrated joint manifold arises which cannot be used for a sliding mode controller. In this case, the switching policy must take place in the Cartesian manifold S_x, which is invariant to uncertainty on $J^{-1}(q)$.

3.2. Uncalibrated joint error manifold

Considering that the Jacobian is not exactly known, the nominal reference defined previously in (9) cannot be used. Then, the uncalibrated nominal reference is defined as

$$\dot{\hat{q}}_r = \hat{J}^{-1}(q)\dot{X}_r, \tag{18}$$

where $\hat{J}^{-1}(q)$ stands as an estimate of $J^{-1}(q)$ so that $rank(\hat{J}^{-1}(q)) = n$, for all t and for all $q \in \Omega$, where $\Omega = \{q \mid rank(J(q))\} = n$. Then, substituting (18) in (6), we obtain a Cartesian error manifold, coined uncertain joint error manifold as follows

$$\hat{S}_q = \dot{q} - \dot{q}_r = J^{-1}(q)\dot{X} - \hat{J}^{-1}(q)\dot{X}_r. \tag{19}$$

Notice that $\hat{S}_q$ is available since $\dot{q}$ and $\dot{\hat{q}}$ are available.

Taking the time derivative of (18) and rearranging the terms into continuous and discontinuous, the uncertain parametrization of $Y_r\hat{\Theta}$ is defined as

$$H(q)\ddot{q}_{cont} + C(q,\dot{q})\dot{q}_r + g(q) = Y_{cont}\hat{\Theta} - \overline{\tau}_d, \tag{20}$$

where $Y_{cont}\hat{\Theta}$ is continuous with $\ddot{q}_{cont} = \hat{J}^{-1}\ddot{X}_{cont} + \dot{\hat{J}}^{-1}\dot{X}_r$ and $\overline{\tau}_d = H(q)\hat{J}^{-1}K_iZ$ means a bounded high frequency signal.

Now, adding and subtracting (20) into (1), the uncertain open loop error equation arises as follows

$$H(q)\dot{\hat{S}}_q + C(q,\dot{q})\hat{S}_q = -\overline{\tau}, \tag{21}$$

where $\overline{\tau} = -\tau + Y_{cont}\hat{\Theta} + \overline{\tau}_d$.

Based on the property of the neural networks to approximate smooth bounded functions, in the next section is presented a design of the neural-adaptive controller τ where $Y_{cont}\hat{\Theta}$ is approximated by a neural network and the endogenous discontinuous disturbance function $\overline{\tau}_d$ is considered not available.

4. CONTROLLER DESIGN

4.1. Neural network architecture

To approximate continuous regressor Y_{cont} a tree network structure that satisfies Stone-Weierstrass theorem [10] is used i.e., many neurons on one layer feed a single neuron on the next layer. The input-output relationship for this generic architecture is given as $y_i = \phi(\sum_{i=1}^{m} x_i w_i)$ where x_i is the input to neural network, w_i is the weight of connections and ϕ the activation function. Notice that tree structure could have one or more hidden layers where the linear activation function is used as the last stage of a multilayer neural network.

Considering the property that neural networks can approximate any smooth function $f(x)$ with x belongs to a compact set **S** of $\Re^n$ [11,18], it is defined regularly a sufficiently large neural network to approximate $f(x)$ such that

$$|f(x) - \phi(\mathbf{X}^T\overline{\mathbf{W}})| \le \varepsilon_1(x),$$

where $\|\varepsilon_1(x)\| \le \varepsilon_N$ is a functional reconstruction error with $\varepsilon_N > 0$ for all $x \in \mathbf{S}$. Now, if for a low dimensional neural network exists a bounded optimal approximating weights $\mathbf{W}_{low}$, the approximation function of $f(x)$ is given as

$$\hat{f}(x) = \phi(\mathbf{X}^T\tilde{\mathbf{W}}) + \varepsilon_2(x),$$

where $\tilde{\mathbf{W}}$ is the estimated of the optimal approximating weights and $\varepsilon_2(x) > \varepsilon_1(x)$.

Taking into account that the regressor Y_{cont} is continuous and assuming that exist a parametric uncertainties in the robot dynamics, we will use the low dimensional neural network to approximate the unknown function $f(x) = Y_{cont}\hat{\Theta}$ as follows

$$\hat{f}(x) = Y_{cont}\hat{\Theta} \equiv \mathbf{X}^T\tilde{\mathbf{W}} + \varepsilon_2(x), \tag{22}$$

where the activation function ϕ is used as a linear function, $\mathbf{X}^T \in \Re^{n\times p}$ are the inputs to the neural network and $\tilde{\mathbf{W}} \in \Re^p$ are the estimated neural network weights. To difference other approaches, the architecture of the low dimensional neural network is based on the ADAptive LINear Element (ADALINE) proposed by Widrow and Hoff [30] which consists of a single neuron of the McCulloch-Pitts type. Additionally, given that the regressor Y_{cont} is a function of q, $\dot{q}$, $\dot{q}_r$ and $\ddot{q}_{cont}$ and is independent of dynamic parameters, the inputs to the neural network can be defined as

$$\mathbf{X}^T = [q, \dot{q}, \dot{q}_r, \ddot{q}_{cont}]. \tag{23}$$

Now, based on the information provided by the input vectors to the neural network is possible to determine its size. Therefore, the number of weights that uses the neural network is defined as $4n$ where n represent the number of degrees of freedom of the robot manipulator and 4 represents the weights for each degree of freedom, Fig. 1.

Remark 4: The neural network proposed in this work can be considered as minimal architecture to approximate the robot dynamics taking into account the regressor elements, that is, the neural network architecture provide a very easy and cheap estimation of $Y_{cont}\hat{\Theta}$. This architecture takes more relevance since the neural network is driven by a second order sliding mode, with bounded adaptive weights, as it will be shown in the next section.

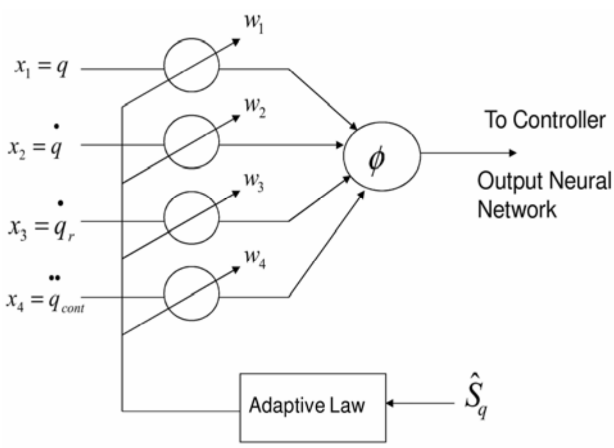

Fig. 1. Proposed neural network architecture.

4.2. Neuro-controller design

Considering the parameterization of $f(x)$ by a low dimensional neural network, we have that the open loop neuro-error equation is defined as

$$H(q)\dot{\hat{S}}_q + C(q,\dot{q})\hat{S}_q = -\overline{\tau}, \tag{24}$$

where $\overline{\tau} = -\tau + \mathbf{X}^T\tilde{\mathbf{W}} + \delta$ with $\delta = \varepsilon_2(x) + \overline{\tau}_d$. Finally, we have the following result.

Theorem 2 (Exponential Stability): Consider the robot dynamics (1) in closed loop with the controller given by

$$\tau = -K_d\hat{S}_q + \mathbf{X}^T\hat{\mathbf{W}}, \tag{25}$$

$$\hat{\mathbf{W}}(t) = \hat{\mathbf{W}}(0) - \int_0^t \Gamma X(\tau)\hat{S}_q(\tau)d\tau, \tag{26}$$

where $K_d = K_d^T \in \Re^{n\times n}$ is a gain matrix, $\hat{\mathbf{W}} \in \Re^{n\times p}$ is a neural network weights, and $\Gamma = \Gamma^T \in \Re^{p\times p}$. Then, for a large enough gain K_d and small errors in the initial conditions, local exponential tracking errors are assured provided that $K_i geq \| \dot{J}^{-1}(q)\hat{S}_x + J^{-1}(q)\dot{\hat{S}}_x + \Delta\dot{J}\dot{X}_r + \Delta J\ddot{X}_r \|$.

Proof: The closed loop error dynamics between equations (25) and (24) is given as

$$H(q)\dot{\hat{S}}_q = -C(q,\dot{q})\hat{S}_q - K_d\hat{S}_q + \mathbf{X}^T\Delta\mathbf{W} - \delta + v, \tag{27}$$

where $\Delta\mathbf{W} = \hat{\mathbf{W}} - \mathbf{W}$ and v is a virtual input defined for analysis proposes. For sake of clarity, the proof is organized in four parts as follows.

Part 1 (Passivity): The passive mapping from virtual input v to output $\hat{S}_q$ is defined as

$$\begin{aligned}\int_{t_0}^t \hat{S}_q^T v d\tau &= \int_{t_0}^t \hat{S}_q^T (H(q)\dot{\hat{S}}_q + C(q,\dot{q})\hat{S}_q \\ &\quad + K_d\hat{S}_q - \mathbf{X}^T\Delta\mathbf{W} + \delta) \\ &= \int_{t_0}^t \frac{dV}{dt}d\tau + \int_{t_0}^t \hat{S}_q^T(K_d\hat{S}_q + \delta)d\tau,\end{aligned} \tag{28}$$

where

$$V = \frac{1}{2}\hat{S}_q^T H\hat{S}_q + \frac{1}{2}\Delta\mathbf{W}^T\Gamma^{-1}\Delta\mathbf{W} \tag{29}$$

represents a storage function. Due to $V \geq 0$ it is considered as candidate Lyapunov function.

Part 2 (Boundedness of Closed-loop Trajectories): The time derivative of (29) leads to

$$\dot{V} = -\hat{S}_q^T K_d\hat{S}_q - \hat{S}_q^T\varepsilon_2(x) - \hat{S}_q^T\overline{\tau}_d. \tag{30}$$

Note that the term $\hat{S}_q^T\overline{\tau}_d$ is radially unbounded[3] only when $\hat{S}_q \to \infty$ and for bounded signals it is zero only at $\hat{S}_q = 0$. These arguments imply that $\| \hat{S}_q^T\overline{\tau}_d \| \leq \eta \| \hat{S}_q \|$ where $\eta = \| H(q) \| \| \hat{J}^{-1}(q) \| \| K_i \|$. Then, (30) becomes

$$\dot{V} \leq -\hat{S}_q^T K_d\hat{S}_q + \eta \| \hat{S}_q \| + \| \hat{S}_q \| \varepsilon_{N2}, \tag{31}$$

where $\| \varepsilon_2(x) \| \leq \varepsilon_{N2}$. For small initial errors belonging to a neighborhood ε_3 with radius $r > 0$ centered in the equilibrium $\hat{S}_q = 0$, there exists a large enough feedback gain K_d such that $\hat{S}_q$ converges intoa set-bounded ε_0. Thus the boundedness of tracking error can be concluded, namely

$$\hat{S}_q \to \varepsilon_3 \quad \text{as} \quad t \to \infty.$$

This result stands for local stability of $\hat{S}_q$ provided that the state is near of the desired trajectories for any initial conditions. This boundedness in the $\mathcal{L}_\infty$ sense, leads to the existence of the constant $\varepsilon_1 > 0$ such that

$$\| \hat{S}_q \| < \varepsilon_4.$$

Then, $(\hat{S}_e, \sigma) \in \mathcal{L}_\infty$ and since desired trajectories are C^2 and feedback gains are bounded, we have that $(\dot{q}_r, \ddot{q}_{cont}) \in \mathcal{L}_\infty$, which implies that $\hat{Y}_{cont} \in \mathcal{L}_\infty$, $\mathbf{X} \in \mathcal{L}_\infty$ and $\dot{\hat{\mathbf{W}}} \in \mathcal{L}_\infty$. Then, the output of the neural network is also bounded. According previous arguments and the boundedness of the robot dynamics -Coriolis matrix, gravitational term; and due to inertia matrix is positive-definite and upper bounded; the right hand side of (27) with v =0 is bounded, such that $\dot{\hat{S}}_q \in \mathcal{L}_\infty$. Then, exists a bounded scalar $\varepsilon_5 > 0$ such that

$$\| \dot{\hat{S}}_q \| < \varepsilon_5.$$

So far, we conclude the boundedness of all closed-loop error signals.

Part 3 (Sliding Mode): Now, we show that a sliding mode at S_e=0 arises for all time. Adding and subtracting $J^{-1}(q)X_r$ to (19) we have that

$$\begin{aligned}\hat{S}_q &= J^{-1}(q)\dot{X} - \hat{J}^{-1}(q)\dot{X}_r \pm J^{-1}(q)\dot{X}_r \\ &= J^{-1}(q)(\dot{X} - \dot{X}_r) - (\hat{J}^{-1}(q) - J^{-1}(q))\dot{X}_r \\ &= J^{-1}(q)S_x - \Delta J\dot{X}_r,\end{aligned} \tag{32}$$

where $\Delta J = \hat{J}^{-1}(q) - J^{-1}(q)$. If we multiply (32) by $J(q)$ we obtain

$$S_x = J(q)\hat{S}_q + J(q)\Delta J\dot{X}_r.$$

Given that $S_x = S_e + K_i\sigma$, we have that

[3] Let $S_q : \Re^m \to \Re$ be a function such that $S_q(x)\to\infty$ as $\| x \| \to \infty$, then S_q is said to be radially unbounded.

$$S_e = -K_i\sigma + J(q)(\hat{S}_q + \Delta J\dot{X}_r). \quad (33)$$

Taking the time derivative of (33), we obtain the following second order sliding mode

$$\dot{S}_e = -K_i sign(S_e) + \frac{d}{dt}\left\{J(q)(\hat{S}_q + \Delta J\dot{X}_r)\right\}. \quad (34)$$

Now, if we multiply (34) by S_e^T, we obtain

$$\begin{aligned} S_e^T\dot{S}_e &= -K_i\,|\,S_e\,| + S_e^T\left(\frac{d}{dt}\left\{J(q)\hat{S}_q + J(q)\Delta J\dot{X}_r\right\}\right) \\ &\le -K_i\,|\,S_e\,| + \varepsilon_6\,|\,S_e\,| \\ &\le -\mu\,|\,S_e\,|, \end{aligned} \quad (35)$$

where μ=K_i–ε_6 with $\varepsilon_6 = \dot{J}(q)\hat{S}_q + J(q)\dot{\hat{S}}_q + \dot{J}(q)\Delta J\dot{X}_r + J(q)\Delta\dot{J}\dot{X}_r + J(q)\Delta J\ddot{X}_r$. Then, if $K_i > \varepsilon_6$ we obtain the sliding mode condition. Therefore, $\mu > 0$ guarantees the existence of a sliding mode at S_e=0 at time $t_q \le \frac{|\,S_e(t_0)\,|}{\mu}$. However, notice that for any initial condition $S_e(t_0)$=0 and hence $t_q \equiv 0$. This implies that a sliding mode in $S_e(t)$=0 is enforced for all time without reaching phase and then $S_x(t)$=S_d for all t.

Part 4 (Exponential Convergence): Sliding mode at S_e implies that $S_x = S_d = S_x(t_0)e^{-kt}$ thus

$$\Delta\dot{X} = -\alpha\Delta X + S_x(t_0)e^{-k(t-t_0)}.$$

Now, if k is tuned large enough so that $S_d \approx 0$ for some small time $0 < t_d \ll 1$ then $S_d = 0\,\forall\, t \ge t_d > 0$, i.e., exponential stability of tracking errors is guaranteed since the solution of S_x=0 goes to zero exponentially, $\Delta\dot{X} = -\alpha\Delta X$.

Remark 5: It is important to notice that the mapping from $\mathbf{X}^T\mathbf{W}$ to $-\hat{S}_q$ defined in (26) is passive, that is

$$-\int_0^t \hat{S}_q^T \mathbf{X}^T \mathbf{W} d\tau \ge -\frac{1}{2}\tilde{\mathbf{W}}^T(0)\Gamma^{-1}\tilde{\mathbf{W}}(0).$$

At the same time, it is showed that dissipativity arises from the input $\dot{S}_e$ to the output S_e where the storage function is defined as $V_e = 1/2 S_e^T S_e$. Then, the passivity property of the robot manipulator is preserved in the closed loop using a neuro-adaptive law.

Remark 6: The integral term in the Cartesian nominal reference $\dot{X}_r$, is a continuous function which introduces stronger error correction of the error trajectories with respect to S_e=0. Then, the second order sliding mode is piecewise continuous, in contrast to the classical first order sliding mode control. Therefore, $K_d\hat{S}_q$ is also a continuous function, i.e., the controller is continuous.

Remark 7: If K_i and S_d are equal to zero, we will obtain the nominal reference frequently defined in the literature [14,18,29]. Although the value of K_i is an important parameter to establish a sliding mode, conservative values are used. K_i cannot be known a priori since it depends on the state space variables.

Remark 8: Unlike other approaches which guarantee only bounded tracking error for low dimensional neural network [15,18] we prove exponential convergence with the low dimensional neural network where its size is defined by the degree of freedomof the robot manipulator. Furthermore, because of the fact that the neural network is driven by $\hat{S}_q$, it is possible to guarantee boundedness of the weights.

5. EXPERIMENTAL RESULTS

In this section we present the experimental results carried out on 2 degree of freedom planar robot arm (Fig. 2), whose parameters are shown in Table 1. These experiments were developed under LabWindows 5.0 with Pentium 4, 1.5 Ghz, and a 1 ms. sampling time. Additionally, the desired task in Cartesian space is defined as a circle centered at X=(0.5,0)m with radius of 0.1m in 2.5s.

Each experiment was executed under different initial conditions, zero initial velocity and assuming that the Jacobian matrix is uncertain, that is, the Jacobian matrix is parameterized in terms of a regressor times as parameter vector. To get parametric uncertainty, this vector is multiplied by a factor to get X% of uncertainty with respect to the nominal value. In these experiments, the neural network has only one layer and four weights per degree of freedom where $w_{ij}(0)$=0. Additionally, as is defined in (23), the input to the neural network is given as $\mathbf{X} = [q_{ji}, \dot{q}_{ji}, q_{r-ji}, \dot{q}_{r-ji}]^T$ for j=1,2 and j=1,2,3,4. It is worth noticing that the term $S_x(t_0)$ can be obtained directly from sensor data in $t = t_0$ i.e., when the robot

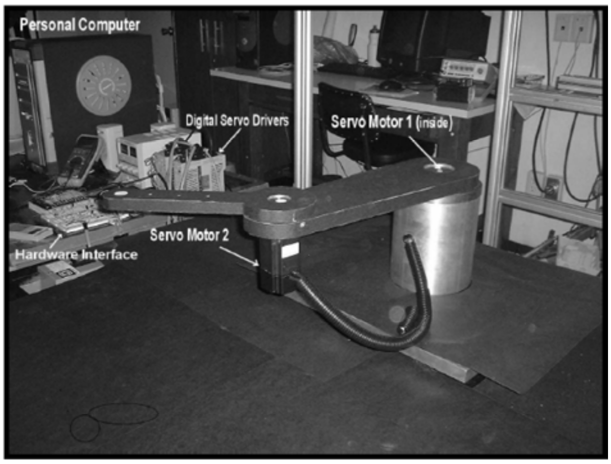

Fig. 2. High speed planar robot.

Table 1. Parameters of the robot arm.

Parameter	m_1	m_2	l_1	l_2
Value	8 Kg	5 Kg	0.5 m	0.35 m
Parameter	I_1	I_2	l_{c1}	l_{c2}
Value	0.02 Kgm^2	0.16 Kgm^2	0.19 m	0.12 m

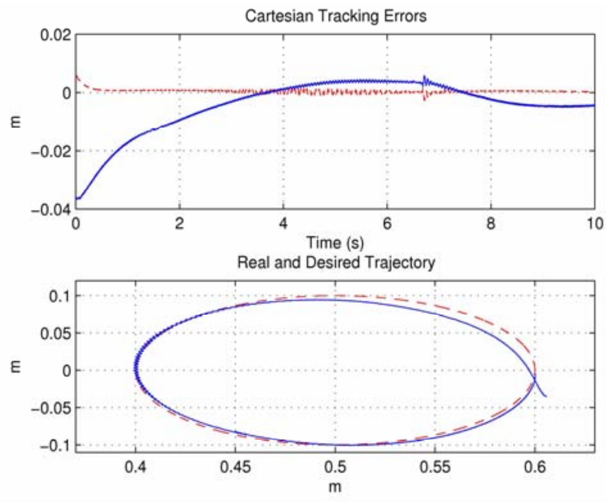

Fig. 3. Theorem 2 (Exponential Tracking): Cartesian tracking errors (Top), End-effector tracking trajectory (Bottom).

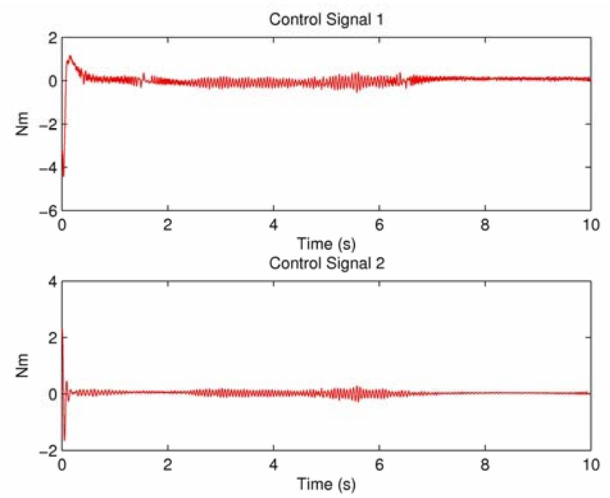

Fig. 4. Theorem 2 (Exponential Tracking): Control input for each joint.

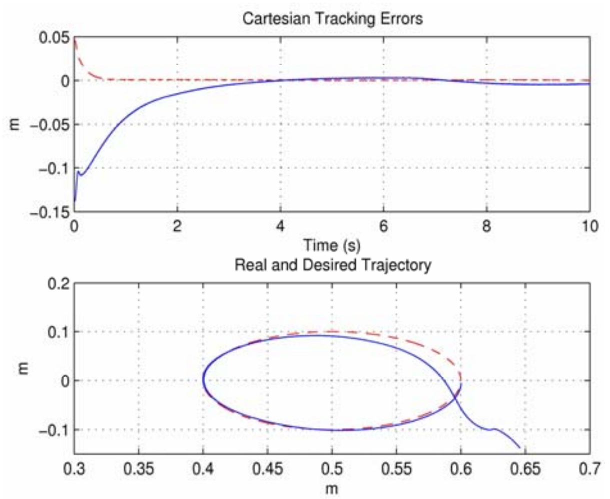

Fig. 5. Theorem 2 (Exponential Tracking): Cartesian tracking errors (Top), End-effector tracking trajectory (Bottom).

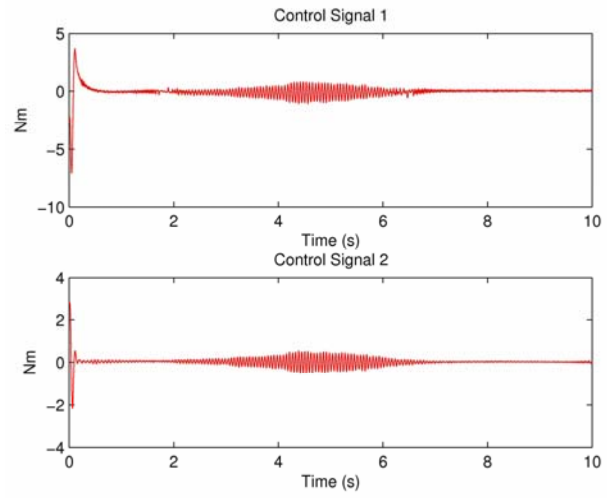

Fig. 6. Theorem 2 (Exponential Tracking): Control input for each joint.

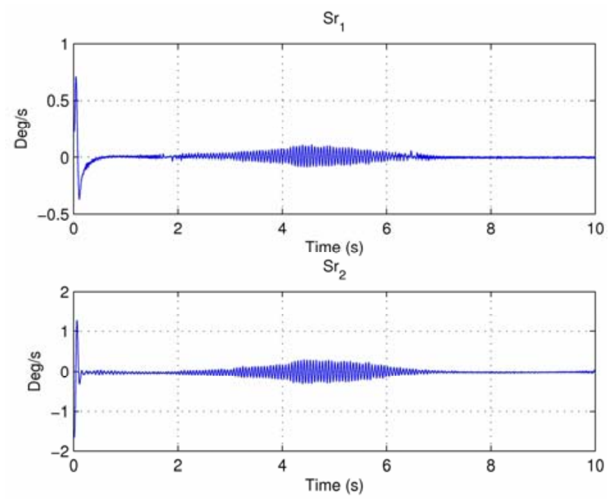

Fig. 7. Cartesian error manifold S_x.

manipulator starts moving we have that $\Delta\dot{q}(t_0)=0$ then $S_x(t_0)=\alpha\Delta q(t_0)$ is available. Therefore, the reaching phase is eliminated.

Assuming 75% of parametric uncertainty in the Jacobian matrix, the exponential convergence of the Cartesian tracking errors and the performance of the end-effector when it tracks a desired trajectories are shown in Fig. 3. Because to the fact that sliding mode condition is relegated to the first order time derivative of $\hat{S}_r$, the possibility of chattering in the closed loop is eliminated. Accordingly, the joint torques applied to the manipulator are assumed free of chattering i.e., the frequency is normal in direct drive motors, see Fig. 4.

Now, if we assume that exists a 90% of parametric uncertainty in the Jacobian matrix and the initial conditions are away from the desired trajectory. The robot manipulator converges exponentially to the desired trajectory in a very short time -in order to enforce sliding mode condition, see Fig. 5. Therefore, the convergence to zero of the Cartesian tracking errors is established as shown in Fig. 5. The performance and smoothness of the

Table 2. Feedback gains.

K_{d1}	K_{d2}	α_1	α_2	K_1	K_2	k	Γ	Fig.
11	2.2	6.5	2	0.01	0.01	10	75	4-3
15	1.5	5	5	0.01	0.01	20	100	6-7

control signal is shown in Fig. 6. In Fig. 7 is shown the performance of the Cartesian error manifold which tends to zero, ensuring the tracking trajectory in a short time. As proved in Theorem 2, the experimental results show an exponential stability without any knowledge of the robots dynamics and considering that the Jacobian is uncertain. The feedback gains used in these experiments are shown in Table 2. Finally, the feedback gains are tuned in trial-and-error basis according to the interplay of each gain in the closed loop system.

6. CONCLUSIONS

An alternative solution to the problem of tracking tasks without knowledge of the robot dynamics and assuming that the Jacobian is uncertain is presented. A neuro-sliding mode controller is proposed to guarantee exponential stability. Moreover, experimental results on a simple but characteristic robot are presented, to visualize the real time stability properties of the proposed scheme.

REFERENCES

[1] S. Arimoto, *Control Theory of Non-linear Mechanical Systems*, Oxford University Press, 1996.

[2] B. Brogliato, I.-D. Landau, and R. Lozano-Leal, “Adaptive motion control of robot manipulators: a unified approach based on passivity,” *International Journal of Robust and Nonlinear Control*, vol. 1, no. 3, pp. 187-202, 1991.

[3] C. C. Cheah, “On duality of inverse Jacobian and transpose Jacobian in task -space regulation of robots,” *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 2571-2576, 2006.

[4] C. C. Cheah, M. Hirano, S. Kawamura, and S. Arimoto, “Approximate Jacobian control for robots with uncertain kinematics and dynamics,” *IEEE Trans. on Robotics and Automation*, vol. 19, no. 4, pp. 692-702, 2003.

[5] C. C. Cheah, M. Hirano, S. Kawamura, and S. Arimoto, “Approximate Jacobian control with task-space damping for robot manipulators,” *IEEE Trans. on Automatic Control*, vol. 19, no. 5, pp. 752-757, 2004.

[6] C. C. Cheah, S. Kawamura, and S. Arimoto, “Feedback control for robotics manipulator with uncertain Jacobian matrix,” *Journal of Robotics Systems*, vol. 16, no. 2, pp. 120-134, 1999.

[7] C. C. Cheah and H. C. Liaw, “Inverse Jacobian regulator with gravity compensation: stability and experiment,” *IEEE Trans. on Robotics*, vol. 21, no. 4, pp. 741-747, 2005.

[8] C. C. Cheah, C. Liu, and J. J. E. Slotine, “Adaptive Jacobian tracking control of robots with uncertainties in kinematics, dynamics and actuator models,” *IEEE Trans. on Automatic Control*, vol. 51 no. 6, pp. 1024-1029, 2006.

[9] C. C. Cheah and J. J. E. Slotine, “Adaptive tracking control for robots with unknown kinematics and dynamics properties,” *International Journal of Robotics Research*, vol. 25, no. 3, pp. 283-296, 2006.

[10] N. E. Cotter, “The stone-weierstrass theorem and its application to neural network,” *IEEE Trans. on Neural Networks*, vol. 1, no. 4, pp. 290-295, 1990.

[11] G. Cybenko, “Approximation by superpositions of a sigmoidal function,” *Mathematics of Control, Signals, and Systems*, vol. 2, pp. 303-314, 1989.

[12] C. Edwards and S. K. Spergeon, *Sliding Mode Control: Theory and Applications*, Taylor & Francis, London, 1998.

[13] M. Ertugrul and O. Kaynak, “Neuro sliding mode control of robotic manipulators,” *Mechatronics*, vol. 10, no. 1-2, pp. 239-263, 2000.

[14] S. S. Ge and C. C. Hang, “Structural network modeling and control of rigid body robots,” *IEEE Trans. on Robotics and Automation*, vol. 14, no. 5, pp. 823-827, 1998.

[15] S. S. Ge and T. H. Harris, *Adaptive Neural Network Control of Robotic Manipulators*, World Scientific, 1994.

[16] A. Karakasoglu and M. Sundareshan, “A recurrent neural network-based adaptive variable structure model following control of robotic manipulators,” *Automatica*, vol. 31, no. 10, pp. 1495-1507, 1995.

[17] F. L. Lewis and C. T. Abdallaah, *Control of Robot Manipulators*, Macmillan, New York, 1994.

[18] F. L. Lewis, A. Yegildirek, and K. Liu, “Multilayer neural-net robot controller with guaranteed tracking performance,” *IEEE Trans. on Neural Networks*, vol. 7, no. 2, pp. 388-399, 1996

[19] C. Liu, C. C. Cheah, and J. J. E. Slotine, “Adaptive Jacobian tracking control of rigid-link electrically driven robots based on visual task-space information,” *Automatica*, vol. 42, pp. 1491-1501, 2006.

[20] L. Cheng, Z.-G. Hou, and M. Tan, “Adaptive neural network tracking control of manipulators using quaternion feedback,” *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 3371-3376, 2008.

[21] L. Cheng, Z.-G. Hou, and M. Tan, “Adaptive neural network tracking control for manipulators with uncertain kinematics, dynamics and actuator model,” *Automatica*, vol. 45, no. 10, pp. 2312-2318, 2009.

[22] G. Loreto and R. Garrido, “Stable neurovisual servoing for robot manipulators,” *IEEE Trans. on Neural Networks*, vol. 17, no. 4, pp. 953-965, 2006.

[23] F. Miyazaki and Y. Masutani, “Robustness of sensory feedback control based on imperfect Jacobian,” *Robotic Research: The Fifth International Symposium*, pp. 201-208, 1990.

[24] S. A. Moosavian and P. Evangelos, “Modified transpose Jacobian control of robotic system,” *Automatica*, vol. 43, pp. 1226-1233, 2007.

[25] R. Ortega and M. Spong, “Adaptive motion control of rigid robots: a tutorial,” *Proc. IEEE Conf. on Decision and Control*, pp. 1575-1883, 1988.

[26] V. Parra-Vega, "Second order sliding mode control for robot arms with time base generator for finite-time tracking," *Dynamics and Control*, vol. 11, pp. 174-186, 2001.

[27] V. Parra-Vega, R. García-Rodríguez, and F. Ruiz Sánchez, "Second order sliding mode adaptive neurocontrol for robots arms with finite time convergence," *Proc. of the 16th IFAC World Congress*, 2005.

[28] J. J. E. Slotine and W. Li, "On the adaptive control of robot manipulator," *Int. Journal of Robotics Research*, vol. 6, no. 3, pp.49-59, 1987.

[29] Y. Stepanenko and Ch.Y. Su, "Variable structure control of robots manipulators with nonlinear sliding manifolds," *Int. Journal of Control*, vol. 58, pp. 285-300, 1993.

[30] B. Widrow and M. E. Hoff, "Adaptive switching circuits," *Institute of Radio Engineering IRE WESCON Convention Record*, Part 4, pp. 96-104,1960.

[31] H. Yazarel and C. C. Cheah, "Task-space adaptive control of robotic manipulators with uncertainties in gravity regressor matrix and kinematics," *IEEE Trans. on Automatic Control*, vol. 47, no. 9, pp. 1580-1585, 2002.

[32] Y. Zhao and C. C. Cheah, "Neural network control of multifingered robot hands using visual feedback," *IEEE Trans. on Neural Networks*, vol. 20, no. 5, pp. 758-767, 2009.

Rodolfo García-Rodríguez has his B.S. degree in Industrial Electronic Engineering from the Instituto Tecnológico de Puebla in 1997. He received his M.S. and Ph.D. degrees from the Research Center for Advanced Studies (CINVESTAV) in 2002 and 2005, respectively. From 1998-2000 he was with Gates Rubber de México as an Engineer in the Engineering Department. Currently, he is a Professor at Universidad de los Andes, Chile in the Facultad de Ingeniería y Ciencias Aplicadas. His research interests include robotic hands, control theory and mechatronic systems.

Vicente Parra-Vega received his B.Eng. degree in Control and Computing and his B.Eng. degree in Electronics and Communications, both from the Nuevo León University, Nuevo León, México, in 1987, and an M.Sc. degree in Automatic Control from the Research Center for Advanced Studies (CINVESTAV), San Pedro Zacatenco, México, in 1989, and a Ph.D. degree in Electrical Engineering from the Mathematical Engineering and Information Physics Department of the University of Tokyo, Tokyo, Japan in 1995, under the supervision of Prof. S. Arimoto. Currently, he is a Full Professor at the CINVESTAV. His current research interests include collaborative multirobots, high-precision servosystems, haptic interfaces, control theory, visual servoing, teleoperators, high-speed CNC, and unmanned aerial and submarine robots. He is the leading cofounder of two new postgraduate programmes in Mexico, the first one in Mechatronics and the second one in Robotics and Advanced Manufacturing. Prof. Parra-Vega is a Regular Member of the Mexican Academy of Science, the National Researcher System, and from 1997, he has participated in several committees of the Mexican Council for Science and Technology.