Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Adaptive $\mathscr{H}_\infty$ Tracking Control Design via Neural Networks of a Constrained Robot System

Andre Petronilho, Adriano A. G. Siqueira and Marco H. Terra*

***Abstract*— In this paper, a nonlinear adaptive neural network tracking control with a guaranteed $\mathscr{H}_\infty$ performance is proposed for a constrained robot manipulator with plant uncertainties. The neural network is used to learn the unknown dynamics by an adaptive algorithm. Moreover, a force sensor is built to measure the forces and torques between the experimental robot UArm II end-effector and the environment. Finally, results obtained from the implementation of the proposed controller in the manipulator UArm II, under a constrained movement, are presented.**

I. INTRODUCTION

With the growth of automation in industries, the number of robots used to perform tasks such as writing, scribing and grinding, where the robot end-effector keeps contact with its environment, has been continuously increasing. In these mechanical systems, named constrained robots, there exist variables that deteriorate the representativity of the nominal model, such as parametric uncertainties, unmodelled dynamics, and external disturbances. Therefore, the study of control techniques to deal with this kind of problem deserves special attention.

An adaptive neural network tracking control with a guaranteed $\mathscr{H}_\infty$ performance is developed in [3] for unconstrained manipulators; the neural network is employed first to approximate a continuous function defined by the robot dynamic equation and torque disturbances; the $\mathscr{H}_\infty$ controller is designed to attenuate the effect of the approximation error (generated by the neural network algorithm and considered as the disturbance to be attenuated) on the tracking error. A comparison of this adaptive neural network tracking control with model-based controllers, based on results obtained from an actual manipulator, is performed in [6].

A fuzzy logic controller equipped with an adaptive algorithm is presented in [4] to achieve $\mathscr{H}_\infty$ performance for both holonomic and nonholonomic constrained robots. As in [3], the approximation error includes the robot dynamics and torque disturbances. All these $\mathscr{H}_\infty$ control strategies are difficult to be implemented in an actual robot. The approximation errors of these controllers include the robot dynamics and it is difficult to verify, as assumed in [3], the L_2 property of these errors, that is essential to guarantee the $\mathscr{H}_\infty$ attenuation criterion.

In [2], a hybrid motion/force tracking control design with an adaptive scheme that guarantees $\mathscr{H}_\infty$ performance for holonomic constrained robots is proposed. For this approach it is required a precise knowledge of the entire robot dynamic model in order to guarantee the best performance of this controller.

In this paper, an adaptive neural network controller with $\mathscr{H}_\infty$ performance is proposed to holonomic constrained robots. It includes the variable structure control (VSC) proposed in [1] to eliminate the condition of square-integrability of the approximation error. Moreover, it is considered that a nominal part of the robot dynamics is known for the control algorithm. In this case, the neural network is employed to approximate only the uncertain part of the robot dynamics.

The proposed adaptive neural network-based controller is implemented in the experimental manipulator UArm II [7], which performs a constrained movement. In order to measure the forces between the UArm II end-effector and the environment, a suitable force sensor device, which uses one-directional piezoelectric force sensors, was designed and built in our laboratory.

This paper is organized as follows: the reduced order model of a constrained robot is presented in Section II; an adaptive neural network-based controller with guaranteed $\mathscr{H}_\infty$ performance for constrained robots is proposed in Section III; and, finally, a description of the force sensor device and experimental results are presented in Section IV.

II. REDUCED ORDER MODEL FOR HOLONOMIC CONSTRAINED ROBOTS

Consider a robot manipulator subject to holonomic constraints on its end-effector. The equation of the m constraints is of the form

$$\phi(q) = 0$$

where $q \in \Re^n$ are the joint positions and $\phi(q) : \Re^n \to \Re^m$ is a smooth function. The dynamic equation of the constrained robot is given from Lagrange theory as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + J^T(q)\lambda + \tau_d \qquad (1)$$

where $M(q) \in \Re^{n\times n}$ is the symmetric positive definite inertia matrix, $C(q,\dot{q}) \in \Re^{n\times n}$ is the Coriolis and centripetal matrix, $G(q) \in \Re^n$ are the gravitational torques, $\tau \in \Re^n$ are the applied torques, $\lambda \in \Re^m$ is a vector of generalized Lagrangian multipliers associated with the constraints, $J(q) = (\partial\phi/\partial q) \in \Re^{m\times n}$ is the Jacobian matrix, and $\tau_d \in \Re^n$ denotes the external disturbances.

If the Jacobian matrix $J(q)$ has full row rank m for all $q \in \Re^n$, then the vector q can be partitioned as $q = [(q^1)^T \ (q^2)^T]^T$, where $q^1 = [q^1_1 \dots q^1_{n-m}]^T$ describes the constrained motion of the manipulator and $q^2 = [q^2_1 \dots q^2_m]^T$

*The authors are with the Electrical Engineering Department - University of São Paulo at São Carlos, C.P.359, São Carlos, SP, 13560-970, Brazil
E-mails: apetroni,siqueira,terra@sel.eesc.usp.br

denotes the remaining joints variables. Moreover, there exist an open set $\Omega_c \subseteq \Re^{n-m}$ and a function $\sigma : \Omega_c \to \Re^m$ such that $\phi(q^1, \sigma(q^1)) = 0$.

Therefore, the reduced form of constrained robot (1) is given as ([2], [4] and [8])

$$M(q^1)L(q^1)\ddot{q}^1 + C_L(q^1,\dot{q}^1)\dot{q}^1 + G(q^1) = \tau + J^T(q)\lambda + \tau_d \tag{2}$$

where

$$L(q^1) = \begin{bmatrix} I_{n-m} \\ \frac{\partial\sigma(q^1)}{\partial q^1} \end{bmatrix},$$

I_{n-m} means identity matrix and $C_L(q^1,\dot{q}^1)\dot{q}^1 = M(q^1)L(q^1) + C(q^1,\dot{q}^1)L(q^1)$. The matrices $M(q^1)$, $C(q^1,\dot{q}^1)$ and $G(q^1)$ in (2) are obtained by substituting $q^2 = \sigma(q^1)$ and $\dot{q} = L\dot{q}^1$ in $M(q)$,$C(q,\dot{q})$ and $G(q)$ in (1), respectively.

III. Adaptive neural network-based $\mathscr{H}_\infty$ control

To obtain the error dynamic equation, it is defined the following tracking error $\bar{x}(t)$, see [2] and references therein for more details,

$$\bar{x}(t) \doteq \begin{bmatrix} \bar{x}_1(t) \\ \bar{x}_2(t) \end{bmatrix} \doteq \begin{bmatrix} q^1(t) - q^1_d(t) \\ \dot{q}^1(t) - \dot{q}^1_d(t) + p(q^1(t) - q^1_d(t)) \end{bmatrix} \tag{3}$$

for some constant $p > 0$, where $q^1_d(t)$ is a desired joint trajectory for $q^1(t)$ with its time derivatives $\dot{q}^1_d(t)$, $\ddot{q}^1_d(t)$ and $d^3(q^1_d(t))/dt^3$ bounded, and satifying $\phi(\sigma(q_1(t)), q_1(t)) = 0$.

Premultiplying both sides of (2) by $L^T(q^1)$, we get

$$\begin{aligned} A_L(q^1)\ddot{q}^1 + L^T(q^1)C_L(q^1,\dot{q}^1)\dot{q}^1 + \\ + L^T(q^1)G(q^1) = L^T(q^1)(\tau + \tau_d) \end{aligned}$$

where $A_L(q^1) = L^T(q^1)M(q^1)L(q^1)$. Observe that $L^T(q^1)$ is formulated such that

$$L^T(q^1)J^T(q) = 0. \tag{4}$$

Then the error dynamic equations can be obtained as

$$\begin{aligned} \dot{\bar{x}}_1 &= -p\bar{x}_1 + \bar{x}_2 \\ A_L(q^1)\dot{\bar{x}}_2 &= -L^T(q^1)C_L(q^1,\dot{q}^1)\bar{x}_2 + L^T(q^1)(-F(x_e) + \tau + \tau_d) \end{aligned} \tag{5}$$

where $x_e \doteq [(q^1)^T \quad (\dot{q}^1)^T \quad (q^1)^T_d \quad (\dot{q}^1)^T_d \quad (\ddot{q}^1)^T_d]^T$ and

$$F(x_e) \doteq M(q^1)L(q^1)(\ddot{q}^1_d - p\dot{\bar{x}}_1) + C_L(q^1,\dot{q}^1)(\dot{q}^1_d - p\bar{x}_1) + G(q^1).$$

A parametric uncertainty can be introduced in (1) dividing the parameter matrices $M(q)$, $C(q,\dot{q})$, and $G(q)$ into a nominal and a perturbed part

$$\begin{aligned} M(q) &= M_0(q) + \Delta M(q) \\ C(q,\dot{q}) &= C_0(q,\dot{q}) + \Delta C(q,\dot{q}) \\ G(q) &= G_0(q) + \Delta G(q) \end{aligned}$$

where $M_0(q)$, $C_0(q,\dot{q})$, and $G_0(q)$ are the nominal matrices and $\Delta M(q)$, $\Delta C(q,\dot{q})$, and $\Delta G(q)$ are the parametric uncertainties. Then, $F(x_e)$ can be expressed as

$$F(x_e) = F_0(x_e) + \Delta F(x_e) \tag{6}$$

where $F_0(x_e)$ is the nominal part of $F(x_e)$ computed with $M_0(q)$, $C_0(q,\dot{q})$, and $G_0(q)$ and $\Delta F(x_e)$ is the uncertain part computed with $\Delta M(q)$, $\Delta C(q,\dot{q})$, and $\Delta G(q)$.

An adaptive neural network, $F(x_e,\Theta)$, is used to estimate the term $\Delta F(x_e)$ in (6), where Θ is a vector containing the tunable network parameters. The procedure to adjust the neural network, presented in this section, was developed in [3] and [1]. Define n neural networks $F_k(x_e,\Theta_k)$, $k = 1,\cdots,n$ composed of nonlinear neurons in every hidden layers and linear neurons in the input and output layers, with the adjustable weights Θ_k in the output layers. The single-output neural networks are of the form

$$\begin{aligned} F_k(x_e,\Theta_k) &= \sum_{i=1}^{p_k} H\left(\sum_{j=1}^{5n} w^k_{ij}x_{ej} + m^k_i\right)\Theta_{ki} \\ &= \xi_k^T\Theta_k \end{aligned} \tag{7}$$

with

$$\xi_k = \begin{bmatrix} H\left(\sum_{j=1}^{5n} w^k_{1j}x_{ej} + m^k_1\right) \\ \vdots \\ H\left(\sum_{j=1}^{5n} w^k_{p_kj}x_{ej} + m^k_{p_k}\right) \end{bmatrix}$$

and

$$\Theta_k = \begin{bmatrix} \Theta_{k1} \\ \vdots \\ \Theta_{kp_k} \end{bmatrix}$$

where p_k is the number of neurons in the hidden layers, the weights w^k_{ij} and the biases m^k_i for $1 \le i \le p_k$, $1 \le j \le 5n$ and $1 \le k \le n$ are assumed to be constant and specified by the designer and $H(.)$ is the hyperbolic tangent function

$$H(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}.$$

The complete neural network can be denoted by

$$\begin{aligned} F(x_e,\Theta) &= \begin{bmatrix} F_1(x_e,\Theta_1) \\ \vdots \\ F_n(x_e,\Theta_n) \end{bmatrix} = \begin{bmatrix} \xi_1^T\Theta_1 \\ \vdots \\ \xi_n^T\Theta_n \end{bmatrix} \\ &= \begin{bmatrix} \xi_1^T & 0 & \dots & 0 \\ 0 & \xi_2^T & \vdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \xi_n^T \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \\ \vdots \\ \Theta_n \end{bmatrix} \\ &= \Xi\Theta. \end{aligned} \tag{8}$$

The following assumptions, defined in [1], are used to guarantee the control law stability based on neural network developed in Theorem 3.1:

1) There exists a parameter value $\Theta^\star \in \Omega_\theta$ such that $F(x_e,\Theta^\star)$ can approximate $\Delta F(x_e)$ as close as possible, where Ω_θ is a pre-assigned constraint region defined as

$$\Omega_\theta \doteq \{\Theta| \ \ \Theta^T\Theta \leq M_\theta, \ M_\theta > 0\}.$$

2) There exists a function $k(x_e) > 0$ such that $|(\delta F(x_e))_i| \leq k(x_e)$, for all $1 \leq i \leq n$, where $\delta F(x_e) = \Delta F(x_e,\Theta^\star) - \Delta F(x_e)$.

And the following definitions are made

$$\lambda_c \doteq \lambda_d - k_\lambda(\lambda - \lambda_d) \quad \text{ and } \quad E \doteq \begin{bmatrix} I_{(n-m)\times(n-m)} \\ 0_{m\times(n-m)} \end{bmatrix} \quad (9)$$

for some $k_\lambda > 0$, where $\lambda_d(t)$ is a desired multiplier related to a desired constrained force $f_d(t)$, that is, $f_d(t) = J^T(q_d(t))\lambda_d(t)$.

Theorem 3.1: Consider the reduced model (2) with plant uncertainties and external disturbances, and desired reference trajectories $q_d(t)$ and $\lambda_d(t)$. The adaptive neural network-based controller given by

$$\dot{\Theta} = \begin{cases} -\rho\Xi^T L\bar{x}_2 & if \ \|\Theta\| < M_\theta \ \ or \\ & (\|\Theta\| = M_\theta \ \ and \ \ \bar{x}_2^T L^T \Xi\Theta \geq 0) \\ -\rho\Xi^T L\bar{x}_2 + \rho\frac{\bar{x}_2^T L^T \Xi\Theta}{\|\Theta\|^2}\Theta & if \ \ \|\Theta\| = M_\theta \\ & and \ \ \bar{x}_2^T L^T \Xi\Theta < 0 \end{cases} \quad (10)$$

$$\tau = F_0(x_e) + \Xi\Theta - k_0 E\bar{x}_2 - k(x_e)sgn(L\bar{x}_2) - J^T\lambda_c \quad (11)$$

($(q(0),\dot{q}(0))$ are considered bounded and satisfy $\Phi(q(0)) = 0$ and $J(q(0)\dot{q}(0) = 0)$ achieves the following performance for a suitable choice of the constant gain k_0:

(1) $\Theta(t) \in \Omega_\theta$ and all the variables $q(t)$, $\dot{q}(t)$ and $\tau(t)$ are bounded for all $t \geq 0$.

(2) The following $\mathscr{H}_\infty$ performance holds

$$\int_0^T \|\bar{x}(t)\|_Q^2 \leq V(0) + \gamma^2 \int_0^T \|\tau_d(t)\|^2, \quad \forall T \geq 0 \quad (12)$$

with $\tau_d(.) \in L_2[0,\infty)$, where Q is a weighting matrix and γ is the attenuation level.

(3) The steady-state force error $\lambda - \lambda_d$ is inversely proportional to the value of $k_\lambda + 1$.

(4) If $\tau_d(.) \in L_2[0,\infty) \cap L_\infty[0,\infty)$, then we can conclude that $lim_{t\to\infty}(q^1(t) - q_d^1(t)) = 0$ and $lim_{t\to\infty}(\dot{q}^1(t) - \dot{q}_d^1(t)) = 0$.

Proof. This proof follows the line of the proof presented in [2] for adaptive tracking control, where the linear parametrization property is used instead of the adaptive neural network approach. The Lyapunov function candidate is chosen as

$$V(t,\bar{x},\tilde{\Theta}) = \frac{\alpha}{2}\bar{x}_1^T\bar{x}_1 + \frac{1}{2}\bar{x}_2^T A_L \bar{x}_2 + \frac{1}{2\rho}\tilde{\Theta}^T\tilde{\Theta} \quad (13)$$

for some $\alpha > 0$, where $\tilde{\Theta} \doteq \Theta - \Theta^*$.

Taking into account the control law (11) and using the property of skew-symmetry of $\dot{A}_L - 2L^T C_L \bar{x}_2$, the derivative of $V(t,\bar{x},\tilde{\Theta})$ along (5) is given as

$$\dot{V}(t,\bar{x},\tilde{\Theta}) = -\alpha p \bar{x}_1^T\bar{x}_1 + \alpha\bar{x}_1^T\bar{x}_2 - k_0\bar{x}_2^T\bar{x}_2 + \bar{x}_2^T L^T\tau_d$$
$$+ (\bar{x}_2^T L^T\Xi + \frac{1}{\rho}\dot{\tilde{\Theta}}^T)\tilde{\Theta} + \bar{x}_2^T L^T(-k(x_e)sgn(L\bar{x}_2) + \delta F(x_e)) \quad (14)$$

The above equation is different from that one presented in [2], since the neural network matrix Ξ is used in place of the regression matrix Y and the VSC term $(-k(x_e)sgn(L\bar{x}_2))$ is added to eliminate the limitation on the approximation error discussed in [1]. From the definition of the update law (10), which is a standard projection algorithm [5], it can be conclude that $(\bar{x}_2^T L^T\Xi + \frac{1}{\rho}\dot{\tilde{\Theta}}^T)\tilde{\Theta} \leq 0$ and $\Theta(t) \in \Omega_\theta$, for all $t \geq 0$ if $\Theta(0) \in \Omega_\theta$.

Considering Assumption (2), it can be guaranteed that

$$\bar{x}_2^T L^T(-k(x_e)sgn(L\bar{x}_2) + \delta F(x_e))$$
$$\leq -k(x_e)\sum_{i=1}^n |(L\bar{x}_2)_i| + \sum_{i=1}^n |(\delta F(x_e)))_i|\,|(L\bar{x}_2)_i| \leq 0$$

Hence, considering the constrains obtained in [2] for k_0 and p, (14) becomes

$$\dot{V} \leq -\|\bar{x}\|_Q^2 + \gamma^2\|\tau_d\|^2. \quad (15)$$

Integrating from $t = 0$ to $t = T$, since $V(T,\bar{x}(T),\tilde{\Theta}(T)) \geq 0$, the above inequality leads to

$$\int_0^T \|\bar{x}\|_Q^2 dt \leq V(0,\bar{x}(0),\tilde{\Theta}(0)) + \gamma^2 \int_0^T \|\tau_d\|^2 dt.$$

Hence, the $\mathscr{H}_\infty$ performance is achieved. The rest of the proof follows [2]. ◇

IV. RESULTS

In this section we are going to show the experimental results obtained with the application of the controller developed in Section III on our experimental manipulator UArm II (Underactuated Arm II), designed and built by H. Ben Brown, Jr. of Pittsburgh, PA, USA [7]. This 3-link planar manipulator contains in each joint a DC motor, a break and an optical encoders with quadrature decoding used to measure the joint positions, Fig. 1. Joint velocities are obtained by numerical differentiation and filtering. The manipulator kinematic and dynamic nominal parameters, which are used to compute the term $F_0(x_e)$, are shown in Table I. The matrices $M_0(q)$ and $C_0(q,\dot{q})$ can be seen in the Appendix.

TABLE I
ROBOT PARAMETERS.

i	m_i	I_i	l_i	lc_i	f_i
	(kg)	(kgm^2)	(m)	(m)	(kgm^2/s)
1	0.850	0.0153	0.203	0.096	0.25
2	0.850	0.0100	0.203	0.096	0.10
3	0.625	0.0100	0.203	0.077	0.10

Since the forces between the robot end-effector and the environment are used in the control law, a force sensor device was designed and built to measure the normal and tangential

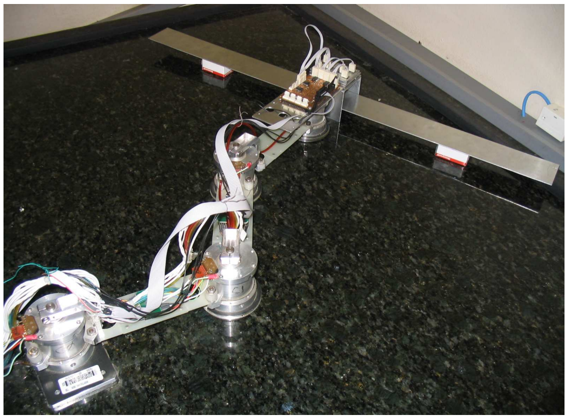

Fig. 1. Manipulator UArm II.

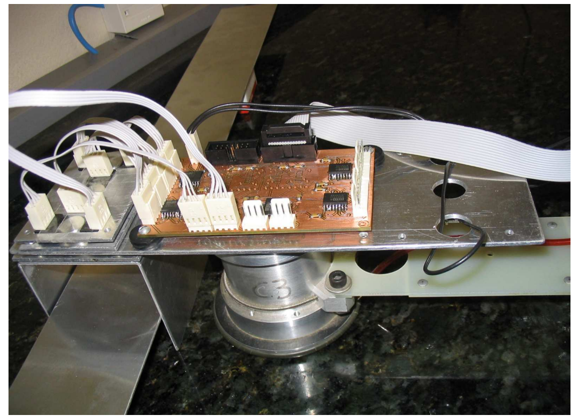

Fig. 3. Force Sensor Device coupled to UArm II end-effector.

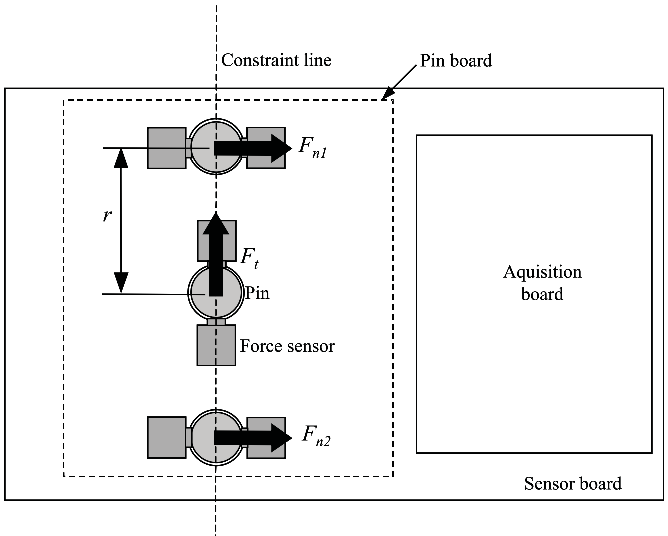

Fig. 2. Force Sensor Device.

forces and the z-direction moment applied in the UArm II end-effector by the constraint surface described in the following. The device has an aluminum-made sensor board where six FSG-15N1A force sensors from Honeywell and the acquisition board are fixed. As the FS series force sensor measures only positive force values, two sensors are disposed in an opposite way for each direction to measure positive and negative values, see Fig. 2. Two force sensor pairs are used to measure the normal force and the z-direction moment, given, respectively, by $F_n = F_{n1} + F_{n2}$ and $M_z = (F_{n2} - F_{n1})r$. If only translational movement is performed on the normal direction, $F_{n1} = F_{n2}$, and the z-direction moment is zero. Otherwise, if only rotational movement is performed, $F_{n1} = -F_{n2}$ and the normal force is zero. Figure 3 shows the experimental manipulator UArm II with the force sensor device coupled.

The constraint surface for the robot end-effector is a segment of a straight line on the X-Y plane, with end-effector orientation perpendicular to the constraint line, that is, the orientation must remain in a constant value c given by the line inclination β. The equations of the $m = 2$ constraints are

$$\phi(q) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} - l_3 s_{123} + \beta[l_1 c_1 + l_2 c_{12} + l_3 c_{123}] + b \\ q_1 + q_2 + q_3 - c \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

where b is the linear coefficient of the constraint line, $s_{i\ldots k} = sin(q_i + \ldots + q_k)$, $c_{i\ldots k} = cos(q_i + \ldots + q_k)$, q_i is the angular position of joint i, and l_i is the length of the i-link. Hence, $\phi : \Re^3 \to \Re^2$, and the Jacobian matrix, $J(q) = \partial\phi/\partial q$, is given as

$$J = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \end{bmatrix}$$

with

$$J_{11} = -l_1 c_1 - l_2 c_{12} - l_3 c_{123} - \beta[l_1 s_1 + l_2 s_{12} + l_3 s_{123}]$$
$$J_{12} = -l_2 c_{12} - l_3 c_{123} - \beta[l_2 s_{12} + l_3 s_{123}]$$
$$J_{13} = -l_3 c_{123} - \beta[l_3 s_{123}]$$
$$J_{21} = J_{22} = J_{23} = 1.$$

Defining $q^1 = [q_1]$ and $q^2 = [q_2 \;\; q_3]$, the matrix $L(q)$ of the constraint line is

$$L(q) = \begin{bmatrix} 1 \\ -\frac{[\cos(q_1)+\cos(q_1+q_2)+\beta(\sin(q_1)+\sin(q_1+q_2))]}{[\cos(q_1+q_2)+\beta\sin(q_1+q_2)]} \\ \frac{[\cos(q_1)+\cos(q_1+q_2)+\beta(\sin(q_1)+\sin(q_1+q_2))]}{[\cos(q_1+q_2)+\beta\sin(q_1+q_2)]} - 1 \end{bmatrix}.$$

Observe that $l_1 = l_2 = l_3$ and $J(q)^T L(q)^T = 0$. The initial and final coordinates of the movement are $(x_0, y_0) = (0.45, 0.30)$ m and $(x(T), y(T)) = (0.49, 0.22)$ m, respectively. In this case, $\beta = -2$, $b = 1.2$, and $c = 26.6°$. The reference trajectory for the joint variables $q_d(t)$ is a fifth-degree polynomial, with trajectory duration time $T = 4.0$ s. It is desired that no force acts on the normal direction of the constrain line and no moment acts on the z-direction, that is, $\lambda_d = [(F_n)_d \;\; (M_z)_d]^T = [0 \;\; 0]^T$.

The level of attenuation and the weighting matrix Q defined for the controller are $\gamma = 3$ and $Q = 0.05 I_2$, respectively. The selected control gains are $p = 0.2$, $k_0 = 1.5$, $k_\lambda = 0.8$, and $\rho = 45$, with $M_\theta = 0.1$. The function $k(x_e)$ is defined as $k(x_e) = 2\sqrt{\bar{x}_1^2 + \bar{x}_2^2}$.

The following auxiliary variable is defined for computation of $F_k(x_e, \Theta_k)$

$$xx = \sum_{i=1}^{n} (q_i - q_i^d) + \sum_{i=1}^{n} (\dot{q}_i - \dot{q}_i^d) - \sum_{i=1}^{n} \ddot{q}_i. \quad (16)$$

The matrix Ξ can be computed as

$$\Xi = \begin{bmatrix} \xi_1^T & 0 & 0 \\ 0 & \xi_2^T & 0 \\ 0 & 0 & \xi_3^T \end{bmatrix}$$

with

$$\begin{aligned} \xi_1 &= [\xi_{11}, \cdots, \xi_{17}]^T, \\ \xi_2 &= [\xi_{21}, \cdots, \xi_{27}]^T, \\ \xi_3 &= [\xi_{31}, \cdots, \xi_{37}]^T \end{aligned}$$

and

$$\begin{aligned} \xi_{11} = \xi_{21} = \xi_{31} &= \frac{e^{xx-1.5} - e^{-xx+1.5}}{e^{xx-1.5} + e^{-xx+1.5}} \\ \xi_{12} = \xi_{22} = \xi_{32} &= \frac{e^{xx-1} - e^{-xx+1}}{e^{xx-1} + e^{-xx+1}} \\ \xi_{13} = \xi_{23} = \xi_{33} &= \frac{e^{xx-0.5} - e^{-xx+0.5}}{e^{xx-0.5} + e^{-xx+0.5}} \\ \xi_{14} = \xi_{24} = \xi_{34} &= \frac{e^{xx} - e^{-xx}}{e^{xx} + e^{-xx}} \\ \xi_{15} = \xi_{25} = \xi_{35} &= \frac{e^{xx+0.5} - e^{-xx-0.5}}{e^{xx+0.5} + e^{-xx-0.5}} \\ \xi_{16} = \xi_{26} = \xi_{36} &= \frac{e^{xx+1} - e^{-xx-2}}{e^{xx+2} + e^{-xx-2}} \\ \xi_{17} = \xi_{27} = \xi_{37} &= \frac{e^{xx+1.5} - e^{-xx-1.5}}{e^{xx+1.5} + e^{-xx-1.5}}. \end{aligned}$$

Note that, with these definitions, 7 neurons in the hidden layer are selected for the neural networks with the weights w_{ij}^k assuming the values 1 or -1 and the biases m_i assuming the values -1.5, -1, -0.5, 0, 0.5, 1, 1.5. The network parameters Θ are defined by

$$\Theta = \begin{bmatrix} \Theta_1 \\ \Theta_2 \\ \Theta_3 \end{bmatrix}$$

with

$$\begin{aligned} \Theta_1 &= [\Theta_{11}\ \Theta_{12}\ \Theta_{13}\ \Theta_{14}\ \Theta_{15}\ \Theta_{16}\ \Theta_{17}]^T \\ \Theta_2 &= [\Theta_{21}\ \Theta_{22}\ \Theta_{23}\ \Theta_{24}\ \Theta_{25}\ \Theta_{26}\ \Theta_{27}]^T \\ \Theta_3 &= [\Theta_{31}\ \Theta_{32}\ \Theta_{33}\ \Theta_{34}\ \Theta_{35}\ \Theta_{36}\ \Theta_{37}]^T. \end{aligned}$$

The experimental results (joint position and torque) for the $\mathscr{H}_\infty$ control are shown in Figures 4 and 5, respectively. Figure 6 shows the measured normal force and moment at the robot end-effector. Finally, the end-effector $X - Y$ position is shown in Figure 7.

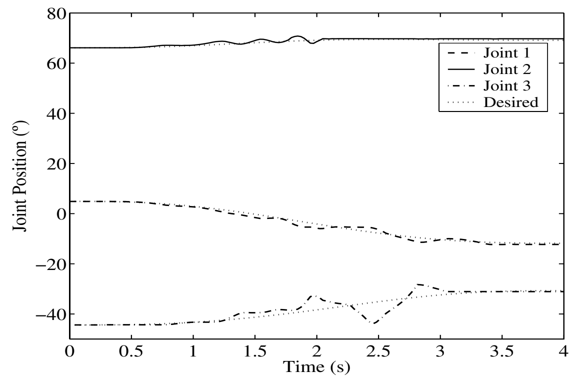

Fig. 4. Joint Position (^{0}).

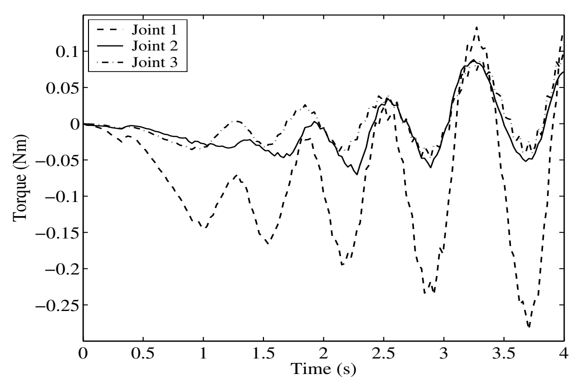

Fig. 5. Torque (Nm).

V. CONCLUSIONS

The most important contribution of this paper are the experimental results obtained with a constrained robot manipulator subject to parametric uncertainties and external disturbances. An adaptive neural network-based control with tracking performance is developed to attenuate the effects of the disturbances on the tracking errors. Also, a force control is employed to track the constraint forces acting at the robot end-effector to a desired value. The force sensor device developed in this paper, that was built in our laboratory, presents simple design and structure, that can be an economical advantage in comparison with available force sensors.

REFERENCES

[1] Y. C. Chang, Neural network-based $\mathscr{H}_\infty$ tracking control for robotic systems, *IEE Proceedings on Control Theory Applications*, vol. 147, no. 3, 2000, pp 303-311.

[2] Y. C. Chang and B. S. Chen, Adaptive Tracking Control Design of Constrained Robot Systems, *International Journal of Adaptive Control Signal Processing*, vol. 12, 1998, pp 495-526.

[3] Y. C. Chang and B. S. Chen, A Nonlinear Adaptive $\mathscr{H}_\infty$ Tracking Control Design in Robotic Systems via Neural Networks, *IEEE Transactions on Control Systems Technology*, vol. 5, 1997, pp 13-28.

[4] Y. C. Chang and B. S. Chen, Robust Tracking Designs for Both Holonomic and Nonholonomic Constrained Mechanical Systems: Adaptive Fuzzy Approach, *IEEE Transactions on Fuzzy Systems* , vol. 8, 2000, pp 46-66.

[5] H. K. Khalil, Adaptive ooutput feedback control of nonlinear systems represented by input-output models, *IEEE Transaction on Automatic Control* , vol. 41, 1996, pp 177-188.

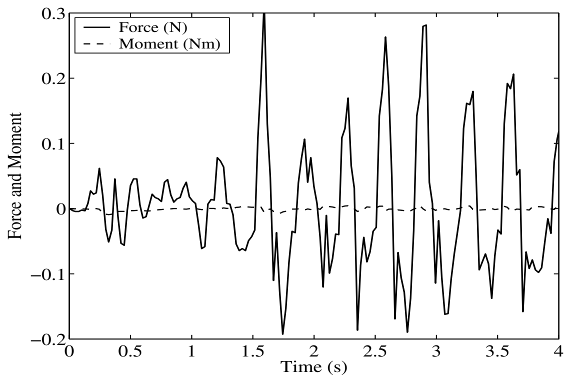

Fig. 6. Force and Moment Applied at the robot end-effector.

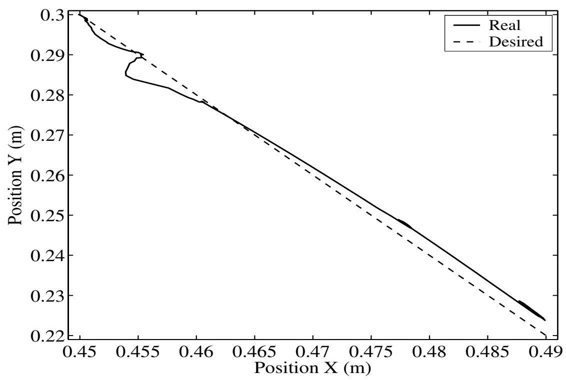

Fig. 7. Position X, Y of the robot end-effector.

[6] A. A. G. Siqueira, A. Petronilho and M. H. Terra, Adaptive nonlinear $\mathscr{H}_\infty$ techniques applied to a robot manipulator, *Proceedings of the IEEE Conference on Control Applications* , Istambul, Turquia, 2003, pp 218-223.

[7] A. A. G. Siqueira and M. H. Terra. Nonlinear and Markovian $\mathscr{H}_\infty$ controls of underactuated manipulators. *IEEE Transactions on Control Systems Technology*, v. 12, n. 6, 2004, p. 811-826.

[8] C. Y. Su, Y. Stepanenko and T. P. Leung. Combined adaptive and varible strucuture control of constrained robots, *Automatica* , vol. 31, 1995, pp 483-488.

APPENDIX

The matrices $M_0(q)$ and $C_0(q,\dot{q})$ for the planar manipulator are given by:

$$M_0(q) = \begin{bmatrix} M_{11}(q) & M_{12}(q) & M_{13}(q) \\ M_{21}(q) & M_{22}(q) & M_{23}(q) \\ M_{31}(q) & M_{32}(q) & M_{33}(q) \end{bmatrix}$$

$$\begin{aligned}
M_{11}(q) &= m_1 l_{c_1}^2 + m_2(l_1^2 + l_{c_2}^2 + 2 l_1 l_{c_2} cos(q_2)) \\
&\quad + m_3(l_1^2 + l_2^2 + l_{c_3}^2 + 2 l_1 l_2 cos(q_2) + 2 l_2 l_{c_3} cos(q_3)) \\
&\quad + 2 m_3 l_1 l_{c_3} cos(q_2+q_3) + I_1 + I_2 + I_3, \\
M_{12}(q) &= m_2(l_{c_2}^2 + l_1 l_{c_2} cos(q_2)) + m_3(l_2^2 + l_{c_3}^2 \\
&\quad + l_1 l_2 cos(q_2)) + m_3(l_1 l_{c_3} cos(q_2+q_3) \\
&\quad + 2 l_2 l_{c_3} cos(q_3)) + I_2 + I_3, \\
M_{13}(q) &= I_3 + m_3(l_{c_3}^2 + l_1 l_{c_3} cos(q_2+q_3) + l_2 l_{c_3} cos(q_3)), \\
M_{21}(q) &= M_{12}(q), \\
M_{22}(q) &= I_2 + I_3 + m_2(l_{c_2}^2) + m_3(l_2^2 + l_{c_3}^2 + 2 l_2 l_{c_3} cos(q_3)), \\
M_{23}(q) &= I_3 + m_3(l_{c_3}^2 + l_2 l_{c_3} cos(q_3)), \\
M_{31}(q) &= M_{13}(q), \quad M_{32}(q) = M_{23}(q), \\
M_{33}(q) &= I_3 + m_3(l_{c_3}^2),
\end{aligned}$$

and

$$C_0(q,\dot{q}) = \begin{bmatrix} C_{11}(q,\dot{q}) & C_{12}(q,\dot{q}) & C_{13}(q,\dot{q}) \\ C_{21}(q,\dot{q}) & C_{22}(q,\dot{q}) & C_{23}(q,\dot{q}) \\ C_{31}(q,\dot{q}) & C_{32}(q,\dot{q}) & C_{33}(q,\dot{q}) \end{bmatrix},$$

$$\begin{aligned}
C_{11}(q,\dot{q}) &= -[(m_2 l_1 l_{c_2} sin(q_2) + m_3 l_1 l_2 sin(q_2) \\
&\quad + m_3 l_1 l_{c_3} sin(q_2+q_3))\dot{q}_2 + (m_3 l_1 l_{c_3} sin(q_2+q_3) \\
&\quad + m_3 l_2 l_{c_3} sin(q_3))\dot{q}_3], \\
C_{12}(q,\dot{q}) &= -[(m_2 l_1 l_{c_2} sin(q_2) + m_3 l_1 l_2 sin(q_2) \\
&\quad + m_3 l_1 l_{c_3} sin(q_2+q_3))(\dot{q}_1 + \dot{q}_2) \\
&\quad + (m_3 l_1 l_{c_3} sin(q_2+q_3) + m_3 l_2 l_{c_3} sin(q_3))\dot{q}_3], \\
C_{13}(q,\dot{q}) &= -[(m_3 l_1 l_{c_3} sin(q_2+q_3) + m_3 l_2 l_{c_3} sin(q_3)) \\
&\quad (\dot{q}_1 + \dot{q}_2 + \dot{q}_3)], \\
C_{21}(q,\dot{q}) &= (m_2 l_1 l_{c_2} sin(q_2) + m_3 l_1 l_2 sin(q_2) \\
&\quad + m_3 l_1 l_{c_3} sin(q_2+q_3))\dot{q}_1 - m_3 l_2 l_{c_3} sin(q_3)\dot{q}_3, \\
C_{22}(q,\dot{q}) &= -m_3 l_2 l_{c_3} sin(q_3)\dot{q}_3, \\
C_{23}(q,\dot{q}) &= -m_3 l_2 l_{c_3} sin(q_3)(\dot{q}_1 + \dot{q}_2 + \dot{q}_3), \\
C_{31}(q,\dot{q}) &= (m_3 l_1 l_{c_3} sin(q_2+q_3) + m_3 l_2 l_{c_3} sin(q_3))\dot{q}_1 \\
&\quad + m_3 l_2 l_{c_3} sin(q_3)\dot{q}_2, \\
C_{32}(q,\dot{q}) &= m_3 l_2 l_{c_3} sin(q_3)(\dot{q}_1 + \dot{q}_2), \\
C_{33}(q,\dot{q}) &= 0,
\end{aligned}$$

where m_i, l_i, l_{c_i}, I_i, q_i and $\dot{q}_i$, are the mass, the length, the center of mass, the inertia momentum, the angular position and the angular velocity of the i-link, respectively.