15th Triennial World Congress, Barcelona, Spain

RECURRENT NEURAL CONTROL FOR ROBOT TRAJECTORY TRACKING

Edgar N. Sanchez * **Jose P. Perez** ** **Luis J. Ricalde** *

* *CINVESTAV, Unidad Guadalajara, Apartado Postal 31-430, Plaza La Luna, Guadalajara, Jalisco C.P. 45091, Mexico, e-mail: sanchez@gdl.cinvestav.mx*
** *CINVESTAV, Unidad Guadalajara, on doctoral studies leave from, Universidad Autonoma de Nuevo Leon, Monterrey, Nuevo Leon, Mexico*

Abstract: This paper extends the results previously obtained for trajectory tracking of unknown plants using recurrent neural networks. The proposed controller structure is composed of a neural identifier and a control law defined by using the inverse optimal control approach, which has been improved so that less inputs than states are needed. The proposed new control scheme is applied to the control a robotic manipulator model.

Keywords: Neural networks, Trajectory tracking, Adaptive control, Lyapunov function, Stability analysis.

1. INTRODUCTION

Since the seminal paper [Narendra and Parthasarathy, 1990], there has been intensive studies in applying neural networks to identification and control of nonlinear systems. Lately, dynamic neural networks were developed, enabling more efficient modeling of complex dynamics [Poznyak *et al.,* 1999]. One recent book, [Rovitahkis and Christodoulou, 2000], has reviewed a broad spectrum of applications of dynamic neural networks for nonlinear systems identification and control. In [Rovitahkis and Christodoulou, 2000] on-line learning is employed, for adaptive identification and control, where stability of closed-loop systems is analyzed using the Lyapunov function method. In [Rovitahkis and Christodoulou, 2000], the trajectory tracking problem is studied, by means of reducing the system to a linear model following problem, in which an application to DC electric motors is discussed.

On the other hand, control methods that are applicable to general nonlinear systems have seen rapid development since the early 1980's. Main approaches include, for example, the use of differential geometry theory [Isidori, 1995]. Recently, the passivity approach has generated increasing interest for synthesizing control laws, also for general nonlinear systems [Hill and Moylan,1996]. An important problem in this approach is how to achieve robust nonlinear control in the presence of unmodelled dynamics and external disturbances. In this direction, worth mentioning is the so-called H_∞ nonlinear control approach [Basar and Bernhard, 1995]. One major difficulty with this approach today, alongside its possible system structural instability, seems to be the requirement of solving some resulting partial differential equations. In order to alleviate this computational problem, the so-called inverse optimal control technique was recently developed, based on the input-to-state stability concept [Krstic and Deng, 1998].

A control law for generating chaos in a dynamic neural network was designed in [Sanchez *et al.*, 2001(a)]. In [Sanchez *et al.*, 2001 (b)] and [Sanchez *et al.*, 2001 (c)], this methodology was modified for stabilization and trajectory tracking of an unknown chaotic dynamical system. The proposed adaptive control scheme is composed of a dynamic neural identifier and a controller, where the former is used to build an on-line model for the unknown plant and the latter, to ensure the unknown plant to track the reference trajectory. In this paper, we further improve the design by using less inputs than states in the control. The approach is based on the methodology developed in [Sanchez *et al.*, 2001(b)] and [Sanchez *et al.*, 2001(c)], in which the control law is optimal with respect to a well-defined cost functional. Trajectory tracking for a robotic manipulator model is demonstrated as an application example.

2. MATHEMATICAL PRELIMINARIES

2.1 *A High-Order Recurrent Neural Network*

In [Kosmatopoulos *et al.*,1997], high-order recurrent neural networks (HORNN) are defined as

$$\dot{x}_i = -a_i x_i + \sum_{k=1}^{L} w_{ik} \prod_{j\in I_k} y_j^{d_j(k)}, i = 1, ..., n \quad (1)$$

where x_i is the ith neuron state, L is the number of high-order connections, $\{I_1, I_2, ..., I_L\}$ is a collection of non-ordered subsets of $\{1, 2, ..., m+n\}$, $a_i > 0$, w_{ik} are the adjustable weights of the neural network, $d_j(k)$ are nonnegative integers, and y is a vector defined as $y = \left[y_1, .., y_n, y_{n+1}, ..., y_{n+m}\right]^\top$ $= [S(x_1), ..., S(x_n), S(u_1), ..., S(u_m)]^\top$,with $u = [u_1, u_2, ..., u_m]^\top$ being the input to the neural network, and $S(\cdot)$ is a smooth sigmoid function formulated by $S(x) = \frac{1}{1+\exp(-\beta x)} + \varepsilon$. For the sigmoid, β is a positive constant and ε is a small positive real number. Here, $S(x) \in [\varepsilon, \varepsilon + 1]$.

Clearly, system (1) allows the inclusion of high-order terms.By defining a new vector,

$z(x, u) = [z_1(x, u), ..., z_L(x, u)]^\top$

$= \left[\prod_{j\in I_1} y_j^{d_j(1)}, ..., \prod_{j\in I_L} y_j^{d_j(L)}\right]^\top$ system (1) can be rewritten as

$$\dot{x}_i = -a_i x_i + \sum_{k=1}^{L} w_{ik} z_k(x, u), \qquad i = 1, ..., n$$

$$\dot{x}_i = -a_i x_i + w_i^\top z_i(x, u), \quad i = 1, ..., n \quad (2)$$

where $w_i = [w_{i,1} ... w_{i,L}]^\top$

In this paper, we assume that the input enters the neural network directly. We are particularly interested in less inputs than states, since this is a more efficient structure. For this purpose, we rewrite system (1) as

$$\dot{x}_i = -a_i x_i + w_i^\top z_i(x) + \Omega u_i, \qquad \Omega \in R^{nxm} \quad (3)$$

We reformulate (2) in a matrix form and obtain

$$\dot{x} = Ax + Wz(x) + \Omega u \quad (4)$$

where $x \in \Re^n$, $W^* \in \Re^{n\times L}$, $z(x) \in \Re^L$, $u \in \Re^n$, and $A = -\lambda I$, with $\lambda > 0$. In the following, we use the following slight modification of the HORNN:

$$\dot{x} = Ax + W\Gamma z(x) + \Omega u$$

where $\Gamma \in \Re^{L\times L}$ is a diagonal matrix defined by

$$\Gamma = diag\left\{\gamma_1, \gamma_2, ..., \gamma_{L-1}, \gamma_L\right\}$$

3. MODELLING THE UNKNOWN PLANT

The unknown nonlinear plant is modeled as

$$x_p = f_p(x_p) + g_p(x_p)u \quad (5)$$

We propose to modelize the unknown nonlinear plant by a dynamic neural network, in the following form:

$$\begin{aligned}\dot{x}_p &= \dot{x} + w_{per} \\ &= Ax + W^*\Gamma z(x) + (x - x_p) + \Omega u\end{aligned} \quad (6)$$

where $x_p, x, z(x) \in \Re^n$, $W^*, \Gamma \in \Re^{n\times n}$, $u \in \Re^{mx1}$, $\Omega \in R^{nxm}$, and $w_{per} = x - x_p$ represents the modelling error.W^* are the unknown values of the neural network which minimize the modelling error.

4. TRAJECTORY TRACKING ANALYSIS

We proceed to analize the tracking error between the unknown plant modeled by (6) and the reference trajectory given by

$$\dot{x}_r = f_r(x_r, u_r), \;\; x_r \in \Re^n \tag{7}$$

we define the tracking error and its time derivative as

$$\begin{aligned} e &= x_p - x_r & (8)\\ \dot{e} &= \dot{x}_p - \dot{x}_r & (9)\\ &= Ax + W^*\Gamma z(x) + w_{per} \\ &\quad +\Omega\, u - f_r(x_r, u_r) \end{aligned}$$

Now, we proceed to add and subtract the terms $\hat{W}\Gamma z(x_r)$, Ae, and $\Omega\alpha_r(t,\hat{W})$, so that

$$\begin{aligned} \dot{e} &= Ae + W^*\Gamma z(x) + \Omega u + \\ &\quad (-f_r(x_r,u_r) + Ax_r + \hat{W}\Gamma z(x_r) + x_r - x_p \\ &\quad +\Omega\alpha_r(t,\hat{W})) - Ae - \hat{W}\Gamma z(x_r) & (10)\\ &\quad -\Omega\alpha_r(t,\hat{W}) - Ax_r - x_r + x + Ax \end{aligned}$$

Note that the plant will track the reference signal even in the presence of uncertainties, if there exists a function $\alpha_r(t,\hat{W})$ such that

$$\begin{aligned} \alpha_r(t,\hat{W}) &= \left(\Omega^T\Omega\right)^{-1}\Omega^T(f_r(x_r,u_r) - Ax_r & (11)\\ &\quad -\hat{W}\Gamma z(x_r) - (x_r - x_p)) \end{aligned}$$

Next, assume that (11) holds, and define $\tilde{W} = W^* - \hat{W}$, $\tilde{u} = u - \alpha_r(t,\hat{W})$, so that (10) is reduced to

$$\begin{aligned} \dot{e} &= Ae + \tilde{W}\Gamma z(x) + \hat{W}\Gamma\left(z(x) - z(x_r)\right) & (12)\\ &\quad +(A+I)(x - x_r) - Ae + \Omega\tilde{u} \\ &= Ae + \tilde{W}\Gamma z(x) + \hat{W}\Gamma(z(x) - z(x_p) \\ &\quad +z(x_p) - z(x_r)) \\ &\quad +(A+I)(x - x_p + x_p - x_r) - Ae + \Omega\tilde{u} \end{aligned}$$

Then, by defining

$$\tilde{u} = u_1 + u_2 \tag{13}$$

with

$$\begin{aligned} u_1 &= \left(\Omega^T\Omega\right)^{-1}\Omega^T(-\hat{W}\Gamma(z(x) - z(x_p)) & (14)\\ &\quad -(A+I)(x - x_p)) \end{aligned}$$

Equation (12) reduces to

$$\begin{aligned} \dot{e} &= Ae + \tilde{W}\Gamma z(x) + \hat{W}\Gamma z(x_p) - z(x_r) & (15)\\ &\quad +(A+I)(x_p - x_r) - Ae + \Omega u_2 \end{aligned}$$

Moreover, by taking into account that $e = x_p - x_r$, equation (15) reduces to the following equation:

$$\begin{aligned} \dot{e} &= (A+I)e + \tilde{W}\Gamma z(x) & (16)\\ &\quad +\hat{W}\Gamma(z(x_p) - z(x_r)) + \Omega u_2 \end{aligned}$$

Therefore, the tracking problem reduces to a stabilization problem for the error dynamics (16). To solve this problem, we next apply the inverse optimal control approach.

5. TRACKING ERROR STABILIZATION

Once (16) is obtained, we consider its stabilization. Note that $(e,\tilde{W}) = (0,0)$ is an equilibrium point of the undisturbed autonomous system. For stability analysis, we define the candidate Lyapunov function as

$$V = \frac{1}{2}\|e\|^2 + \frac{1}{2}tr\left\{\tilde{W}^\top\tilde{W}\right\} \tag{17}$$

Its time derivative, along the trayectories of (16), is

$$\begin{aligned} \dot{V} &= e^\top(A+I)e + e^\top\tilde{W}\Gamma z(x) & (18)\\ &\quad +e^\top\hat{W}\Gamma\left(z(x_p) - z(x_r)\right) \\ &\quad +e^\top\Omega u_2 + tr\left\{\dot{\tilde{W}}^\top\tilde{W}\right\} \end{aligned}$$

As in [Rovitahkis and Christodoulou, 2000], we propose the following learning law:

$$tr\left\{\dot{\tilde{W}}^\top\tilde{W}\right\} = -e^\top\tilde{W}\Gamma z(x) \tag{19}$$

$$\begin{aligned} \dot{\hat{w}}_{i,j} &= e_i(\gamma_j z(x_j)) & (20)\\ & i = 1,2,...,n, \;\; j = 1,2,...,L. \end{aligned}$$

Then, we substitute (19) into (18), to obtain

$$\begin{aligned} \dot{V} &= -(\lambda - 1)e^\top e & (21)\\ &\quad +e^\top\hat{W}\Gamma\left(z(x_p) - z(x_r)\right) \\ &\quad +e^\top\Omega u_2 \\ L_fV &= -(\lambda-1)e^\top e + e^\top\hat{W}\Gamma\left(z(x_p) - z(x_r)\right) \\ L_gV &= e^\top\Omega \end{aligned}$$

Next, we consider the following inequality, proved in [Poznyak, *et al.* 1999]:

$$X^\top Y + Y^\top X \leq X^\top\Lambda X + Y^\top\Lambda^{-1}Y \tag{22}$$

which holds for all matrices $X, Y \in \Re^{nxk}$ and $\Lambda \in \Re^{nxn}$ with $\Lambda = \Lambda^\top > 0$. Applying it to the second term of (21), we obtain

$$\dot{V} \leq -(\lambda - 1)e^\top e + \frac{1}{2}e^\top e + \frac{1}{2}\|\hat{W}\|^2 \|\Gamma\|^2 \|z(x_p) - z(x_r)\|^2 + e^\top \Omega u_2$$

where $\|\hat{W}\|$, $\|\Gamma\|$ are the Euclidean norm of $\hat{W}$, Γ, respectively. Taking in account $x_p = e + x_r$ and denoting $\phi_z = z(x_p) - z(x_r) = z(e + x_r) - z(x_r)$, we have

$$\begin{aligned} \dot{V} \leq & -(\lambda - 1)e^\top e + \frac{1}{2}e^\top e \\ & + \frac{1}{2}L_{\phi z}^2 \|\hat{W}\|^2 \|\Gamma\|^2 \|e\|^2 \\ & + e^\top \Omega u_2 \\ \dot{V} \leq & -(\lambda - 1)e^\top e \\ & + e^\top \left(\frac{1}{2} + \frac{1}{2}L_{\phi_z}^2 \|\hat{W}\|^2 \|\Gamma\|^2 \right) e \\ & + e^\top \Omega u_2 \end{aligned} \tag{23}$$

where $L_{\phi z}^2$ is the Lipschitz constant of ϕ_z.

To this end, we define the following control law:

$$\begin{aligned} u_2 = & -\mu(\Omega^T\Omega)^{-1}\Omega^T(1 + L_{\phi_z}^2 \|\hat{W}\|^2 \|\Gamma\|^2)e \quad (24) \\ \mu > & \frac{1}{2} \\ \triangleq & -\beta \left(R(e, \hat{W})\right)^{-1} \left(L_g V\right)^\top \\ & \beta \left(R(e, \hat{W})\right)^{-1} \\ = & \mu \left(\Omega^T\Omega\right)^{-1} \Omega^T \left(1 + L_{\phi_z}^2 \|\hat{W}\|^2 \|\Gamma\|^2\right) \end{aligned}$$

with scalars $R(e, \hat{W}) > 0, \beta > 0$. Now, substituting (24) in (23) yields

$$\begin{aligned} \dot{V} = & -(\lambda - 1)e^\top e \\ & - \left(\mu - \frac{1}{2}\right) e^\top \left(1 + L_{\phi_z}^2 \|\hat{W}\|^2 \|\Gamma\|^2\right) e \\ < & \, 0, \qquad \forall e, \hat{W} \neq 0 \end{aligned} \tag{25}$$

Finally, the control law, which affects the plant and the neural netwok, is given by

$$\begin{aligned} u = & \tilde{u} + \alpha_r(t, \hat{W}) \\ = & \left(\Omega^T\Omega\right)^{-1} \Omega^T(-\hat{W}\Gamma(z(x) - z(x_p)) \\ & -(A + I)(x - x_p) - \mu \left(1 + L_{\phi_z}^2 \|\hat{W}\|^2 \|\Gamma\|^2\right) e \\ & + f_r(x_r, u_r) - Ax_r - \hat{W}\Gamma z(x_r) - x_r + x_p) \end{aligned} \tag{26}$$

This control law gives asymptotic stability of the error dynamics and thus ensures the tracking to the reference signal.

6. COST FUNCTIONAL OPTIMIZATION

Once the problem of finding the control control law (24) is formulated based on the inverse optimal control approach, which stabilizes (16), we can proceed to formulate a cost functional defined by

$$J(\tilde{u}) = \lim_{t \to \infty} 2\beta V + \int_0^t \left(l(e, \hat{W}) + u_2^\top R(e, \hat{W}) u_2\right) d\tau \tag{27}$$

where the Lyapunov function solves the Hamilton-Jacobi-Bellman family of partial derivative equations parametrized with $\beta > 0$ as follows:

$$l(e, \hat{W}) + 2\beta L_f V - \beta^2 L_g V R(e, \hat{W})^{-1} L_g V^\top = 0 \tag{28}$$

Note that $2\beta V$ in (27) is bounded when $t \to \infty$, since by (25) V is decreasing and bounded from below by $V(0)$. Then, $\lim_{t \to \infty} V(t)$ exists and is finite. Recall that in [Krstic and Deng.,1998], we need $l(e, \hat{W})$ to be positive define and radially unbounded with respect to e. Here, from (28) we have

$$\begin{aligned} l(e, \hat{W}) = & -2\beta L_f V \\ & + \beta^2 L_g V R(e, \hat{W})^{-1} L_g V^\top \end{aligned} \tag{29}$$

Substituting (24) into (29) and then applying (22) to the second term on the right side of $L_f V$, we have

$$\begin{aligned} l(e, \hat{W}) \geq & (\lambda - 1)\|e\|^2 \\ & + (\mu - 1) \left(1 + L_\phi^2 \|\Gamma\|^2 \|W\|^2\right) \|e\|^2 \end{aligned}$$

Since we select $\lambda > 1$ and $\mu > 1$, we know that $l(e, \hat{W})$ satisfies the condition of being positive definite and radially unbounded. Hence, (27) is a cost functional. It is easy to verify that, in (27),

$$l(e, \hat{W}, \hat{W}_1) + \tilde{u}^\top R(e, \hat{W}, \hat{W}_1)\tilde{u} = -2\beta\dot{V}$$

with optimal value $J^* = 2\beta V(0)$. This is achieved by the control law (24).

7. SIMULATION RESULTS ON ROBOTIC TRAJECTORY TRACKING

In order to test the applicability of the proposed control scheme, we consider the trajectory tracking problem for a robot manipulator model.

In the absence of friction, the dynamics of a 2-link rigid robot arm with torque control input can be written as

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

where $q = \begin{bmatrix} q_1 & q_2 \end{bmatrix}^T$, $\dot{q} = \begin{bmatrix} \dot{q}_1 & \dot{q}_2 \end{bmatrix}^T, \ddot{q} = \begin{bmatrix} \ddot{q}_1 & \ddot{q}_2 \end{bmatrix}^T$

$$D(q) = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix},$$

$$D_{11} = m_1 l_{c1}^2 + m_2\left(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos(q_2)\right) + I_1 + I_2$$

$$D_{12} = m_2\left(l_{c2}^2 + l_1 l_{c2}\cos(q_2)\right) + I_2$$

$$D_{21} = D_{12} \qquad D_{22} = m_2 l_{c2}^2 + I_2$$

$$C = \begin{bmatrix} -2m_2 l_1 l_{c2}\dot{q}_2 sen(q_2) & -m_2 l_1 l_{c2}\dot{q}_2 sen(q_2) \\ m_2 l_1 l_{c2}\dot{q}_1 sen(q_2) & 0 \end{bmatrix}$$

$$G(q) = \begin{bmatrix} G_{11} \\ G_{21} \end{bmatrix} \qquad \tau = \begin{bmatrix} \tau_1 & \tau_2 \end{bmatrix}^T$$

$$G_{11} = m_1 g l_{c1}\cos(q_1) + m_2 g\left(l_1\cos(q_1) + l_{c2}\cos(q_1+q_2)\right)$$

$$G_{21} = m_2 g l_{c2}\cos(q_1+q_2)$$

$q_1, q_2 \rightarrow$ Angular position of link i

$l_1, l_2 \rightarrow$ Length of link i

$l_{c1}, l_{c2} \rightarrow$ Position of the center of gravity of link i

$I_1, I_2 \rightarrow$ Inertia of link i

$g \rightarrow$ Gravitational acceleration ($9.81 \ m/s^2$)

where $D(q)$ is a positive definite and symmetric inertia matrix, $C(q,\dot{q})$ is the matrix containing the effect of centripedal and Coriolis forces, and $G(q)$ contains the gravitational torques.

We consider the example studied in [Llama *et al.*, 2000], here as the nonlinear unknown plant. We try to force this manipulator to track a reference signal given by

$$q_{r_1} = 1.57 + 0.78(1-e^{-2t^3}) + 0.17(1-e^{-2t^3})\sin w_1 t$$

$$q_{r_2} = 1.57 + 1.04(1-e^{-1.8t^3}) + 2.18(1-e^{-1.8t^3})\sin w_2 t$$

where w_1 and w_2 are the frequencies of the desired trajectories for link 1 and link 2, respectively. This trajectory requires large velocity and acceleration from the manipulator. In the simulation, $w_1 = 15$ rad/s and $w_2 = 3.5$ rad/s were used. Furthermore, we consider a torque pertubation applied to link 1 as a impulse train function of amplitude 10 N m. We select the initial position at $q_1 = -90°$ and $q_2 = 0°$, which correspond to the inferior equilibrium point. The parameters for this robot model used in [Llama *et al.*, 2000] are the following:

$m_1 = 23.902$	$kg.$	$m_2 = 1.285$	$kg.$
$l_1 = 0.45$	$m.$	$l_2 = 0.45$	$m.$
$l_{c1} = 0.091$	$m.$	$l_{c2} = 0.048$	$m.$
$I_1 = 1.226$	$kg.m^2$	$I_2 = 0.093$	$kg.m^2$

We use the following dynamical neural network to modelize the same system:

$$\dot{x}_p = Ax + W^*\Gamma z(x) + (x - x_p) + \Omega u$$

with $A = -150I \quad ,\Gamma = 0.5I, \quad k = 0.45, \ I \in R^{4x4}$

$$z(x) = \tanh(kx) \quad ,\Omega = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}^T$$

For the control law (26), we select $\mu = 85$. The time evolution for the angles and torques applied to the links are shown in Figs. 1 and 2. As can be seen, trajectory tracking is successful. The applied torques are below of those obtained in [Llama *et al.*, 2000], which constitutes a reference workbench.

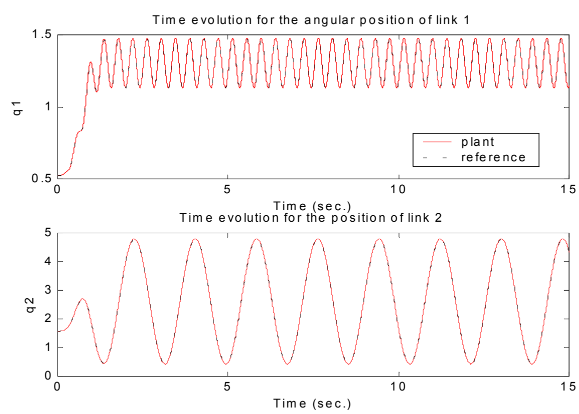

Fig. 1. Time evolution for the angular position

8. CONCLUSIONS

We have extended the designed previously developed in [Sanchez *et al.*, 2001 (b)] and [Sanchez *et al.*, 2001 (c)] for the adaptive trajectory tracking control problem, based on the inverse optimal control approach. We have further relaxed the condition

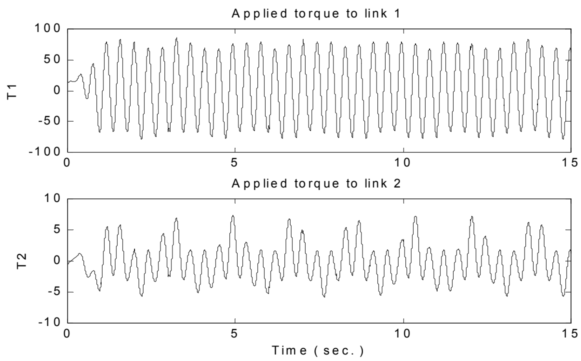

Fig. 2. Applied torques

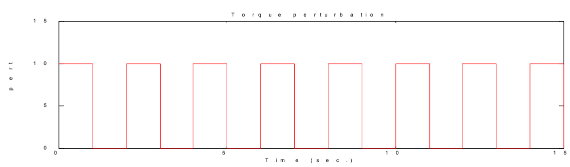

Fig. 3. Torque perturbation

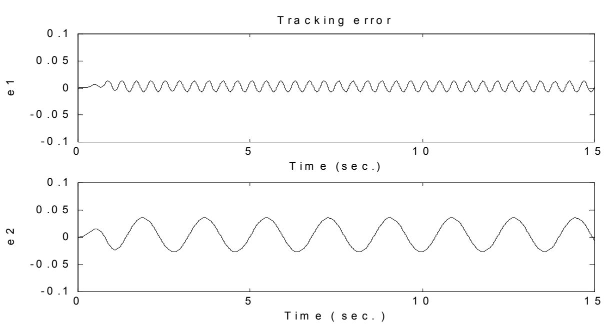

Fig. 4. Tracking error

of having the same number of inputs and states, so as to allow for less inputs than states, in the control law. The proposed scheme consists of a neural identifier and a controller, in which the former builds an on-line model of the unknown plant and the latter is optimal with respect to a meaningful cost functional. A robot model is used to verify the design for trajectory tracking, with satisfactory performance, where the plant model is assumed to be unknown. Research will continue to implement the scheme in real time and to further test it in a laboratory environment.

Acknowledgments. The authors thank the support of CONACYT, Mexico on project 32059A.

Basar T. and P. Bernhard, *H-Infinity Optimal Control and Related Minimax Design Problems*, Birkhauser, Boston, USA, (1995).

Krstic M. and H. Deng, *Stabilization of Nonlinear Uncertain Systems*, Springer Verlang, New York, USA, (1998).

Hill D. J. and P. Moylan , "The Stability of nonlinear dissipative systems", *IEEE Trans. on Auto. Contr.*, vol. 21, 708-711, (1996).

Hunt K., G. Irwin and K. Warwick (Eds.), *Neural Networks Engineering in Dynamic Control Systems,* Springer Verlag, New York, USA, (1995).

Isidori A. , *Nonlinear Control Systems*, 3rd Ed., Springer Verlag, New York, USA, (1995).

Khalil H., *Nonlinear Systems*, 2nd Ed., Prentice Hall, Upper Saddle River, NJ, USA, (1996).

Kosmatopoulos E. B. *et al.*, "Dynamical neural networks that ensure exponential identification error convergence", *Neural Networks*, vol. 1, no. 2, pp 299-314, (1997).

Llama M. A., R. Kelly, V. Santibañez, "Stable Computed torque of Robot manipulators via fuzzy self-tuning", *IEEE Trans. on Systems, Man. and Cybernetics*, Vol. 30, 143-150, (Feb. 2000)

Narendra K. S. and K. Parthasarathy, "Identification and control of dynamical systems using neural networks", *IEEE Trans. on Neural Networks*, vol. 1, no. 1, pp 4-27, (1990).

Poznyak A. S., W. Yu, E. N. Sanchez, and J. P. Perez, "Nonlinear adaptive trajectory tracking using dynamic neural networks", *IEEE Trans. on Neural Networks*, vol. 10, no 6, pp 1402-1411, (1999).

Rovitahkis G. A. and M. A. Christodoulou, *Adaptive Control with Recurrent High-Order Neural Networks*, Springer Verlag, New York, USA,(2000).

Sanchez E. N. and J. P. Perez, "Input-to-state stability analysis for dynamic neural networks", *IEEE Trans. Circuits Syst. I*, vol. 46, pp 1395-1398, (1999).

Sanchez E. N. , J. P. Perez and G. Chen, "Using dynamic neural control to generate chaos: An inverse optimal control approach", *Int. J. of Bifurcation and Chaos*, vol. 11, no. 3, pp. 857-863 (2001*a*).

Sanchez E. N. , J. P. Perez, L. Ricalde and G. Chen, "Trajectory tracking via adaptive neural control", *Proc. of IEEE Int. Symp. on Int. Cont.*, Mexico City, pp. 286-289, (2001*b*).

Sanchez E. N. , J. P. Perez, L. Ricalde and G. Chen "Chaos production and synchronization via adaptive neural control", to appear in *Proc. of IEEE Conf. on Dec. and Cont.*, Orlando, Fl, USA,(Dec. $4-7, 2001c$).