Computación y Sistemas
ISSN: 1405-5546
computacion-y-sistemas@cic.ipn.mx
Instituto Politécnico Nacional
México

Perez, Joel; Perez,, Jose P.; Rodríguez, Francisco; Flores, Angel
Trajectory Tracking for Chaos Synchronization via PI Control Law between Roosler-Chen
Computación y Sistemas, vol. 18, núm. 2, 2014, pp. 399-407
Instituto Politécnico Nacional
Distrito Federal, México

# Trajectory Tracking for Chaos Synchronization via PI Control Law between Roosler-Chen

Joel Perez, Jose P. Perez, Francisco Rodríguez, and Angel Flores

Facultad de Ciencias Físico-Matemáticas, Universidad Autónoma de Nuevo Leon (UANL), Monterrey, Mexico

joelperezp@yahoo.com, josepazp@gmail.com, francisco_rdz_rmz@hotmail.com

**Abstract.** This paper presents an application of adaptive neural networks based on a dynamic neural network to trajectory tracking of unknown nonlinear plants. The main methodologies on which the approach is based are recurrent neural networks and Lyapunov function methodology and Proportional-Integral (PI) control for nonlinear systems. The proposed controller structure is composed of a neural identifier and a control law defined by using the PI approach. The new control scheme is applied via simulations to Chaos Synchronization. Experimental results have shown the usefulness of the proposed approach for Chaos Production. To verify the analytical results, an example of a dynamical network is simulated and a theorem is proposed to ensure tracking of the nonlinear system.

**Keywords.** Dynamic neural networks, chaos production, chaos synchronization, trajectory tracking, Lyapunov function stability, PI control.

## Seguimiento de trayectorias para sincronización de caos vía ley de control PI entre Roosler-Chen

**Resumen.** Este artículo presenta la aplicación de redes neuronales adaptables, basada sobre una red neuronal dinámica, para seguimiento de trayectorias de plantas no lineales desconocidas. La principal metodología, sobre el cual la aproximación es basada, son redes neuronales recurrentes, metodología de las funciones de Lyapunov y control Proporcional-Integral (PI) para sistemas no lineales. La estructura del controlador propuesto es compuesta de un identificador neuronal y una ley de control definida usando la aproximación PI. El nuevo esquema de control es aplicado vía simulación para sincronización de caos. Resultados experimentales han mostrado la utilidad del enfoque propuesto para la producción de caos. Para verificar el resultado analítico, un ejemplo de una red dinámica es simulado y un teorema es propuesto para asegurar el seguimiento del sistema no lineal.

**Palabras clave.** Red neuronal dinámica, producción de caos, sincronización de caos, seguimiento de trayectorias, estabilidad de funciones de Lyapunov, control PI.

## 1 Introduction

Artificial neural networks as computational models of the brain are widely used in engineering applications due to their ability to estimate the relation between inputs and outputs from a learning process. Motivated by the seminal paper [1], there exists a continuously increasing interest in applying neural networks to identification and control of nonlinear systems. Most of these applications use feedforward structures [2, 3]. Recently, recurrent neural networks were developed as an extension of the static neural network capability to approximate nonlinear functions, therefore recurrent neural networks can approximate nonlinear systems. They allow more efficient modeling of the underlying dynamical systems [4].

Three representative books [5, 6, 7] reviewed the application of recurrent neural networks to nonlinear system identification and control. In particular, [5] uses off-line learning, while [6] analyzes adaptive identification and control by means of on-line learning, where stability of the closed-loop system is established based on the Lyapunov function method. In [6], the trajectory tracking problem is reduced to a linear model following problem, with application to DC electric motors. In [7], analyses of Recurrent Neural Networks for identification, estimation and control are developed, with applications


ISSN 1405-5546
http://dx.doi.org/10.13053/CyS-18-2-2014-040

to chaos control, robotics and chemical processes. Main approaches include the use of differential geometry theory [8]. Recently, the passivity approach has generated increasing interest for synthesizing control laws [9]. An important problem for these approaches is how to achieve robust nonlinear control in the presence of unmodelled dynamics and external disturbances. In this direction, there exists the so-called $H_\infty$ nonlinear control approach [10].

One major difficulty with this approach, alongside with 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 [11]. On the basis of the inverse optimal control approach, a control law for generating chaos in a recurrent neural network was designed in [12]. In [13, 14] this methodology was modified for stabilization and trajectory tracking of an unknown chaotic dynamical system. The proposed adaptive control scheme is composed of a recurrent 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 adequating it to systems with less inputs than states. The approach is based on the methodology developed in [13, 14], in which the control law is optimal with respect to a well-defined Lyapunov function.

The new approach is illustrated by Chaos Synchronization as an example of a complex dynamical system.

## 2 Modeling of the Plant

The unknown nonlinear plant is given by

$$\dot{x}_p = F_p(x_p, u) \triangleq f_p(x_p) + g_p(x_p)u, \qquad (1)$$

where $x_p$, $f_p \in R^n$, $u \in R^m$,and $g_p \in R^{n\times m}$. $x_p$ is the plant, $u$ is the control input and both $f_p$ and $g_p$ are unknown. We propose to model $(1)$ by the neural network state space representation $\dot{x}_p$ = $Ax + W^*\Gamma_z(x) + \Omega u$, plus one more modeling error term. $A$ is a $n \times m$ matrix; without loss of generality we can assume that $A = \lambda I$, $\lambda > 0$. $x$ are the neural states, giving an approximation to the real plant by a neural network, $W^*$ is the weights matrix, $\Gamma_z(x)$ is the hyperbolic tangent and $\Omega$ is $n \times m$ matrix that modifies the input $u$.

We define the modeling error between the neural network and the plant by

$$w_{per} = x - x_p. \qquad (2)$$

We assume the following.

Hypotheses 1. (Objective of Modeling): Modeling error is exponentially stable, that is,

$$\dot{w}_{per} = -kw_{per}. \qquad (3)$$

This condition guarantees that $w_{per} \to 0$ when $t \to \infty$ and it is usually found in dynamical systems that model real applications.

In this work we consider $k = 1$, and now, from (2) we have $\dot{w}_{per}$= $\dot{x} - \dot{x}_p$where: $\dot{x}_p$= $\dot{x}$+ $w_{per}$

The unknown plant can be modeled as

$$\dot{x}_p = \dot{x} + w_{per} = A(x) + W^*\Gamma_z(x) + w_{per} + \Omega u. \qquad (4)$$

$W^*$ are the fixed but unknown weights from the neural network. They minimize the modeling error.

## 3 Trajectory Tracking

We proceed now to analyze the modeling error between the unknown plant modeled by (4) and the reference signal defined by

$$\dot{x}_r = f_r(x_r, u_r), \text{ with } u_r \text{ and } x_r \in R^n, \qquad (5)$$

where $x_r$ are the reference states, $u_r$ is the input and $f_r$ is a nonlinear function.

For this purpose, we define the control error between the plant and the reference signal by

$$e = x_p - x_r, \qquad (6)$$

whose derivative with respect to time is

$$\begin{aligned} \dot{e} &= \dot{x}_p - \dot{x}_r \\ &= A(x) + W^*\Gamma_z(x) + w_{per} \\ &\quad + \Omega u - f_r(x_r, u_r). \end{aligned} \qquad (7)$$


ISSN 1405-5546
http://dx.doi.org/10.13053/CyS-18-2-2014-040

Adding and subtracting to the right hand side from (7) the terms $\widehat{W}\Gamma_z(x_r)$, $\alpha_r(t,\widehat{W})$, $Ae$, where $\widehat{W}$ is the estimate of $W^*$ and taking into account that $w_{per} = x - x_p$, we have

$$\begin{aligned}\dot{e} &= A(x) + W^*\Gamma_z(x) + x - x_p + \Omega u - f_r(x_r,u_r)\\ &+\widehat{W}\Gamma_z(x_r) - \widehat{W}\Gamma_z(x_r) + \Omega\alpha_r(t,\widehat{W})\\ &-\Omega\alpha_r(t,\widehat{W}) + Ae - Ae,\end{aligned}$$

$$\begin{aligned}\dot{e} &= Ae + W^*\Gamma_z(x) + \Omega u + (-f_r(x_r,u_r)\\ &+\widehat{W}\Gamma_z(x_r) + \Omega\alpha_r(t,\widehat{W}) - \widehat{W}\Gamma_z(x_r) \\ &-\Omega\alpha_r(t,\widehat{W}) - e - x_r - Ae + x + A(x).\end{aligned} \tag{8}$$

In this part, we consider the following supposition: the neural network will follow the reference signal, even with the presence of disturbances if

$$Ax_r + \widehat{W}\Gamma_z(x_r) + x_r - x_p + \Omega\alpha_r(t,\widehat{W}) = f_r(x_r,u_r).$$

Then

$$\Omega\alpha_r(t,\widehat{W}) = f_r(x_r,u_r) - Ax_r - \widehat{W}\Gamma_z(x_r) - x_r + x_p \tag{9}$$

and we get

$$\begin{aligned}\dot{e} &= Ae + W^*\Gamma_z(x) - \widehat{W}\Gamma_z(x_r) - Ae\\ &+(A+I)(x-x_r)\\ &+\Omega(u-\alpha_r(t,\widehat{W})).\end{aligned} \tag{10}$$

Now, adding and subtracting in (10) the term $\widehat{W}$ $\Gamma_z(x)$ and defining $\Gamma_z(x) = \Gamma(z(x) - z(x_r))$ we have

$$\begin{aligned}\dot{e} &= Ae + (W^* - \widehat{W})\Gamma_z(x) + \widehat{W}\Gamma(z(x) - z(x_r))\\ &+(A+I)(x-x_r) - Ae + \Omega(u - \alpha_r(t,\widehat{W})).\end{aligned} \tag{11}$$

We define

$$\widetilde{W} = W^* - \widehat{W} \text{ and } \widetilde{u} = u - \alpha_r(t,\widehat{W}) \tag{12}$$

and replacing (12) in (11), we obtain

$$\begin{aligned}\dot{e} &= Ae + \widetilde{W}\Gamma_z(x) + \widehat{W}\Gamma(z(x) - z(x_r))\\ &+(A+I)(x-x_r) - Ae + \Omega\widetilde{u},\end{aligned}$$

$$\begin{aligned}\dot{e} &= Ae + \widetilde{W}\Gamma_z(x) + \widehat{W}\Gamma(z(x) - z(x_p) + z(x_p)\\ &-z(x_r)) + (A+I)(x - x_p + x_p - x_r)\\ &-Ae + \Omega\widetilde{u}.\end{aligned} \tag{13}$$

Now we set

$$\widetilde{u} = u_1 + u_2. \tag{14}$$

So we define

$$\Omega u_1 = -\widehat{W}\Gamma(z(x) - z(x_p)) - (A+I)(x-x_p) \tag{15}$$

and (13) is reduced to

$$\begin{aligned}\dot{e} &= Ae + \widetilde{W}\Gamma_z(x) + \widehat{W}\Gamma(z(x_p) - z(x_r))\\ &+(A+I)(x_p - x_r) - Ae + \Omega u_2.\end{aligned}$$

Considering that $e = x_p - x_r$, the last equation can be written as

$$\dot{e} = (A+I)e + \widetilde{W}\Gamma_z(x) + \widehat{W}\Gamma(z(e+x_r) - z(x_r)) + \Omega u_2,$$

$$\dot{e} = (A+I)e + \widetilde{W}\sigma(x) + \widehat{W}(\sigma(e+x_r) - \sigma(x_r)) + \Omega u_2.$$

If $\phi(e) = \sigma(e+x_r) - \sigma(x_r)$, we get

$$\dot{e} = (A+I)e + \widetilde{W}\sigma(x) + \widehat{W}\phi(e) + \Omega u_2. \tag{16}$$

Now the problem is to find the control law $\Omega u_2$ that stabilizes the system (16). We will obtain the control law by using the Lyapunov methodology.


ISSN 1405-5546
http://dx.doi.org/10.13053/CyS-18-2-2014-040

## 4 Stability of the Tracking Error

Once (16) is obtained, we consider its stabilization in feedforward networks. We note that $(e, \widehat{W})= 0$ is an asymptotically stable equilibrium point of the undisturbed autonomous system $(A = -\lambda I$ and $\lambda > 0)$. For its stability, we propose the next PI control law:

$$\Omega u_2 = K_p e + K_i \int_0^t e(\tau)d\tau - \Upsilon(\frac{1}{2} + \frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e. \tag{17}$$

The parameters $K_p$ and $K_i$ will be determined later, and $L_{\phi z}^2$ is the Lipschitz constant of $\phi_z$, with $\Upsilon > 0$.

We will show that the feedback system is asymptotically stable. Replacing (17) in (16), we obtain

$$\begin{aligned} \dot{e} &= (A+I)e + \widetilde{W}\sigma(x) + \widehat{W}\phi(e) \\ &\quad + K_p e + K_i \int_0^t e(\tau)d\tau \\ &\quad - \Upsilon(\frac{1}{2} + \frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e, \end{aligned} \tag{18}$$

$$\begin{aligned} \dot{e} &= -(\lambda - 1 - K_p)e + \widetilde{W}\sigma(x) + \widehat{W}\phi(e) \\ &\quad + K_i \int_0^t e(\tau)d\tau \\ &\quad - \gamma(\frac{1}{2} + \frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e, \end{aligned} \tag{19}$$

and if $w = K_i \int_0^t e(\tau)d\tau$, then $\dot{w} = K_i\, e(\tau)d\tau$, so we can rewrite (19) as

$$\begin{aligned} \dot{e} &= -(\lambda - 1 - K_p)e + \widetilde{W}\sigma(x) + \widehat{W}\phi(e) + w \\ &\quad - \gamma(\frac{1}{2} + \frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e. \end{aligned} \tag{20}$$

We will show that the new state $(e, w)^T$ is asymptotically stable and that the equilibrium point is $(e, w)^T = (0, 0)^T$, when $\widetilde{W}\sigma(x_r) = 0$, which is taken as an external disturbance.

Let $V$ be the candidate Lyapunov function given by

$$V = \frac{1}{2}(e^T, w^T)(e, w)^T + \frac{1}{2} tr\left\{\widetilde{W}^T \tilde{W}\right\}. \tag{21}$$

The time derivative of (21) along the trajectories of (20) is

$$\begin{aligned} \dot{V} &= (e^T, w^T)\,(\dot{e}, \dot{w})^T + tr\left\{\dot{\widetilde{W}}^T \widetilde{W}\right\} \\ &= e^T \dot{e} + w^T \dot{w} + tr\left\{\dot{\widetilde{W}}^T \widetilde{W}\right\}, \end{aligned} \tag{22}$$

$$\begin{aligned} \dot{V} &= e^T(-(\lambda - 1 - K_p)e + \widetilde{W}\sigma(x) + \widehat{W}\phi(e) \\ &\quad + w + w^T K_i e - \gamma(\frac{1}{2} + \frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2))e \\ &\quad + tr\left\{\dot{\widetilde{W}}^T \widetilde{W}\right\}. \end{aligned} \tag{23}$$

In this part, we select the following learning law from the neural network weights as in [6] and [15]:

$$tr\left\{\dot{\widetilde{W}}^T \widetilde{W}\right\} = -e^T \widetilde{W}\sigma(x). \tag{24}$$

Then (23) is reduced to

$$\begin{aligned} \dot{V} &= -(\lambda - 1 - K_p)e^T e + e^T \widehat{W}\phi(e) \\ &\quad + (1 + K_i)e^T w \\ &\quad - \gamma(\frac{1}{2} + \frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e^T e. \end{aligned} \tag{25}$$

We apply the next inequality to the second term in the right hand side of (25)

$$x^T y \leq \frac{1}{2}x^T x + \frac{1}{2}y^T y \tag{26}$$


ISSN 1405-5546
http://dx.doi.org/10.13053/CyS-18-2-2014-040

to get

$$\begin{aligned} \dot{V} \leq & -(\lambda-1-K_p)e^T e+(\frac{1}{2}+\frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e^T e \\ & +(1+K_i)e^T w \\ & -\gamma(\frac{1}{2}+\frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e^T e. \end{aligned} \quad (27)$$

The parameters in (27) are reduced to

$$\dot{V} \leq -(\lambda-1-K_p)e^T e-(\Upsilon-1)(\frac{1}{2}+\frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e^T e. \quad (28)$$

Here, if we choose $\lambda-1-K_p>0$, and $\Upsilon-1>0$, then $\dot{V}<0$, $\forall\ e,w,\widehat{W}\neq 0$, the error tracking is asymptotically stable and it converges to zero for every $e\neq 0$, this means that the plant follows the reference asymptotically. Finally, the control law which affects the plant and the neural network is given by

$$\begin{aligned} u = & \Omega^{\dagger}[-\widehat{W}\Gamma(z(x)-z(x_p))-(A+I)(x-x_p) \\ & +K_p e+K_i\int_0^t e(\tau)d\tau-\Upsilon(\frac{1}{2}+\frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e \\ & +f_r(x_r,u_r)-Ax_r-\widehat{W}\Gamma_z(x_r)-x_r+x_p] \end{aligned} \quad (29)$$

**Remark 1** $\Omega^{\dagger}$ *is the pseudo inverse in the sense of Moore–Penrose.*

This control law gives asymptotic stability of error dynamics and thus ensures the tracking to the reference signal. The results obtained can be summarized as follows.

**Theorem 2** *For the unknown nonlinear system (1) modeled by (4), the on-line learning law (24) and the control law (29) together ensure the tracking to the nonlinear reference model (5).*

**Remark 3** *From (28) we have*

$$\dot{V}\leq-(\lambda-1-K_p)e^T e-$$

$$(\Upsilon-1)(\frac{1}{2}+\frac{1}{2}\left\|\widehat{W}\right\|^2 L_\phi^2)e^T e<0,\forall e\neq 0,\forall\widehat{W}.$$

where $V$ is decreasing and bounded from below by $V(0)$, and then $V=\frac{1}{2}(e^T,w^T)(e,w)^T+\frac{1}{2}tr\left\{\tilde{W}^T\tilde{W}\right\}$.

We conclude that $e,\widetilde{W}\in L_1$; this means that the weights remain bounded.

## 5 Simulations

In order to demonstrate the applicability of the proposed adaptive control scheme, the following example is tested.

In this example, the unknown plant considered is Chen´s chaotic attractor generated by

$$\begin{aligned} \dot{x}_{p1} &= 35x_{p2}-35x_{p1},\ \ x_{p1}(0)=-10, \\ \dot{x}_{p2} &= -7x_{p1}-x_{p1}x_{p3}+28x_{p2},\ \ x_{p2}(0)=0, \\ \dot{x}_{p3} &= x_{p1}x_{p2}-3x_{p3},\ \ x_{p3}(0)=37. \end{aligned} \quad (30)$$

The goal is to force the chaotic Chen´s attractor to track the reference—the Roosler attractor—generated by

$$\begin{aligned} \dot{x}_{r1} &= -x_{r2}-x_{r3},\ \ x_{r1}(0)=0.1, \\ \dot{x}_{r2} &= x_{r1}+0.2x_{r2},\ \ x_{r2}(0)=0, \\ \dot{x}_{r3} &= x_{r1}x_{r3}-5.7x_{r3}+0.2,\ x_{r3}(0)=0. \end{aligned} \quad (31)$$

In the simulations, the following dynamic neural network was used:

$$\dot{x}=Ax+W^{*}\Gamma z(x)+\Omega u \quad (32)$$

with

$$A=\begin{pmatrix} -5 & 0 & 0 \\ 0 & -5 & 0 \\ 0 & 0 & -5 \end{pmatrix}$$

and

$$\Gamma=\begin{pmatrix} 150 & 0 & 0 \\ 0 & 150 & 0 \\ 0 & 0 & 150 \end{pmatrix},$$

while

$$z(x)=\begin{pmatrix} \tanh(0.5x_1) \\ \tanh(0.5x_2) \\ \tanh(0.5x_3) \end{pmatrix}.$$

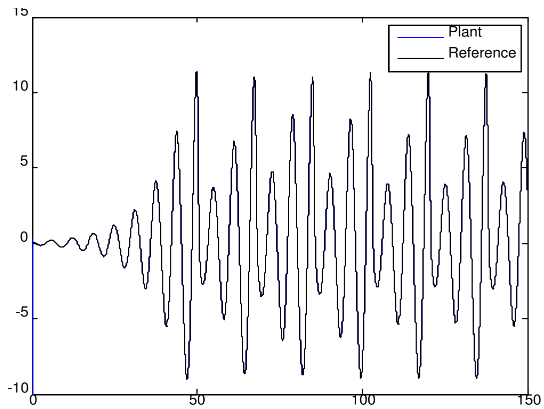


**Fig. 1.** XPlant_Chen-Xref_Roosler

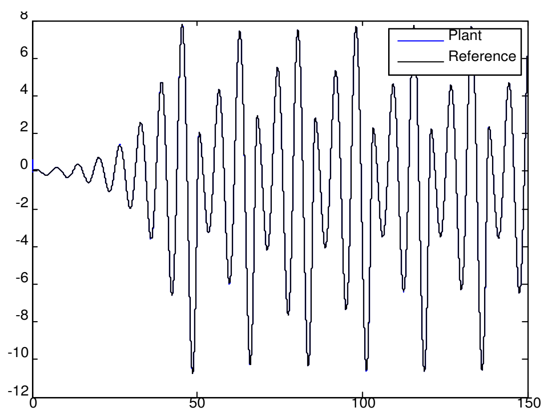


**Fig. 2.** X_Plant_Chen-X_Reference_Roosler

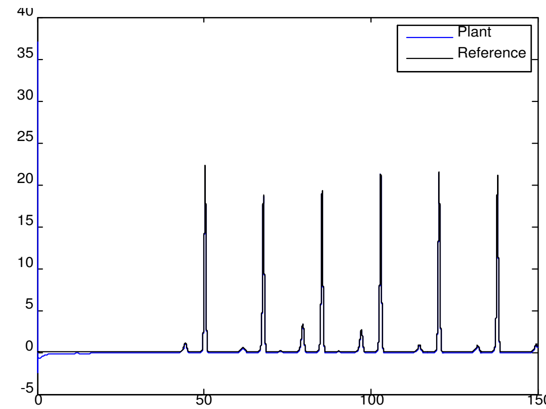


**Fig. 3.** YPlant_Chen-Yref_Roosler

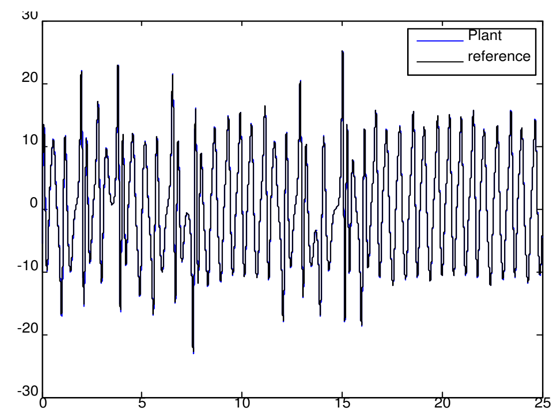


**Fig. 4.** Y_Plant_Chen-Y_Reference_Roosler

$W^*$ is estimated using the learning law given in (24), and the $u$ is calculated using (29).

The results of simulations are shown in Figures 1-11, where the time evolution of the states and phase portraits are presented.

We can see that the Recurrent Neural Controller ensures rapid convergence of the system outputs to the reference trajectory [17]. Another important issue of this approach related to other neural controllers is that most neural controllers are based on indirect control: firstly, the neural network identifies the unknown system and when the identification error is small enough, the control is applied. In our approach, direct control is considered, the learning laws for the neural networks depend explicitly on the tracking error instead of the identification error. This approach results in faster response of the system.

## 6 Conclusions

We have extended the adaptive recurrent neural control previously developed in [13,14,16] to trajectory tracking control problem in order to consider less inputs than states. Stability of the tracking error is analyzed via Lyapunov control functions and the control law is obtained based on the PI approach. A new adaptive control structure based on a dynamic neural network for chaotic orbit tracking of unknown nonlinear systems has been developed. This structure is composed of a neural network identifier and a control law for orbit tracking. Stability of the tracking control system has


ISSN 1405-5546
http://dx.doi.org/10.13053/CyS-18-2-2014-040

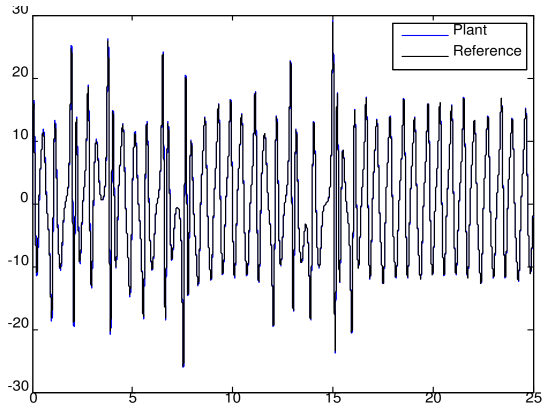


**Fig. 5.** ZPlant_Chen-Zref_Roosler

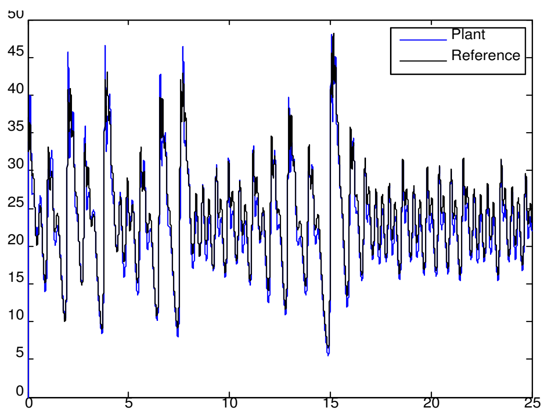


**Fig. 6.** Z_Plant_Chen-Z_Reference_Roosler

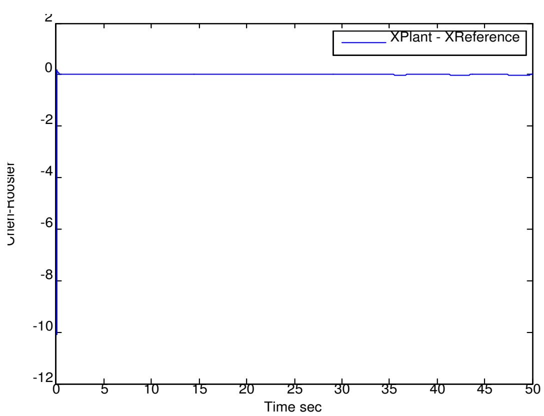


**Fig. 7.** ZPlant_Roosler-Zref_Chen

also been established by means of the Lyapunov function method.

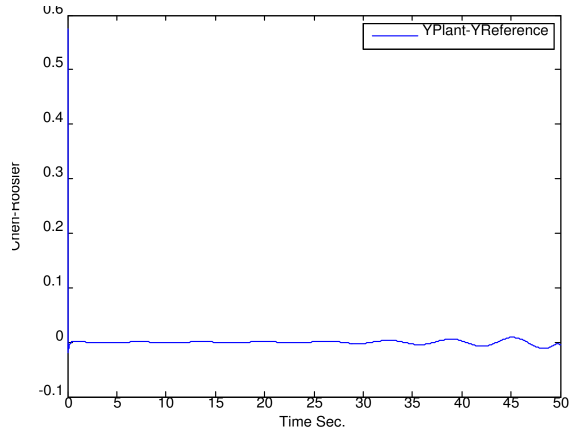


**Fig. 8.** Aphase space trajectory. Roosler tracks Chen

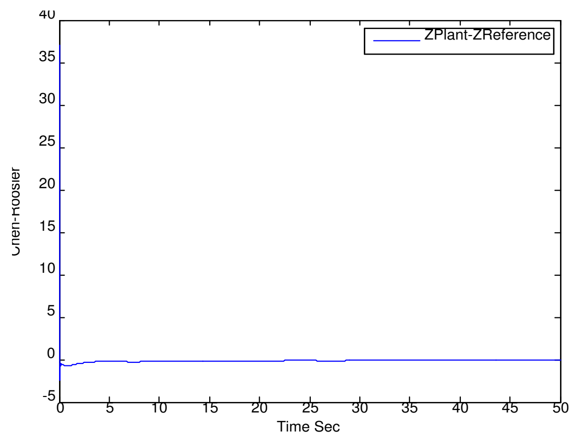


**Fig. 9.** Aphase space trajectory. Chen tracks Roosler

The applicability of the proposed structure was tested via simulations, using Chaos Synchronization as an example of a complex dynamical system. The results are quite encouraging. Research along this line will consist in further relaxing the required condition of having the same number of inputs and states in the control system. Also, we will continue to implement the control algorithm in real time and to perform more tests in a laboratory.

## Acknowledgements

The first two authors appreciate the support of CONACYT, Mexico, and the Matematicas Apli-

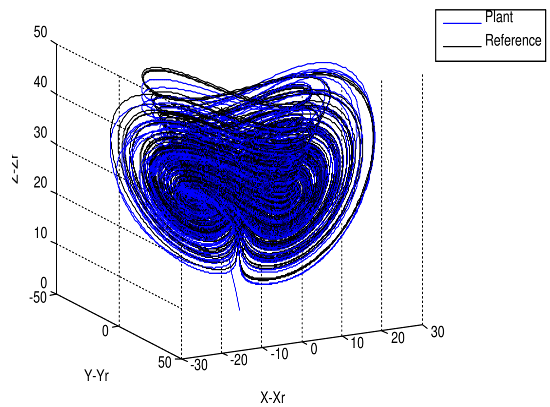


**Fig. 10.** XPlant_Roosler-Xref_Chen

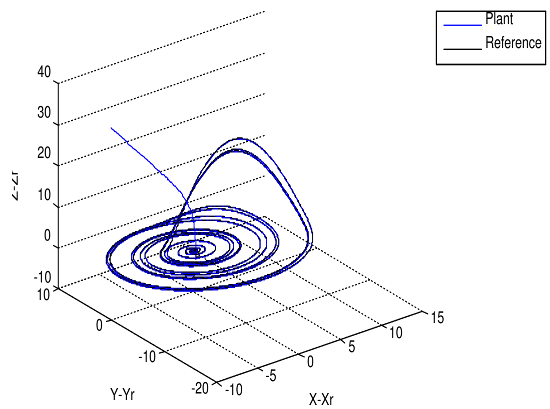


**Fig. 11.** YPlant_Roosler-Yref_Chen

cadas Group of the Facultad de Ciencias Fisico-Matematicas, UANL, Mexico.

## References

1. **Gupta, M.M. & Rao, D.H. (Eds.) (1994).** *Neuro-Control Systems, Theory and Applications.* IEEE Press, Piscataway, N.J., USA.
2. **Hunt, G.I. & Warwick, K. (Eds.) (1995).** *Neural Networks Engineering in Dynamic Control Systems.* Springer Verlang, New York, USA.
3. **Poznyak, A.S., Yu, W., Sanchez, E.N., & Perez, J.P. (1999).** Nonlinear adaptive trajectory tracking using dynamic neural networks. *IEEE Trans. on Neural Networks*, 10(6), 1402–1411.
4. **Narendra, K.S. & Parthasarathy, K. (1990).** Identification and control of dynamical systems using neural networks. *IEEE Trans. on Neural Networks*, 1(1), pp. 4–27.
5. **Suykens K., Vandewalle, L., & De Moor, R. (1996).** *Artificial Neural Networks for Modelling and Control of Nonlinear Systems*. Kluwer academic Publishers, Boston, USA.
6. **Rovitahkis, G.A. & Christodoulou, M.A. (2000).** *Adaptive Control with Recurrent High-Order Neural Networks*. Springer Verlang, New York, USA.
7. **Poznyak, A.S., Sanchez, E.N., & Yu, W. (2000).** *Differential Neural Networks for Robust Nonlinear Control*. World Scientific, USA.
8. **Isidori, A. (1995).** *Nonlinear Control Systems.* 3rd Ed., Springer Verlang, New York, USA,1995.
9. **Hill, D.J. & Moylan, P. (1996).** The Stability of nonlinear dissipative systems. *IEEE Trans. on Auto. Contr.*, vol. 21, 708–711.
10. **Basar, T. & Bernhard, P. (1995).** *H-Infinity Optimal Control and Related Minimax Design Problems*. Birkhauser, Boston, USA.
11. **Krstic, M. & Deng, H. (1998)** *Stabilization of Nonlinear Uncertain Systems*. Springer Verlang, New York, USA.
12. **Sanchez, E.N., Perez, J.P. & Chen, G. (2001).** Using dynamic neural control to generate chaos: An inverse optimal control approach. *Int. J. Bifurcation and Chaos*.
13. **Sanchez, E.N., Perez, J.P., Ricalde, L., & Chen, G. (2001).** Trajectory tracking via adaptive neural control. In *Proceeding of IEEE Int. Symposium on Intelligent Control*, Mexico City, pp. 286–289.
14. **Sanchez, E.N., Perez, J.P., Ricalde, L., & Chen, G. (2001).** Chaos production and synchronization via adaptive neural control. In *Proceeding of IEEE Conference on Decision and Control*, Orlando, Fl, USA.
15. **Ioannou, P.A. & Sun, J.** *Robust Adaptive Control*. PTR Prentice-Hall, Upper Saddle River, NJ 07458.
16. **Astrom, K.J. & Wittenemark, B. (1989).** *Adaptive Control*. Addison-Wesley Publishing Company.
17. **Perez P., J., Perez, J.P., Soto, R., Flores, A., Rodriguez, F., & Meza, J.L. (2012).** Trajectory Tracking Using PID Control Law for Two-Link Robot Manipulator via Adaptive Neural Networks. In *The 2012 Iberoamerican Conference on Electronics Engineering and Computer Science*.

**Joel Perez Padron** obtained his B.Sc. degree in Mathematics in 1991, from the Faculty of Physical and Mathematical Sciences, a Master's degree in Electrical Engineering with specialization in Control in 2001, from the Faculty of Mechanical and Electrical Engineering, and a Ph.D. in Industrial Physics Engineering with specialization in Control in 2008, from the Faculty of Physical and Mathematical Sciences of Nuevo Leon Autonomous University.

**Jose Paz Perez Padron** obtained his B.Sc. degree in Mathematics in 1990, from the Faculty of Physical and Mathematical Sciences, a Master's degree in Electrical Engineering with specialization in Control in 1999, from the Faculty of Mechanical and Electrical Engineering, and a Ph.D. in Science in Electrical Engineering in 2004, from the Cinvestav-Gdl.

**Francisco Rodriguez Ramirez** obtained his B.Sc. degree in Physics in 1989, from the Faculty of Physical and Mathematical Sciences, and a Ph.D. in Sciences with specialization in Mathematics in 2014, from the Faculty of Physical and Mathematical Sciences of Nuevo Leon Autonomous University.

**Angel Flores Hernandez** obtained a degree in Electronics and Communications Engineering in 1997, from the Faculty of Mechanical and Electrical Engineering and currently is pursuing his Ph.D. degree in Industrial Physics Engineering with specialization in Control at the Faculty of Physical and Mathematical Sciences of Nuevo Leon Autonomous University. He completes his Ph.D. studies in August, 2014.

*Article received on 22/10/2012; accepted on 21/09/2013.*