Master-Slave Synchronization for Trajectory Tracking Error Using Fractional Order Time-Delay Recurrent Neural Networks via Krasovskii-Lur'e Functional for Chua's Circuit

J. Javier Pérez D.[1], José Paz Pérez Padron[2],
Atilano Martínez Huerta[2], Joel Pérez Padron[2]

[1] Universidad Autónoma de Nuevo León,
Facultad de Ingeniería Mecánica y Eléctrica,
Mexico

[2] Universidad Autónoma de Nuevo León,
Facultad de Ciencias Fisicomatemáticas,
Departamento de Sistemas Dinámicos,
Mexico

{javiersub20, josepazp}@gmail.com,
{atilano.martinezhrt, joel.perezpd}@uanl.edu.mx

Abstract. Paper presents an application of a Fractional Order Time Delay Neural Networks to chaos synchronization. The two main methodologies, on which the approach is based, are fractional order time-delay recurrent neural networks and the fractional order inverse optimal control for nonlinear systems. The problem of trajectory tracking is studied, based on the fractional order Lyapunov-Krasovskii and Lur'e theory that achieves the global asymptotic stability of the tracking error between a delayed recurrent neural network and a reference function is obtained. The method is illustrated for the synchronization, the analytic results we present a trajectory tracking simulation of a fractional order time-delay dynamical network and the Fractional Order Chua's circuits

Keywords. Trajectory tracking, fractional order time-delay recurrent neural network, fractional order Lyapunov-Krasovskii and Lur'e analysis.

1 Introduction

This paper analyzes the Trajectory Tracking for a Fractional Order Nonlinear System for a Fractional Order Time-Delay Neural Network, which are forced to follow a Fractional Order Reference signal generated by a nonlinear chaotic system. The control law that guarantees trajectory tracking is obtained by using the Fractional Order Lyapunov-Krasovskii and Lur'e methodology. Chaotic behavior, as a characteristic of a dynamical system, could be desirable or undesired, depending of the current application.

In mixing substances, a chaotic behavior might improve the efficiency of the system, while in process which involves vibrations, chaos could produce critical structural failures. Consequently, it is important to be able to manipulate the chaotic nature of the system, driving a stable system to be chaotic or otherwise stabilize a chaotic system. In many applications, it is also important to change the chaotic nature of a system without losing the chaotic behavior.

Controlling and synchronizing chaotic dynamical systems has recently attracted a great deal of attention within the engineering society, in which different techniques have been proposed. For instance, linear state space feedback, Lyapunov-Krasovskii function methods [1],

adaptive control [2]. Using the inverse optimal control approach, a control law [3], which allows reproduce chaos on a Dynamical Neural Network, was discussed in [4]. We further extend these results to the Fractional Order Time-Delay Neural Networks case for nonlinear system trajectory tracking. The proposed new scheme is composed of a Fractional Order delayed dynamical neural identifier, which builds an on-line model for the unknown delayed neural network, and a control law, which ensures that the unknown delayed neural network tracks the reference trajectory.

There are several ways of defining the derivative and fractional integral, for example, the derivative of Grunwald-Letnikov given by eq. (1):

$$aD_t^\alpha f(t) = \lim_{h \to 0} \frac{1}{h^\alpha} \sum_{j=0}^{\left[\frac{(t-\alpha)}{h}\right]} (-1)^j \binom{\alpha}{j} f(t-jh), \qquad (1)$$

where [.] is a flooring-operator while the RL definition is given by eq. (2):

$$aD_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau, \qquad (2)$$

for $(n-1 < \alpha < n)$ and $\Gamma(x)$ is the well-known Euler's Gamma function. Similarly, the notation used in ordinary differential equations, we will use the following notation, eq. (3), when we are referring to the fractional order differential equations where, $\alpha_k \in \mathbb{R}^+$ which is:

$$g\left(t, x, aD_t^{\alpha_1} x, aD_t^{\alpha_2} x, \dots\right) = 0. \qquad (3)$$

The Caputo's definition can be written as:

$$aD_t^\alpha f(t) = \frac{1}{\Gamma(\alpha-n)} \int_a^t \frac{f^{(n)}(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau, \qquad (4)$$

$$\text{for: } (n-1 < \alpha < n).$$

Trajectory tracking, synchronization and control of linear and nonlinear systems are a very important problem in science and control engineering. In this paper, we extend these concepts to force the nonlinear system (Fractional Order Delayed Plant) to follow any linear and nonlinear Fractional reference signals generated by fractional order differential equations.

The applicability of the approach is illustrated by one example: chaos synchronization. In the following, we first briefly describe the dynamic of the fractional order Time-Delay Neural Network to be used.

2 Mathematical Models

The differential equation will be modeled by the neural network [5]:

$$aD_t^\alpha x_p = A(x) + W^* \Gamma_z(x(t-\tau) + \Omega_u, \qquad (5)$$

$x, u \in \mathbb{R}^n, A, W \in \mathbb{R}^{nxn}$, where τ is the fixed known time delay, x is the state, u is the input, $A = -\lambda I$, with λ being a positive constant, W is the state-feedback matrix, and $\sigma(*) = \tanh(*)$ is a Lipschitz function [6] such that $\sigma(x) = 0$ only at $x = 0$, with Lipschitz constant L_σ. It is clear that $x = 0$ is an equilibrium point of this system, when $u = 0$.

The system, to be tracked by the neural network, is defined as:

$$aD_t^\alpha x_r = f(x_r) + g(x_r)u_r,$$

$$x_r, u_r \in \mathbb{R}^n, f(*) \in \mathbb{R}^n, g(*) \in \mathbb{R}^{nxn}, \qquad (6)$$

where $aD_t^\alpha x_r$, is the state, u_r is the input, $f(*)$ and $g(*)$ are smooth nonlinear functions of appropriate dimensions.

As is clear, this is very general, and the model (6) can be a complex such as chaotic nonlinear system.

3 Trajectory Tracking

The objective is to develop a control law such that the delayed neural network (5) tracks the trajectory of the dynamical system (6). We de.ne the tracking error as $e = x - x_r$, whose derivative with respect to time is:

$$aD_t^\alpha e = aD_t^\alpha x - aD_t^\alpha x_r. \qquad (7)$$

Substituting (5, 6, 7), we obtain:

$$aD_t^\alpha e = Ax + W\sigma[x(t-\tau)] + u - f(x_r) \\ - g(x_r)u_r,$$

$$aD_t^\alpha e = Ae + W\sigma[x(t-\tau)] + u - f(x_r) \\ - g(x_r)u_r + Ax_r. \qquad (8)$$

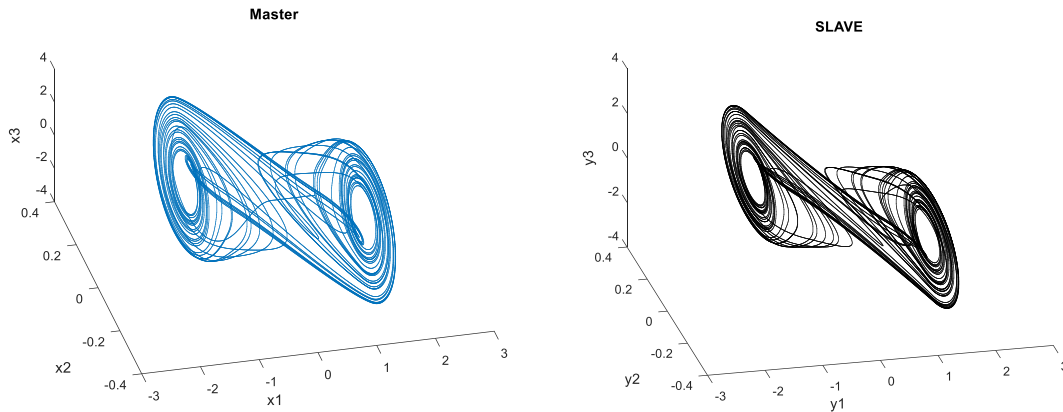Adding and subtracting to (8) the terms $W\sigma[x_r(t-\tau)]$ and $\alpha(t)$, we have:

Fig. 1. α=1: Simulation result for master-slave synchronization between the delayed neural network and the Chua.s circuits. The master system is coupled to the slave system with the first state variable and delay $(\tau = 15\,Sec)$: Three-dimensional view on the double scroll attractor generated for (a) master system and (b) slave system
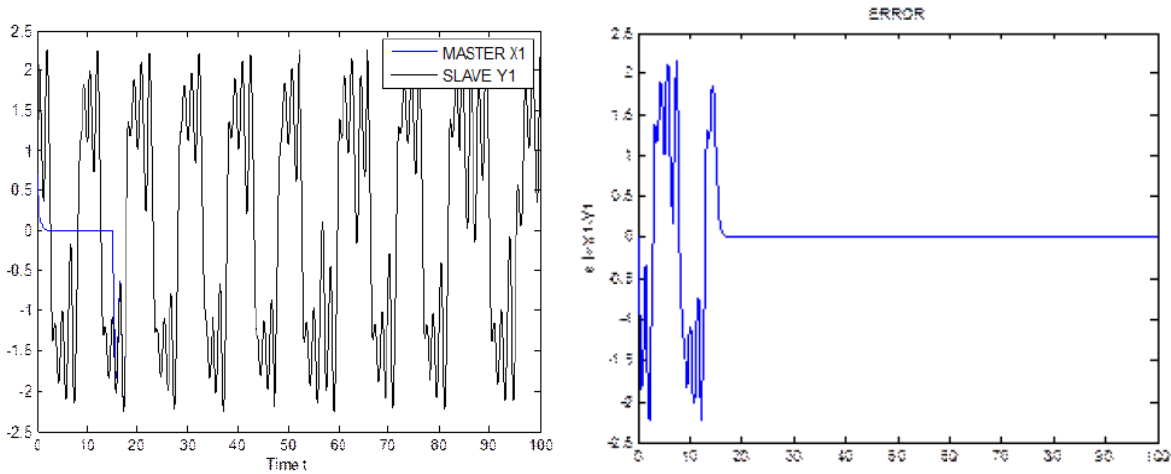

Fig. 2. Time evolution for Delayed Neural Network. Error signal $(x_1(t) - y_1(t))$ with respect to and Chua's circuits with initial condition (0.7; 0; 0) time.

$$aD_t^\alpha e = Ae + W(\sigma[x(t-\tau)] - \sigma[x_r(t-\tau)]) \\ + (u - \alpha(t)) \\ + [Ax_r + W\sigma[x_r(t-\tau)] \\ + \alpha(t)] - f(x_r) - g(x_r)u_r \quad (9)$$

where $\alpha(*)$ is a function to be determined.

For system (5) to follow model (6), the following solvability assumption is needed, as discussed in [7]:

Assumption 1. There exist functions $\rho(t)$ and $\alpha(t)$, such that:

$$aD_t^\alpha \rho = A\rho(t) + W\sigma[x(t-\tau)] + \alpha(t); \rho(t) \\ = x_r(t). \quad (10)$$

It follows from (10 and 6) that:

$$[Ax_r + W\sigma[x_r(t-\tau)] + \alpha(t)] = aD_t^\alpha x_r \\ = f(x_r) + g(x_r)u_r, \quad (11)$$

$$\alpha(t) = f(x_r) + g(x_r)u_r - Ax_r - W\sigma[x_r(t-\tau)].$$
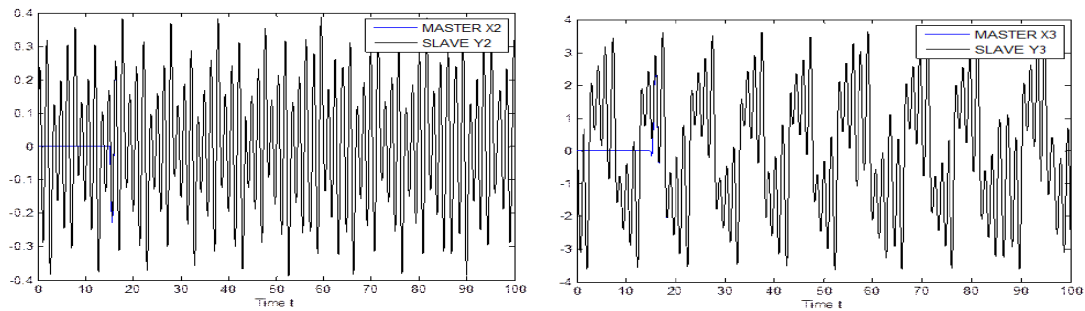
So that (9) becomes:

Fig. 3. Time evolution for Delayed Neural Network. Error signal $(x_2(t) - y_2(t))$ with respect to and Chua's circuits with initial condition (0.7; 0; 0) time
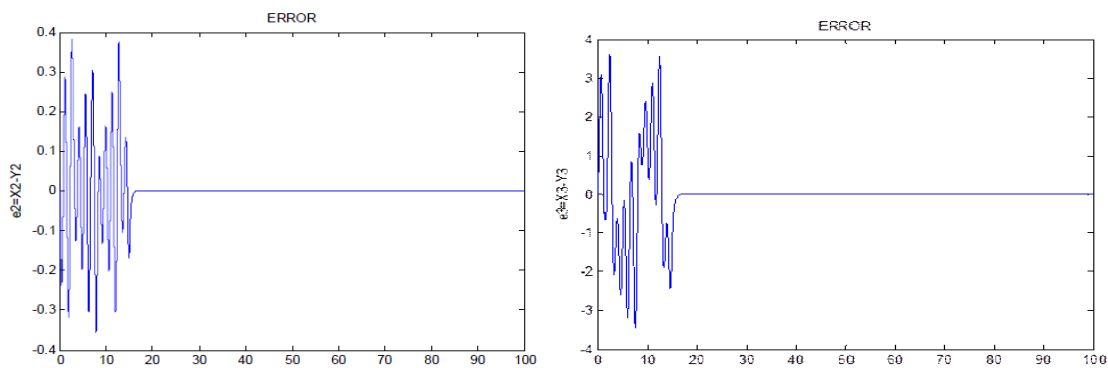

Fig. 4. Time evolution for Delayed Neural Network. Error signal $(x_3(t) - y_3(t))$ with respect to and Chua's circuits with initial condition (0.7; 0; 0) time
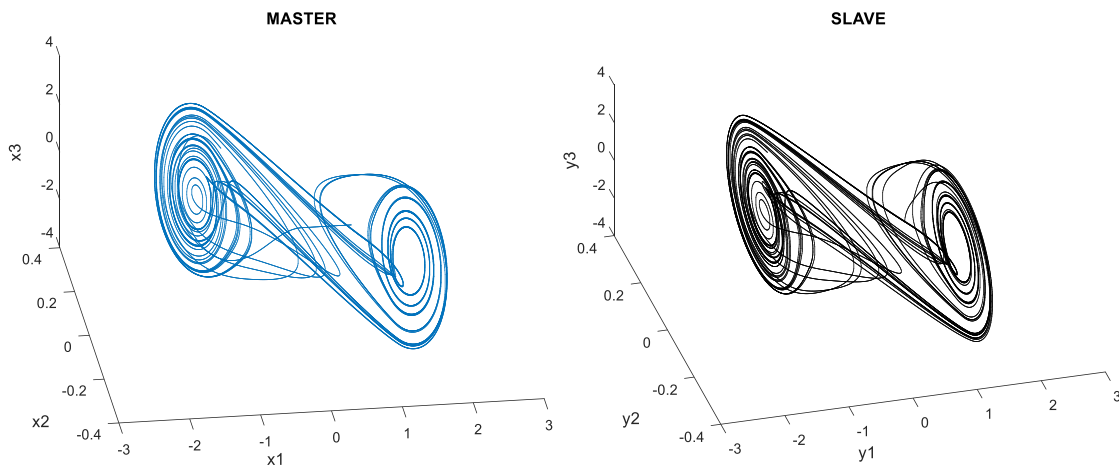

Fig. 5. α=0.00001: Simulation result for master-slave synchronization between the fractional order delayed neural network and the Chua's circuits. The master system is coupled to the slave system with the first state variable and delay ($\tau = 15\ Sec$): Three-dimensional view on the double scroll attractor generated for (a) master system and (b) slave system

$$aD_t^\alpha e = Ae + W(\sigma[x(t-\tau)] - \sigma[x_r(t-\tau)]) + (u - \alpha(t)).$$

Let us define $\tilde{u} = (u - \alpha(t))$,

$$aD_t^\alpha e = Ae + W(\sigma[x(t-\tau)] - \sigma[x_r(t-\tau)]) + \tilde{u}. \quad (12)$$

It is clear that $e = 0$, is an equilibrium point of (12), when $\tilde{u} = 0$. In this way, the tracking problem can be restated as a global asymptotic stabilization problem for the system (12).

4 Tracking Error Stabilization and Control Design

In order to establish the convergence of (12) to e = 0, which ensures the desired tracking, first, we propose the following Krasovskii [8] and Lur'e functional [9].

This is essential for the design of a globally and asymptotically stabilizing control law. We select:

$$V(e) = \sum_{i=1}^{n} \int_0^{e_i} \emptyset(\tau, x_r) d\tau + \int_{t-\tau}^{t} (\emptyset_\sigma^T(s) W^T W \emptyset_\sigma(s) ds. \quad (13)$$

The time derivative of (13), along the trajectories of (12):

$$aD_t^\alpha V = \emptyset(\tau, x_r)^T \dot{e} + \emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t) \\ - \emptyset_\sigma^T(t-\tau) W^T W \emptyset_\sigma(t-\tau), \quad (14)$$

$$aD_t^\alpha V = \emptyset(e, x_r)^T Ae + \emptyset(e, x_r)^T W(\sigma[x(t-\tau)] \\ - \sigma[x_r(t-\tau)]) + \emptyset(e, x_r)^T \tilde{u} \\ + \emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t) - \emptyset_\sigma^T(t \\ - \tau) W^T W \emptyset_\sigma(t-\tau). \quad (15)$$

We select:

$$\emptyset_\sigma^T(t-\tau) = (\sigma[x(t-\tau)] - \sigma[x_r(t-\tau)]) aD_t^\alpha V \\ = -\lambda \emptyset(e, x_r)^T e \\ + \emptyset(e, x_r)^T W \emptyset_\sigma^T(t-\tau) \\ + \emptyset(e, x_r)^T \tilde{u} \\ + \emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t) - \emptyset_\sigma^T(t \\ - \tau) W^T W \emptyset_\sigma(t-\tau). \quad (16)$$

Next, let consider the following inequality, proved in [10]:

$$X^T Y + Y^T X \le X^T \Lambda X + Y^T \Lambda^{-1} Y, \quad (17)$$

which holds for all matrices $X, Y \in \mathbb{R}^{nxk}$ and $\Lambda \in \mathbb{R}^{nxn}$ with $\Lambda = \Lambda^T > 0$. Applying (17) with $\Lambda = I$ to the term $\emptyset(e, x_r)^T W \emptyset_\sigma^T(t-\tau)$, we get:

$$aD_t^\alpha V \le -\lambda \emptyset(e, x_r)^T e + \frac{1}{2} \emptyset(e, x_r)^T \emptyset(e, x_r) \\ + \frac{1}{2} \emptyset_\sigma^T(t-\tau) W^T W \emptyset_\sigma(t \\ - \tau) + \emptyset(e, x_r)^T \tilde{u} \\ + \emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t) - \emptyset_\sigma^T(t \\ - \tau) W^T W \emptyset_\sigma(t-\tau). \quad (18)$$

By simplifying (18), we obtain:

$$aD_t^\alpha V \le -\lambda \emptyset(e, x_r)^T e + \frac{1}{2} \emptyset(e, x_r)^T \emptyset(e, x_r) \\ - \frac{1}{2} \emptyset_\sigma^T(t-\tau) W^T W \emptyset_\sigma(t-\tau) \\ + \emptyset(e, x_r)^T \tilde{u} \\ + \emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t). \quad (19)$$

Since $\emptyset(e, x_r)$ is a sector function with respect to e, there exist positive constants k_1 and k_2 such that $k_1 \parallel e \parallel_2^2 \le \emptyset(e, x_r)^T \le k_2 \parallel e \parallel_2^2$ [11]. Also, since $\emptyset(e, x_r)$ is Lipschitz with respect to e, there exist a positive constant L_σ such that $\emptyset(e, x_r)^T \emptyset(e, x_r) \le L_\sigma^2 \parallel e \parallel_2^2$. Henceforth (19) can be rewritten and then we have that:

$$aD_t^\alpha V(e) \le -\left[\lambda k_1 - \frac{1}{2} L_\sigma^2\right] \parallel e \parallel_2^2 \\ - \frac{1}{2} \emptyset_\sigma^T(t-\tau) W^T W \emptyset_\sigma(t-\tau) \\ + + \emptyset(e, x_r)^T \tilde{u} \\ + \emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t). \quad (20)$$

By simplifying (20), we have:

$$aD_t^\alpha V(e) \le -\left[\lambda k_1 - \frac{1}{2} L_\sigma^2\right] \parallel e \parallel_2^2 + \emptyset(e, x_r)^T \tilde{u} \\ + \emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t). \quad (21)$$

Since $\emptyset_\sigma$ is Lipschitz with Lipschitz constant L_σ [12], then:

$$\|\emptyset_\sigma(t)\| = \|\sigma(x(t)) - \sigma(x_r(t))\| \\ \le L_\sigma^2 \|x(t) - x_r(t)\| \\ = L_\sigma^2 \|e(t)\|_2^2. \quad (22)$$

Applying to:

$$\emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t),$$

$$\emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t) \le \|\emptyset_\sigma^T(t) W^T W \emptyset_\sigma(t)\| \\ \le L_\sigma^2 \|W\|_2^2 \|e(t)\|_2^2. \quad (23)$$

With L_σ^2 the Lipschitz constant of (23) $\sigma(*)$. To the right hand, third term of (17), we obtain:

$$aD_t^\alpha V(e) \le -\left[\lambda k_1 - \frac{1}{2} L_\sigma^2\right] \parallel e \parallel_2^2 \\ + L_\sigma^2 \|W\|_2^2 \|e(t)\|_2^2 \\ + \emptyset(e, x_r)^T \tilde{u}. \quad (24)$$

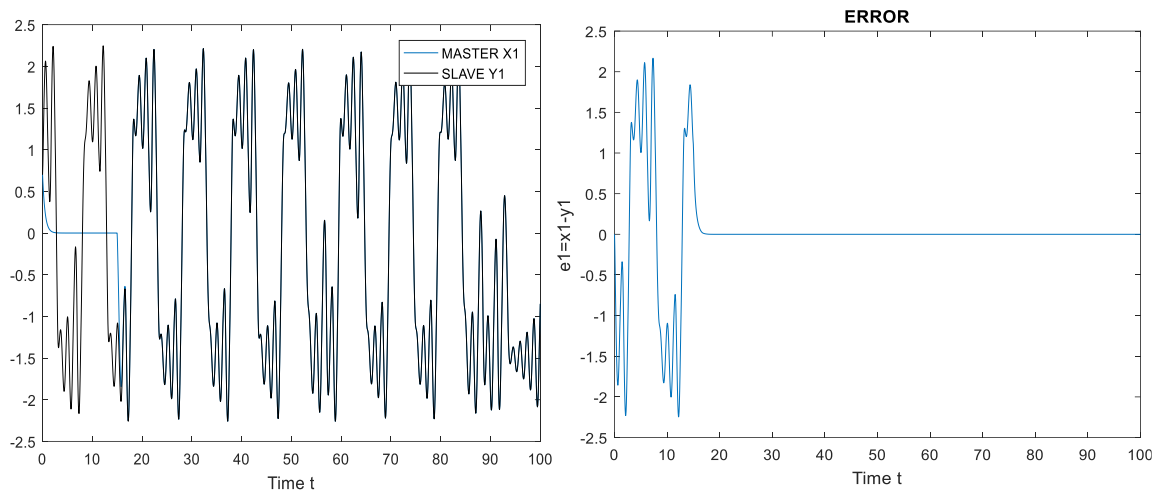Now, we suggest to use the following control law:

Fig. 6. Time evolution for the Fractional Order. Error signal $(x_1(t) - y_1(t))$ with respect to Delayed Neural Network and Chua's circuits time, with initial condition (0.7; 0; 0)
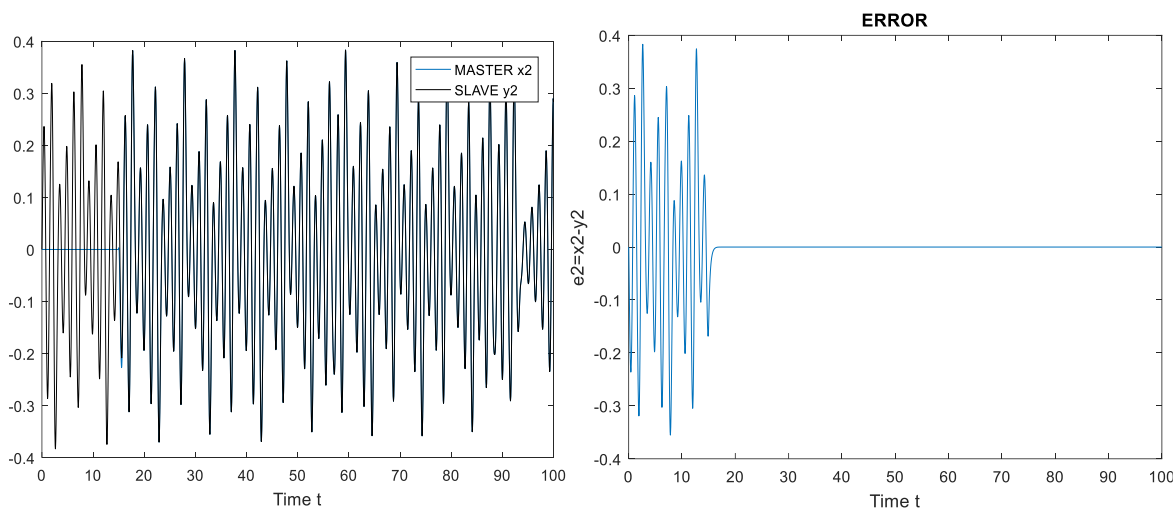

Fig. 7. Time evolution for the Fractional Order. Error signal $(x_2(t) - y_2(t))$ with respect to Delayed Neural Network and Chua's circuits' time, with initial condition (0.7; 0; 0)

$$\tilde{u} = -(2 + 2 * L_\sigma^2 \|W\|_2^2)\emptyset(e, x_r)^T$$
$$= -\beta\big(R(e)\big)^{-1}\big(L_g V\big)^T, \qquad (25)$$

where β is a positive constant and $(R(e))^{-1}$ is a function of e: At this point, substituting (25) into (24), we obtain:

$$aD_t^\alpha V(e) \leq -[\lambda + L_\sigma^2 + L_\sigma^2\|W\|_2^2] \parallel e \parallel_2^2 \qquad (26)$$

Then $aD_t^\alpha V(e) < 0$ for all $e \neq 0$. This means that the proposed control law (27) can globally and asymptotically stabilize the error system, therefore ensuring the tracking of (5) by (6).

Finally, the control action driving the recurrent neural networks is given by:

$$u = -(2 + 2 * L_\sigma^2\|W\|_2^2)\emptyset(e, x_r)^T + f(x_r)$$
$$+ g(x_r)u_r - Ax_r \qquad (27)$$
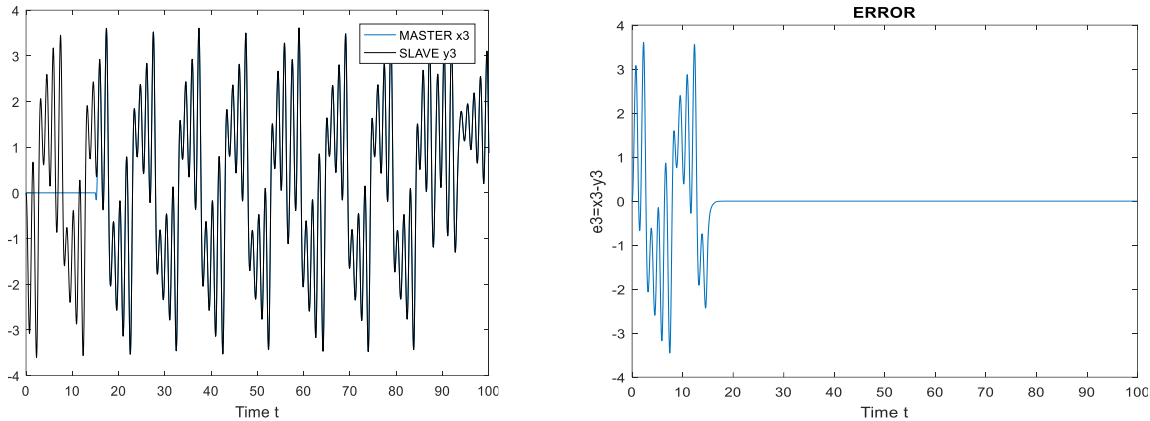$$- W\sigma[x_r(t - \tau)].$$

Fig. 8. Time evolution for the Fractional Order Error signal $(x_3(t) - y_3(t))$ with respect to Delayed Neural Network and Chua's circuits time, with initial condition (0.7; 0; 0)

We summarize the above developed analysis in the following Theorem.

Theorem 1: The control law (27) ensures that the Time-Delay Neural Network (5) tracks the reference system (6).

5 Simulations

In order to illustrate the applicability of the discussed results, we consider we consider the following delayed neural network: $aD_t^\alpha x_p = A(x) + W\sigma[x(t - \tau)] + u$, where:

$$A = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix},$$

$$W = \begin{pmatrix} 0.3 & 0.8 & 0 \\ 0.4 & 0.3 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$\sigma(x(t - \tau)) = \begin{pmatrix} \tanh(x_1(t - \tau)) \\ \tanh(x_2(t - \tau)) \\ \tanh(x_3(t - \tau)) \end{pmatrix},$$

$$\tau(28) = 15 \ .$$

(28)

The reference model to be tracked is the Chua.s circuits [13]. This system is described by:

$$(\blacksquare(aD\_t\hat{}\alpha\ x\_r = 15.6y\_r - 15.6x\_r \\ - 15.6\ \llbracket\{-1.143x\rrbracket\ \_r + ((-1.143 \\ + 0.714))/2\ [|x\_r + 1| - |x\_r \\ - 1|]\}@@aD\_t\hat{}\alpha\ y\_r\ \llbracket \\ = x\rrbracket\ \_r - y\_r - z\_r\ @aD\_t\hat{}\alpha\ z\_r \\ = -28y\_r\ ))$$

(29)

The experiment is performed as follow. Both systems, the delayed neural network and the Chua's circuits, evolve independently until $\tau = 15$ seconds: at that time, the proposed control law (23) is incepted.

Simulation results are presented in Fig. 2, 3 y 4, for state 1, state 2 and state 3, respectively. As can be seen, tracking is successfully achieved.

The experiment is performed as follow. Both systems, the fractional order delayed neural network and the Chua's circuits, evolve independently until $\tau = 15$ seconds: at that time, the proposed control law (23) is incepted. Simulation results are presented in Fig. 6, Fig. 7, Fig. 8, for state 1, state 2 and state 3, respectively. As can be seen, tracking is successfully achieved.

6 Conclusions

We have presented the controller design for trajectory tracking determined by a Fractional Order Time-Delay dynamical network.

This framework is based on dynamic Fractional Order delayed neural networks and the methodology is based on Fractional Order Lyapunov-Krasovskii and Lur'e theory.

The proposed Inverse Optimal Control Law is applied to a dynamical fractional order delayed neural network and the Chua's circuits, respectively, being able to also stabilize in

asymptotic form the tracking error between two systems.

The results of the simulation show clearly the desired tracking. In future work, we will consider the stochastic case for the complex dynamical network.

Acknowledgements

References

1. **Perez, J.P., Perez-Padron, J., Flores-Hernández, A., Arroyo, S. (2014).** Complex dynamical network for trajectory tracking using Delatey recurrent neural networks. Hindawi Publishing. DOI: 10.1155/2014/ 162610.

2. **Perez-P, J., Perez, J.P., Hernández, J.J., Arroyo, S., Flores, A. (2012).** Trajectory tracking error using PID control law for a 2 DOF helicopter model via adaptive time-delay neural networks. Congreso Nacional de Control Automatico. Procedia Technology, Vol. 3, pp. 139–146. DOI: 10.1016/j.protcy.2012. 03.015.

3. **Nanda-Kumar, M.P., Dheeraj, K. (2014).** An inverse optimal control approach for the nonlinear system design using ANN. International Journal of Computer, Information, Systems and Control Engineering, Vol. 8, No. 6.

4. **Kousaka, T., Ueta, T., Yue Ma, Kawakami, H. (2006).** Control of chaos in a piecewise smooth nonlinear system. Chaos, Solitons and Fractals, Vol. 27, pp. 1019–1025.

5. **Delay, Xia Huang, Zhen Wang, Yuxia Li (2013).** Nonlinear dynamics and chaos in fractional-order Hopfield neural networks. Advances in Mathematical Physics, Vol. 2013. DOI: 10.1155/2013/ 657245.

6. **Khalil, H. (1996).** Nonlinear system analysis. Prentice Hall, Upper Saddle River, NJ, USA.

7. **Perez-P., J. (2016).** Fractional complex dynamical systems for trajectory tracking using fractional neural network. Computación y Sistemas, Vol. 20, No. 2, pp. 219–225. DOI: 10.13053/CyS-20-2-2201.

8. **Hai-Peng Jiang, Yong-Qiang Liu (2016).** Disturbance rejection for fractional-order time-delay systems. Mathematical Problems in Engineering, Vol. 2016. DOI: 10.1155/ 2016/1316046.

9. **Ka Song, Huaiquin Wu, Lifei Wang, Lur'e-Postnikov Lyapunov.** Function approach to global robust Mitag-Leffler stability of fractional-order neural networks. Advances in Difference Equations, Open Journal. DOI 10.1186/s13662-017-1298-8.

10. **Weiyuan Ma, Changpin Li, Yujiang Wu, Yongqing Wu (2014).** Adaptive synchronization of fractional neural networks with unknown parameters and time delays. Entropy, Vol. 16, pp. 6286–6299. DOI: 10.3390/e16126286.

11. **Zhuo Li (2015).** Fractional order modeling and control of multi-input-multi-output processes. University of California.

12. **Diethelm, K., Ford, N.J. (2002).** Analysis of fractional differential equations. Journal of Mathematical Analysis and Applications, Vol. 265, pp. 229–248. DOI: 10.1006/jmaa.2000.7194.

13. **Petras, I. (2000).** Control of fractional-order Chua's system. Technical University of Kosice.