Trajectory tracking error using fractional order time-delay recurrent neural networks using Krasovskii-Lur’e functional for Chua’s circuit via inverse optimal control

J. Pérez Padron and J.P. Pérez Padron

e-mail: joel.perezpd@uanl.edu.mx; josepazp@gmail.com

C.F. Mendez-Barrios
Centro de Investigación y Estudios de Posgrado, Facultad de Ingeniería, Dr. Manuel Nava # 8, Zona Universitaria, 78290, San Luis Potosí, S.L.P., México,
e-mail: fernando.barrios@uaslp.mx

E.J. González-Galván
Centro de Investigación y Estudios de Posgrado, Facultad de Ingeniería, Dr. Manuel Nava # 8, Zona Universitaria,78290, San Luis Potosí, S.L.P., México,
e-mail: egonzale@uaslp.mx

Received 14 September 2019; accepted 30 September 2019

This 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.

DOI: https://doi.org/10.31349/RevMexFis.66.98

1. Introduction

This paper analyzes the Trajectory Tracking for a Fractional Order Nonlinear System for a Fractional Order Time-Delay Neural Network, which is 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 on the current application. In mixing substances process, a chaotic behavior might improve the efficiency of the system, while in the process which involves vibrations, chaos could produce critical structural failures. As a consequence, 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 reproducing 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 control law.

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

\[ a D_t^\alpha f(t) = \lim_{\Box \to 0} \frac{1}{\Box^\alpha} \sum_{j=0}^{[(t-\alpha)/\Box]} (-1)^j \left( \begin{array}{c} \alpha \\ j \end{array} \right) f(t-j\Box). \]  

(1)

Where \( \Box \) is a flooring-operator while the RL definition is given by:

\[ a D_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau. \]  

(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, \( \alpha, k \in \mathbb{C}^+ \).

Which is:

\[
    g(t, x, \alpha, D_t^{\alpha_1} x, aD_t^{\alpha_2} x, \ldots) = 0. \tag{3}
\]

The Caputo’s definition can be written as

\[
    aD_t^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha)} \int_0^t (t-\tau)^{n-\alpha-1} f(\tau) d\tau. \tag{4}
\]

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 will 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.

This obtained result, modeling by differential equations of fractional order, is new, unlike other results obtained by the authors, modeled by ordinary differential edifications.

The effectiveness of the methodologies, from our point of view, are equal, the difference is that we have observed that the response of the controller is in smaller magnitude in the systems of fractional order than in the systems of ordinary differential equations, and also, in the systems of fractional order have some slack by varying the order of the fractional system, which causes the response of the system to be smoother, which does not happen with ordinary non-linear systems.

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]:

\[
    aD_t^{\alpha} x_{\rho} = A(x) + W^* \Gamma z x(t-\tau) + \Omega u
\]

\[
    x, u \in \mathbb{R}^n, \quad A, W_i \in \mathbb{R}^{n \times n} \tag{5}
\]

where \( \tau \) is the fixed known time delay, \( x \) is the state, \( u \) is the input, \( A = -\lambda I \), with \( \lambda \) being a positive constant, \( W \) is the state-feedback matrix, and \( \sigma(*) = \tan h(*) \) is a Lipschitz function [6] such that \( \sigma(x) = 0 \) only at \( x = 0 \), with Lipschitz constant \( L_\sigma \). It is clear that \( x = 0 \) is an equilibrium point of this system, when \( \alpha = 0 \).

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

\[
    aD_t^{\alpha} x_r = f(x_r) + g(x_r) u_r,
\]

\[
    x_r, u_r \in \mathbb{R}^n, \quad f(*) \in \mathbb{R}^n, \quad g(*) \in \mathbb{R}^{n \times n} \tag{6}
\]

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 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 define the tracking error as \( e = x - x_r \), whose derivative for time is:

\[
    aD_t^{\alpha} e = aD_t^{\alpha} x - aD_t^{\alpha} x_r. \tag{7}
\]

Substituting (5) and (6) in (7), we obtain

\[
    aD_t^{\alpha} e = Ae + W\sigma(x(t-\tau)) + u - f(x_r) - g(x_r) u_r
\]

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

\[
    - g(x_r) u_r + Ax_r \tag{8}
\]

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

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

\[
    + [Ax_r + W\sigma(x_r(t-\tau)) + \alpha(t)]
\]

\[
    - f(x_r) - g(x_r) u_r. \tag{9}
\]

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

\[
    aD_t^{\alpha} \rho = A\rho(t) + W\sigma(x(t-\tau)) + \alpha(t);
\]

\[
    \rho(t) = x_r(t). \tag{10}
\]

It follows from (10) and (6) that

\[
    [Ax_r + W\sigma(x_r(t-\tau)) + \alpha(t)] = aD_t^{\alpha} x_r = f(x_r) + g(x_r) u_r
\]

\[
    \alpha(t) = f(x_r) + g(x_r) u_r - Ax_r - W\sigma(x_r(t-\tau)). \tag{11}
\]

So that (9) becomes

\[
    aD_t^{\alpha} e = Ae + W(\sigma(x(t-\tau))
\]

\[
    - \sigma(x_r(t-\tau)) + (u - \alpha(t))
\]

Let’s define \( \ddot{u} = (u - \alpha(t)) \)

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

It is clear that \( e = 0 \), is an equilibrium point of (12), when \( \ddot{u} = 0 \). 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

To establish the convergence of (12) to \( e = 0 \), which ensures the desired tracking, 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

\[
V(e) = \sum_{i=1}^{n} \int_{0}^{\tau_i} \phi(\tau, x_r) d\tau + \int_{t}^{t-\tau} \phi_0^T(s)W^T W\phi_0(s) ds.
\]  

(13)

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

\[
aD^\alpha_t V(e) = \phi(\tau, x_r)^T \dot{e} + \phi_0^T(t)W^T W\phi_0(t)
\]

\[- \phi_0^T(t-\tau)W^T W\phi_0(t-\tau)
\]  

(14)

\[
aD^\alpha_t V(e) = \phi(e, x_r)^T Ae + \phi(e, x_r)^T W(\sigma[x(t-\tau)])
\]

\[- \sigma[x_r(t-\tau)]) + \phi(e, x_r)^T W\phi_0(t)
\]

\[- \phi_0^T(t-\tau)W^T W\phi_0(t-\tau)
\]  

(15)

We select \( \phi_0^T(t-\tau) = (\sigma[x(t-\tau)]) - \sigma[x_r(t-\tau)]) \)

\[
aD^\alpha_t V(e) = -\lambda(\phi(e, x_r)^T e + \phi(e, x_r)^T W\phi_0^T(t-\tau)
\]

\[+ \phi(e, x_r)^T \dot{u} + \phi_0^T(t-\tau)W^T W\phi_0(t-\tau)
\]

\[- \phi_0^T(t-\tau)W^T W\phi_0(t-\tau)
\]  

(16)

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

\[X^T Y + Y^T X \leq X^T \Lambda X + Y^T \Lambda^{-1} Y \]  

(17)

which holds for all matrices \( X, Y \in \mathbb{R}^{n \times k} \), and \( \Lambda \in \mathbb{R}^{n \times n} \) with \( \Lambda = \Lambda^T > 0 \). Applying (17) with \( \Lambda = I \) to the term \( \phi(e, x_r)^T W\phi_0^T(t-\tau) \), we get

\[
aD^\alpha_t V(e) \leq -\lambda(\phi(e, x_r)^T e + \frac{1}{2}(\phi(e, x_r)^T \phi(e, x_r)
\]

\[+ \phi(e, x_r)^T \dot{u} + \phi_0^T(t-\tau)W^T W\phi_0(t-\tau)
\]

\[- \phi_0^T(t-\tau)W^T W\phi_0(t-\tau).
\]  

(18)

By simplifying (18), we obtain

\[
aD^\alpha_t V(e) \leq -\lambda(\phi(e, x_r)^T e + \frac{1}{2}(\phi(e, x_r)^T \phi(e, x_r)
\]

\[- \frac{1}{2}\phi_0^T(t-\tau)W^T W\phi_0(t-\tau)
\]

\[+ \phi(e, x_r)^T \dot{u} + \phi_0^T(t)W^T W\phi_0(t).
\]  

(19)

Since \( \phi(e, x_r) \) is a sector function for \( e \), there exist positive constants \( k_1 \) and \( k_2 \) such that \( k_1 \| e \|_2 \leq \phi(e, x_r)^T e \leq k_2 \| e \|_2^2 \) [11]. Also, since \( \phi(e, x_r) \) is Lipschitz for \( e \), there exist a positive constant \( L_\sigma \) such that \( \phi(e, x_r)^T \phi(e, x_r) \leq L_\sigma \| e \|_2^2 \). Henceforth (19) can be rewritten and then we have that

\[
aD^\alpha_t V(e) \leq -\left[ \lambda k_1 - \frac{1}{2} L_\sigma^2 \right] \| e \|_2^2
\]

\[- \frac{1}{2}\phi_0^T(t-\tau)W^T W\phi_0(t-\tau)
\]

\[+ \phi(e, x_r)^T \dot{u} + \phi_0^T(t)W^T W\phi_0(t).
\]  

(20)

By simplifying (20), we have

\[
aD^\alpha_t V(e) \leq -\left[ \lambda k_1 - \frac{1}{2} L_\sigma^2 \right] \| e \|_2^2
\]

\[+ \phi(e, x_r)^T \dot{u} + \phi_0^T(t)W^T W\phi_0(t).
\]  

(21)

Since \( \phi_0 \) is Lipschitz with Lipschitz constant \( L_\sigma \) [12], then

\[
\| \phi_0(t) \| = \| \sigma(x(t)) - \sigma(x_r(t)) \| \leq L_\sigma \| x(t) - x_r(t) \| = L_\sigma \| e(t) \|_2^2.
\]  

(22)

Applying to \( \phi_0^T(t)W^T W\phi_0(t) \)

\[
\phi_0^T(t)W^T W\phi_0(t) \leq \phi_0^T(t)W^T W\phi_0(t) \| \leq L_\sigma \| W \|_2^2 \| e(t) \|_2^2.
\]  

(23)

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

\[
aD^\alpha_t V(e) \leq -\left[ \lambda k_1 - \frac{1}{2} L_\sigma^2 \right] \| e \|_2^2
\]

\[+ L_\sigma^2 \| W \|_2^2 \| e(t) \|_2^2 + \phi(e, x_r)^T \dot{u}.
\]  

(24)

Now, we suggest to use the following control law:

\[
\dot{u} = -(2 + 2 * L_\sigma^2) \| W \|_2^2 \| e \|_2^2 - \beta(R(e))^{-1}(L_\sigma V)^T.
\]  

(25)

Where \( \beta \) is a positive constant and \( (R(e))^{-1} \) is a function of \( e \). At this point, substituting (25) into (24), we obtain

\[
aD^\alpha_t V(e) \leq -[\lambda + L_\sigma^2 + L_\sigma^2 \| W \|_2^2] \| e \|_2^2.
\]  

(26)

Then \( aD^\alpha_t V(e) < 0 \) 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:

\[
u = -(2 + 2 * L_\sigma^2) \| W \|_2^2 \| e \|_2^2 + f(x_r)
\]

\[+ g(x_r)u_r - Ax_r - W\sigma[x_r(t-\tau)]
\]  

(27)

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

To illustrate the applicability of the discussed results, we selected the following delayed neural network: 

$$aD_\alpha^t x_p = A(x) + W\sigma(x(t-\tau)) + u,$$

where

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

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

$$\tau = 15 \text{ sec.}$$

(28)

The reference signal, which the neural network has to follow is the chaotic circuit of Chua [13], described by the differential equation of fractional order:

$$aD_\alpha^t x_r = 15.6 y_r - 15.6 x_r - 15.6 \left\{ -1.143 x_r + \frac{(-1.143 + 0.714)}{2} \left[ |x_r + 1| - |x_r - 1| \right] \right\}$$

(29)

$$aD_\alpha^t y_r = x_r - y_r - z_r$$

$$aD_\alpha^t z_r = -28 y_r$$

(30)
Figure 4. Time evolution for Delayed Neural Network and Chua’s circuits with initial condition \((0.7; 0; 0)\) and the error signal \((x_3(t) - y_3(t))\) concerning time.

Figure 5. \(\alpha = 0.00001\) 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.

Figure 6. Time evolution for the Fractional Order Delayed Neural Network and Chua’s circuits with initial condition \((0.7; 0; 0)\) and the error signal \((x_1(t) - y_1(t))\) concerning time.

Figure 7. Time evolution for the Fractional Order Delayed Neural Network and Chua’s circuits with initial condition \((0.7; 0; 0)\) and the error signal \((x_2(t) - y_2(t))\) concerning time.

Rev. Mex. Fís. 66 (1) 98–104
The experiment is performed as follows. 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, 7, 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, it can be mentioned that the results will be extended to model non-linear systems, whose mathematical model is not completely known, and in that sense, it can be decided that the laws of control and laws of learning are robust.

It is important to mention that, we will are work on simulation in real time to control a humanoid, and these results are very promising, since they would help people who have lost some lower limb, and to control humanoids, which could help in tasks that are dangerous for humans.

Acknowledgements

The authors thank the support of CONACyT and the Dynamical Systems group of the Facultad de Ciencias Físico-Matemáticas, Universidad Autónoma de Nuevo León, México.

9. Ka Song, Huaiquin Wu and Lifei Wang, Advances in Difference


13. Ivo Petras, Control of fractional-order Chua’s systems, e-mail: petras@tuke.sk