Ingeniería Investigación y Tecnología
volumen XXI (número 3), julio-septiembre 2020 1-12
ISSN 2594-0732 FI-UNAM artículo arbitrado
Información del artículo: Recibido: 29 de julio de 2019, reevaluado: 23 de agosto de 2019, aceptado: 10 de junio de 2020

https://doi.org/10.22201/fi.25940732e.2020.21.3.026

Ingeniería Investigación y Tecnología

Real-time neural control of all-terrain tracked robots with unknown dynamics and network communication delays

Control neuronal en tiempo real para robots todo terreno tipo oruga con dinámicas y retardos de red de comunicaciones desconocidos

Alanis Alma Y.
Universidad de Guadalajara, México
Centro Universitario de Ciencias Exactas e Ingenierías
Departamento de Ciencias Computacionales
E-mail: almayalanis@gmail.com
https://orcid.org/0000-0001-9600-779X

Arana-Daniel Nancy
Universidad de Guadalajara, México
Centro Universitario de Ciencias Exactas e Ingenierías
Departamento de Ciencias Computacionales
E-mail: nancyaranad@gmail.com
https://orcid.org/0000-0002-8803-9502

Rios Jorge D.
Universidad de Guadalajara, México
Centro Universitario de Ciencias Exactas e Ingenierías
Departamento de Ciencias Computacionales
E-mail: jorge_rios.1xyz@yahoo.com
https://orcid.org/0000-0001-7565-0874

López-Franco Carlos
Universidad de Guadalajara, México
Centro Universitario de Ciencias Exactas e Ingenierías
Departamento de Ciencias Computacionales
E-mail: clzfranco@gmail.com
https://orcid.org/0000-0003-4854-101X

Abstract

Currently, wireless communication networks have been acquired great relevance in our daily life, including data acquisition, data processing, control and analysis in different applications. Therefore, robotic systems cannot be an exception, in such a way, it is an important study that considers the effects caused by inclusion of wireless networks in the control loop of robotic systems, as well as, the designing of intelligent control systems that can deal with such effects in real-time. Hence, this research work focuses on the designing of an on-line intelligent controller that achieves trajectory tracking of a robotic mobile system which is in a networked communication environment. The proposed controller can deal with unknown dynamics, unknown external and internal disturbances, unknown communication delays and packet losses. Such a controller is designed using a discrete-time approach based on an inverse optimal control methodology for trajectory tracking and a recurrent high-order neural network identifier. Applicability of the proposed scheme is shown through real-time results using a tracked robot platform controlled through a wireless network under different network scenarios. Besides, obtained results, show good performance. The designed scheme can be extended to any unknown or uncertain nonlinear system which lies in a networked environment. One of the main advantages of the proposed scheme is the robustness of the proposed intelligent controller to work on networked environments under direct communication channels, as well as through two different communication channels, evidently for more complex configurations of the used communication channel, the performance of the proposed scheme can be deteriorated.
Keywords: Neural networks, networked control system, cyber-physical systems, time delay, tracked robot.

Resumen

Actualmente, las redes de comunicación inalámbricas han adquirido gran relevancia en nuestra vida diaria, incluida la adquisición de datos, su procesamiento, el control y el análisis de datos, en diferentes aplicaciones. Así pues, los sistemas robóticos no pueden ser una excepción, por lo que es importante y relevante estudiar los efectos causados por la inclusión de redes inalámbricas en el lazo de control de sistemas robóticos, así como el diseño de sistemas de control inteligentes que pueden manejar tales efectos en tiempo real. Por tanto, este trabajo de investigación se centra en el diseño de un controlador inteligente en línea que logra el seguimiento de la trayectoria de un sistema móvil robótico que se encuentra en un entorno de comunicación en red. El controlador propuesto puede manejar dinámicas desconocidas, saturación de los actuadores, perturbaciones externas e internas desconocidas, retrasos de comunicación desconocidos y pérdidas de paquetes. Dicho controlador está diseñado utilizando un enfoque de tiempo discreto basado en un control óptimo inverso y un identificador neuronal de alto orden recurrente. La aplicabilidad del esquema propuesto se muestra a través de resultados en tiempo real utilizando una plataforma de robot tipo oruga, controlada a través de una red inalámbrica bajo diferentes escenarios de la red. Además, los resultados obtenidos, muestran un buen desempeño. El esquema diseñado puede extenderse a cualquier sistema no lineal desconocido o incierto que se encuentre en un entorno de red. Una de las principales ventajas del esquema propuesto es la robustez del controlador inteligente para trabajar en entornos de red bajo canales de comunicación directa, así como a través de dos canales de comunicación diferentes, evidentemente entre más compleja sea la configuración del canal de comunicación, el desempeño del esquema propuesto puede deteriorarse.
Descriptores: Neural networks, networked control system, cyber-physical systems, time delay, tracked robot.

https://doi.org/10.22201/fi.25940732e.2020.21.3.026

Introduction

The contribution of Norbert Wiener has had a profound effect on many fields including signal processing, systems, control theory and cybernetics. He is known as the father of cybernetics, which, in his own words, is "the scientific study of control and communication in the animal and the machine (Wiener, 1961)". Recently, cybernetics has contributed tremendously to the development of intelligent control and the smart management of cyber-physical systems (CPS). Examples of real-life systems considered as CPS are smart grids, automotive and transportation systems, smart healthcare, unmanned aerial vehicles (UAVs), robotics, and the Internet of Things, among others (Lee, 2008; Rajkumar *et al*., 2010; Haque *et al*., 2014; Ochoa *et al*., 2017; Dressler, 2018; Lopez *et al*., 2018; Alguliyeb *et al*., 2018; Do *et al*., 2018).

Typically, a CPS is a smart networked system that is designed to interact with the physical world including human users; this interaction is achieved through its embedded sensors, processors, and actuators (Lee, 2008; Rajkumar *et al*., 2010; Li *et al*., 2017). Therefore, emerging technical challenges arise with the rapid increase in functionalities, significant uncertainties and stringent requirements on performance, representation, analytics, smart management, safety, security, flexibility and reliability, modeling, analysis, control, and others; these aspects lead to increasingly complex CPS design (Lee, 2008; Rajkumar *et al*., 2010). In this way, communication channels are essential for networked control in CPS, since they provide the required information for feedback in the control law. Information on the physical system dynamics is sent to the controller by the implemented sensors using the communication channels. Hence, the efficiency and reliability of communication channels are crucial to developing an adequate control to support real-time and trustworthy CPS applications.

Today, Internet communication has become an essential part of the modern world. The recent progress of communication technologies has led to extensive research on the possible applications of remote-control technology implemented over a communication line in the field of control engineering (Kobayashi *et al*., 2017; Ochoa *et al*., 2017; Do *et al*., 2018). Nevertheless, a networked control system is permanently affected by negative factors, which include induced time-delay and packets losses, typical characteristics during network operation. One possible solution to these issues is updating the network equipment; however, this solution is costly. Another solution is the use of robust and reliable control methodologies needed to decrease the negative impact of time-delay and packet losses. Among these problems, the proposed controller should address other common problems found in CPS, like complex interactions among potentially conflicting actuations, continuous and discrete dynamics with discrete controllers, unknown and unmodeled dynamics, internal and external unknown uncertainties, substantial delays that imply reduced stability regions, unknown variable time delays and so on Yu *et al*. (2017); Kobayashi *et al*. (2017).

In recent years, intelligent controllers, particularly those for neural control, have showed the ability to control complex systems subject to the abovementioned complications, even with no exact, incomplete or unknown mathematical models of the system or if the application is complex; therefore, neural control is an obvious methodology for the modeling and control of CPS (Selyunin *et al*., 2015a & 2015b; Lv *et al*., 2017). In Jiang *et al*. (2016), it is presented a deep comparative analysis between the two main methodologies of intelligent control: fuzzy and neural based controllers, then in Farias (2018) it is developed the same comparison with an applicability focused to robotics, these two works illustrate the great capability of intelligent controllers to deal with unknown and unstructured environments, fuzzy controllers are mainly recommended for their simplicity to implementation, besides, fuzzy controller allows the user to include known characteristics of the system by a specialist which cannot be done with neural controllers, however, neural controllers are indicated in complex systems or tasks, where the designer has limited information about the system to be controlled. Use of intelligent control for robotics applications has been deeply studied, for example in Aouf *et al*. (2019) and Tong *et al*. (2020) consider implementation of fuzzy controllers for robots, in Pillai & Suthakorn (2019) it is presented a compendium of challenges in motion control of rough terrain rescue robots. In Rios *et al.* (2017), an implementation of the NIOC scheme applied to a modified HD2® (HD2 is a registered trademark of SuperDroid Robots) is presented, such work presents the simulation and experimental results as well as a comparison of the NIOC scheme with a super twisting scheme. In Villaseñor *et al*. (2018), a germinal center optimization algorithm is used to find a better set of values for the NIOC scheme designed parameters for its implementation to the HD2®.

The main contribution of this paper is to show how the NIOC scheme is capable of working in different network conditions, mainly in presence of unknown delays, packet losses, disturbances and uncertainties, without their previous knowledge. In order to illustrate applicability of the NIOC, three tests are presented, they are implemented on real-time for a tracked robot platform controlled through a wireless network under different network scenarios: Test 1, direct communication between the computer that process signals and the HD2®, which sends its status through a state vector and receives and applies a control signal calculated by the computer. Test 2 and Test 3 communicate through two different routers connected to an inner network in addition to other services; the network provides Internet connection services to all devices connected to it.

In this work, the networked control strategy is described first, and then, the analysis of robustness towards these negative factors is presented through experimental results. The results are shown in detail and prove that the proposed networked control strategy has strong robustness against unknown dynamics, unknown external and internal disturbances, unknown communication delays and packet losses, the fact of the proposed controller deals with all the above explained conditions can be considered as the main strengths of the proposed NIOC, in this sense the main weaknesses and threats remains in the complexity of the proposed algorithm which implies the need of a robust processor to implement all the elements of the NIOC, what can be expensive, besides the implementation of this kind of controller requires previous knowledge of modern control strategies.

This work is focused on the intelligent control of networked systems. In the first stage, the system to be controlled is identified by a recurrent high-order neural network (RHONN) identifier; then, an inverse optimal control (IOC) is designed based on the mathematical model obtained with the RHONN. The designed neural inverse optimal controller (NIOC) is applied in real time to a networked system for trajectory tracking under three different scenarios.

Neural optimal inverse control scheme

This section presents elements of NIOC scheme for unknown discrete-time delayed nonlinear systems. This scheme is composed by a neural identifier trained with an EKF based algorithm, this identifier provides a mathematical model for the unknown delayed system, therefore the obtained neural model is used to design a controller based on inverse optimality technique for trajectory tracking.

First consider the following system:

$$x\,(k+1) = f\,(x\,(k-1) + g\,(x(k\ -l)))\ u\,(k) \tag{1}$$

where $x \in \Re^n$ is the state vector, $u \in \Re^m$ is the control input vector, and $f \in \Re^m \to \Re^n$ and $g \in \Re^n \to \in \Re^{n\times m}$ are smooth maps.

RHONN Identifier

A RHONN is a generalization of the first-order recurrent neural network known as the Hopfield network. A recurrent neural network has memory and dynamic behavior due to its inner feedback connections. Moreover, in a RHONN, the high-order connections enhance the approximation capabilities, convergence, storage capability and fault tolerance of the neural network (Haykin *et al*., 2004; Sanchez *et al*., 2008; Zhang, 2008). The following RHONN identifier based on the RHONN series-parallel model is used to identify system (1) (Alanis *et al*., 2016):

$$\hat{X}_i\,(k+1) = \omega_i\,(k) Z_i(x\,(k-l),\,u\,(k)),\ i = 1,2,\,\dots\,,\,n \tag{2}$$

Where:

$\hat{X}_i$ = i-the state variable of the neural network
n = state dimension
ω_i = weight vector of $\hat{X}_i$ and $Z_i(\cdot)$ is defined as:

$$Z_i(x(k),u(k)) = \begin{bmatrix} Z_{i_1} \\ Z_{i_2} \\ \vdots \\ Z_{iL_i} \end{bmatrix} = \begin{bmatrix} \prod_{j\in I_1} \xi_{i_j}^{d_{ij}(1)} \\ \prod_{j\in I_2} \xi_{i_j}^{d_{ij}(2)} \\ \vdots \\ \prod_{j\in I_{L_i}} \xi_{i_j}^{d_{ij}(Li)} \end{bmatrix} \tag{3}$$

x is the state vector of the system to be identified, and u is the input vector and the high order terms are defined as:

$$\xi_i = \begin{bmatrix} \xi_{i_1} \\ \vdots \\ \xi_{i_n} \\ \xi_{i_n+1} \\ \vdots \\ \xi_{i_{n+m}} \end{bmatrix} = \begin{bmatrix} S(x_1(k-l)) \\ \vdots \\ S(x_n(k-l)) \\ u_1(k) \\ \vdots \\ u_m(k) \end{bmatrix} \tag{4}$$

where x_i is the i-th state variable of the system (with $i = 1,\dots,n$), is the unknown system delay and u_j is the j-th input component of system input u (with $i = 1,\dots,m$) and

$$S(v) = \frac{1}{1+exp(-Bv)} \tag{5}$$

where $\beta > 0$ and v is a variable with any real value.

https://doi.org/10.22201/fi.25940732e.2020.21.3.026

It is important to note that the RHONN identifier (2) does not require previous knowledge of the system model to be identified nor information of its disturbances and delays.

The selected training algorithm for the RHONN identifier (2) is based on an extended Kalman filter (EKF). The EKF finds the optimal weight vector that minimizes the prediction error at each iteration, and it is computed between sampling time instants from the previous estimation and current input in an iterative manner. The EKF-based training algorithm (Sanchez *et al.*, 2008) is:

$$\omega_i(k+1) = \omega_i(k) + \eta_i K_i(k)\, e_i(k)$$

$$K_i(k) = P_i(k)\, H_i(k)\, [R_i(k) + H_i(k)\, P_i(k) H_i(k)]^{-1} \tag{6}$$

$$P_i(k+1) = P_i(k) - K_i(k)\, H_i(k)\, P_i(k) + Q_i(k)$$

where $i = 1, 2, \ldots, n$, $\omega_i \in \Re^{Li}$, is the weight vector, $K_i \in \Re^{Li}$ is the Kalman gain vector, $e_i \in \Re$ is the i-th identification error and L_i represents the number of high order terms for the i-th neural state variable. $P_i \in \Re^{Li \times Li}$ is the weight estimation error covariance matrix, $\hat{X}^i$ is the i-th state variable of the neural network, $Q_i \in \Re^{Li \times Li}$ is the estimation noise covariance matrix, $R_i \in \Re$ is the error noise covariance matrix and $H_i \in \Re^{Li}$ is a vector of derivatives. Then:

$$e_i(k) = x_i(k) - \hat{X}_i(k) \tag{7}$$

where $e_i \in \Re$ is the i-th identification error defined as the difference between the i-th state variable x^i and the respective neural state variable $\hat{X}_i$ and $H_i \in \Re^{Li}$ is a vector with entries H_{ij}, are defined as:

$$H_{ij} = \left[\frac{\partial \hat{X}_i(k)}{\partial \omega_{ij}(k)}\right]^T \tag{8}$$

where ω_{ij} is the j-th element of vector ω_i. Finally, P_i and Q_i are initialized as diagonal matrices with entries $P_i(0)$ and $Q_i(0)$, respectively.

Inverse Optimal Control

The optimal control leads to a control law that minimizes a performance criterion. This control law is obtained through a process that involves the solution of a Hamilton-Jacobi–Bellman (HJB) equation (Sanchez *et al.*, 2016). Since solving this equation is not an easy task, an alternative is to use inverse optimal control (IOC), which avoids this solution. In the IOC approach, a stabilizing feedback control law is designed based on a priori knowledge of a control Lyapunov function (CLF). Then, it is established that the control law optimizes a cost function. Finally, the CLF is modified to achieve asymptotic tracking for given references (Sanchez *et al.*, 2016).

The system (1) is supposed to have an equilibrium point $x(0) = 0$. Moreover, the full state $x(k)$ is assumed to be available. In order to ensure system stability of (1), the following control Lyapunov function is proposed:

$$V(z(k)) = 1/2\, z(k)\, \boldsymbol{P}\, z(k) \tag{9}$$

Where:

V = candidate Lyapunov function
$\mathbf{P}$ = positive matrix such that $\mathbf{P} = \mathbf{P} > 0$
z = trajectory tracking error, with:

$$z(k) = x(k) - x_d(k) \tag{10}$$

where $z(k)$ represents trajectory tracking error, which is the difference between system state $x(k)$ and desired reference signal $x_d(k)$.

The inverse optimal control law for the system (1) with (9) is:

$$\begin{aligned} u(k) &= -\frac{1}{2}\boldsymbol{R}^{-1}(z(k))g^T(z(k))\frac{\partial V(z(k))}{\partial z(k+1)} \\ &= -\frac{1}{2}(\boldsymbol{R}(z(k)) + \frac{1}{2}\mathrm{g}^T(z(k))\mathbf{P}\,\mathrm{g}(z(k)))^{-1}\mathrm{g}^T(z(k))\mathbf{P}f(z(k)) \end{aligned} \tag{11}$$

Where:

$u(k)$ = control law
$\boldsymbol{R}(z(k)) = \boldsymbol{R}(z(k))^T > 0$ = matrix whose elements can be functions of the system state or can be fixed
$\boldsymbol{P}$ = matrix such that the inequality (12) holds. Therefore:

$$V_f(z(k)) - \frac{1}{4}\boldsymbol{P}_1(z(k))(\boldsymbol{RP}(z(k)))^{-1}\mathbf{P}_1(z(k)) \le -z^T(k)\boldsymbol{Q}(k) \tag{12}$$

where V_f is the first increment of (9) defined as:

$$V_f(z(k)) = \frac{1}{2}f^T(z(k))\mathbf{P}_f(z(k)) - V(z(k)) \tag{13}$$

where $z(k)$ defined as in (10), $\mathbf{Q} = \mathbf{Q}^T > 0$ is a gain matriz, $\mathbf{P}$ is defined as in (9) and $\boldsymbol{RP}(z(k))$ is defined as:

$$\boldsymbol{RP}(z(k)) = \boldsymbol{R}(z(k)) + \boldsymbol{P}_2(z(k)) \tag{14}$$

https://doi.org/10.22201/fi.25940732e.2020.21.3.026

with $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ functions of $g\,(\cdot)$, defined as:

$$\boldsymbol{P}_1(z(k)) = g^{T}(z(k))\boldsymbol{P}_f(z(k)) \tag{15}$$

$$\boldsymbol{P}_2(z(k)) = \frac{1}{2}g^{T}(z(k))\boldsymbol{P}_g(z(k)) \tag{16}$$

The globally asymptotic stability of the control law (11) is demonstrated in (Sanchez & Ornelas, 2016).

Neural Inverse Optimal Control for Unknown Delayed Systems

Control law (11) need the complete knowledge of mathematical model (1), however in this paper we consider the problem to control unknown discrete-time delayed systems, therefore to solve this problem it is used the neural identifier (2) to obtain a mathematical model of system (1), therefore control law (11) is designed using (2) then, inverse optimal control based on a neural model is called NIOC and the whole scheme is depicted in Figure 1.

Discussion and results

The following results were obtained using a modified HD2® all-terrain tank robot (Figure 2). The modification consists of a replacement of the original board for a system based on two Arduino® (Arduino is a registered trademark of Arduino LLC), boards and a wireless router. All other parts of the HD2 remained unmodified.

For the implementation of the presented identifier-control scheme, the mathematical model of the HD2® robot is not needed, and the RHONN identifier provides the model. This implementation is achieved by adapting RHONN's weight vectors using the error between its output and its inputs, which are the measured signal from the robot. For reference, a mathematical model of an all-terrain tracked robot can be found in (Rios *et al*., 2017), such model is not necessary to design RHONN, however it can be used as a guide.

Test 1: Wireless connection to the all-terrain tracked robot

Test 1 description: The computer that processes the signals and computes the control signal u is connected via wireless communication to the router mounted in the HD2®. The HD2® system sends the information of the measured state variables to the computer using TCP/IP protocol. At the time the controller starts, a command prompt (CMD) window with a ping loop to the HD2® is initialized to record the network behavior with respect to time (Figure 4) in addition to testing the ability of the source computer (the HD2®). A visual representation of Test 1 is presented in Figure 3.

Figures 5, 6 and 7 show the comparison of the real measured signals (blue), identified signals (orange) and reference signals (yellow) for position x, position y and position θ, respectively. In these figures, the blue measured real signals are not visible because they are covered by the orange identified signals, visually showing that the errors between them are close to zero. Meanwhile, the errors between the position signals and their references are not zero; however, they are small and bounded.

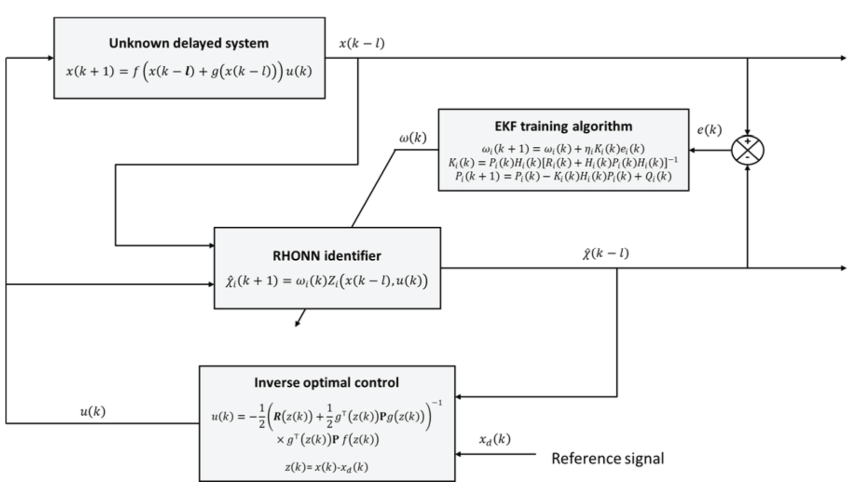

Figure 1. Neural inverse optimal control scheme

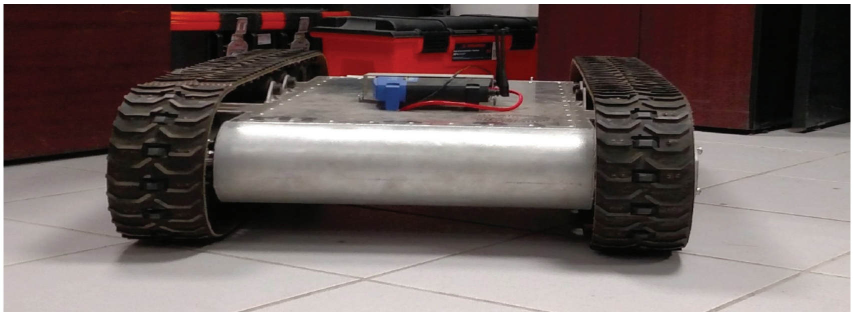

Figure 2. HD2® All-Terrain Tracked Robot

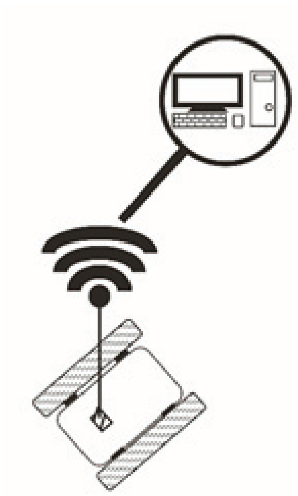

Figure 3. Graphical description of Test 1

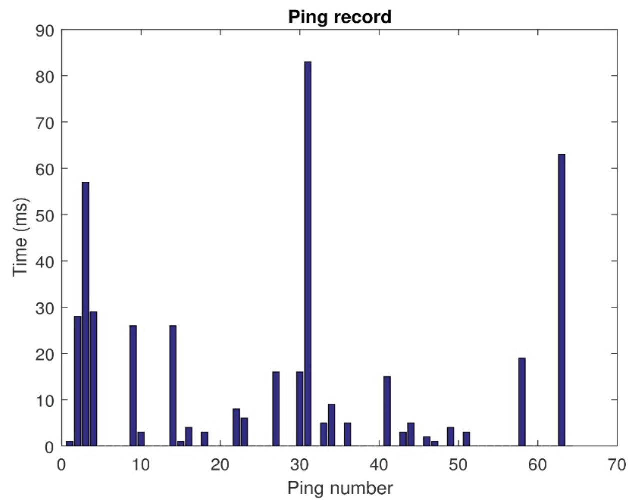

Figure 4. Ping record of Test 1. Ping statistics: Packets: Sent = 64, Received = 64, Lost = 0 (0 % Loss). Approximate round trip in milliseconds: Minimum = 0 *ms*, Maximum = 83 *ms*, Media = 6 *ms*. Please note that in the graphic values of zero represent results of less than 1 *ms*

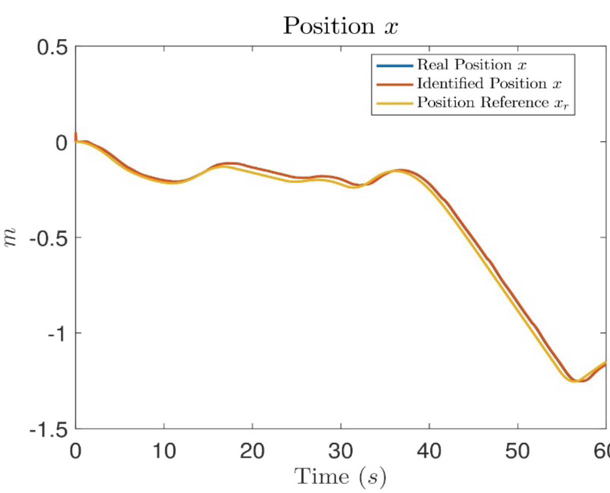

Figure 5. Comparative graph between the measured real position x, the identified $\hat{x}$ and reference x_d of Test 1

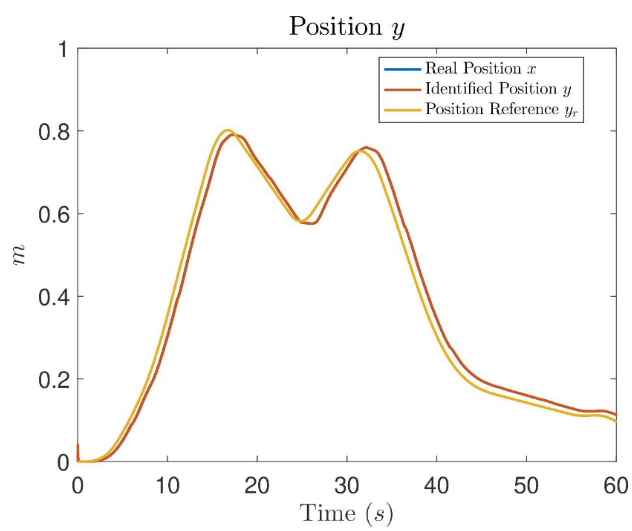

Figure 6. Comparative graph between the measured real position y, the identified $\hat{y}$ and reference y_d of Test 1

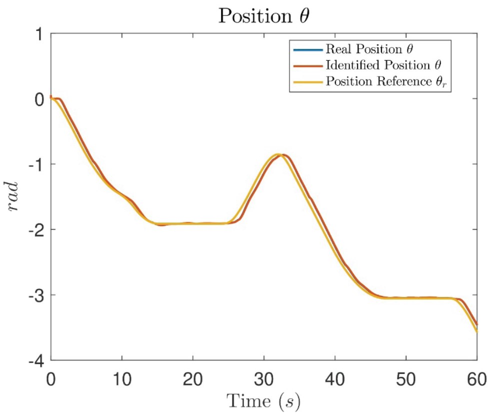

Figure 7. Comparative graph between the measured real position θ, the identified $\hat{\theta}$ and reference θ_d of Test 1

Table 1 shows the identification root mean square errors (RMSEs) for all state variables of the HD2®. Moreover, Table 2 shows the tracking RMSEs, which are calculated from the identified signals and the reference signals.

Table 1. Identification RMSE of Test 1

	RMSE
Position x	0.001865 m
Position y	0.001577 m
Position θ	0.005342 rad
Velocity v_1	0.036624 m/s
Velocity v_2	0.038498 m/s
Current i_1	0.131179 A
Current i_2	0.125383 A

Table 2. Tracking RMSE for Position x, Position y and Position θ of Test 1

	RMSE
Position x	0.026473 m
Position y	0.030891 m
Position $\boldsymbol{\theta}$	0.072156 rad

The control signals of Test 1 are shown in Figures 8 and 9.

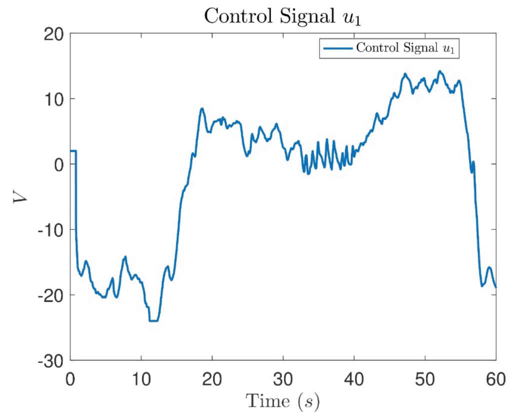

Figure 8. Control signal u_1 of Test 1

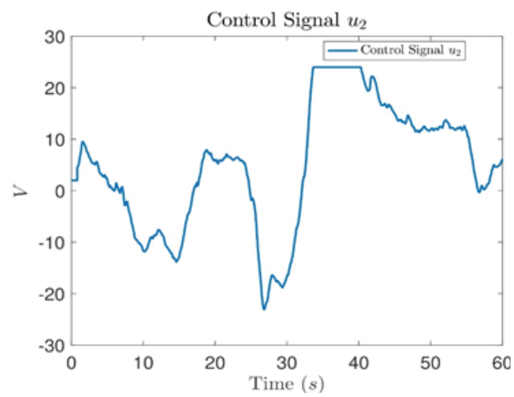

Figure 9. Control signal u_2 of Test 1

Test 2: Communication to the all-terrain tank robot through different routers

Test 2 description: The computer that processes the signals and computes the control signal u is connected via wireless communication to a router connected to an inner network. The router mounted in the HD2® is also connected to the same network.

The HD2® system sends the information of the measured stated variables to the computer designated IP address, and the computer responds to the designated IP address for the HD2® using TCP/IP protocol. Similar to Test 1, at the time the controller starts, the CMD window initializes a ping loop to the HD2® to record the behavior of the network with respect to time (Figure 11). A visual representation of Test 2 is presented in Figure 10.

Figures 12, 13 and 14 show the comparison of real measured signals (blue), identified signals (orange) and reference signals (yellow) for position x, position y and position θ, respectively. In these figures, similar to Test 1, the blue measured real signals are covered by the

https://doi.org/10.22201/fi.25940732e.2020.21.3.026

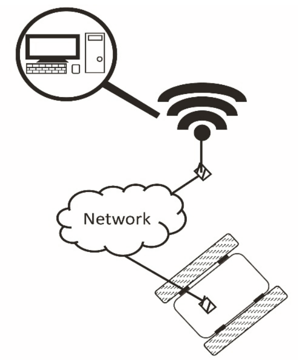

Figure 10. Graphical description of Test 2 and Test 3

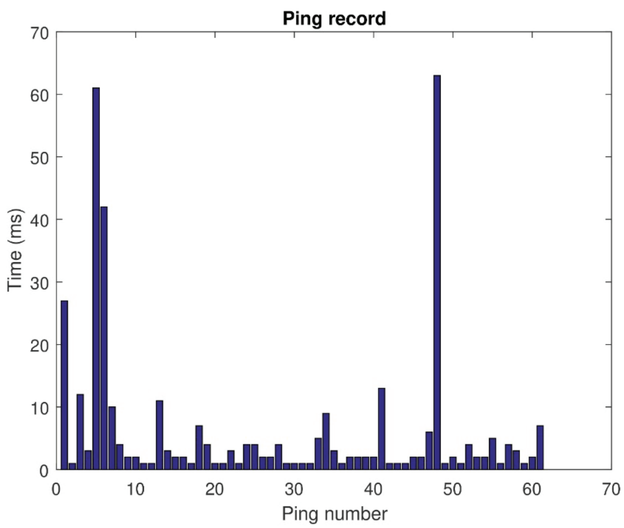

Figure 11. Ping record of Test 2. Ping statistics: Packets: Sent = 61, Received = 61, Lost = 0 (0 % Loss). Approximate round trip in milliseconds: Minimum =1 *ms*, Maximum = 63 *ms*, Media =6 *ms*

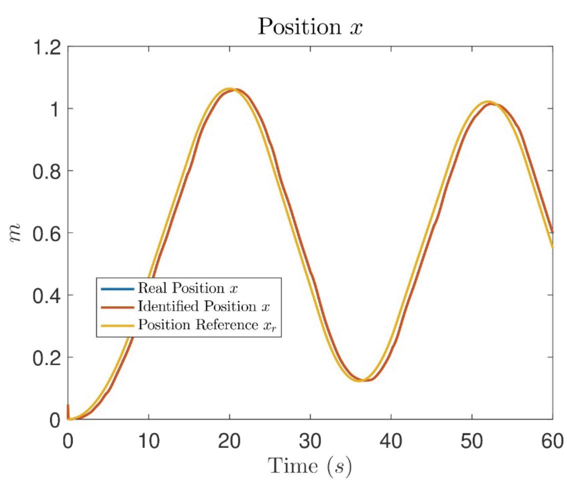

Figure 12. Comparative graph between the measured real position x, the identified $\hat{x}$ and reference x_r of Test 2

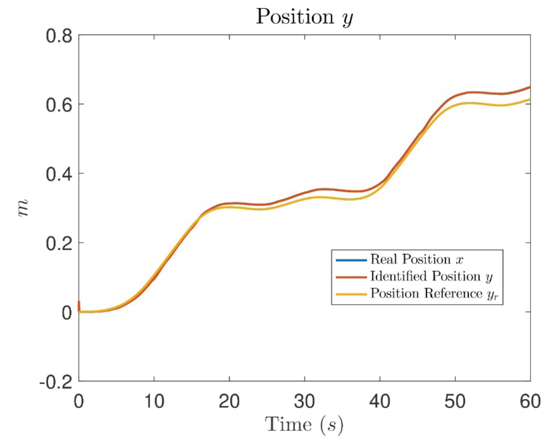

Figure 13. Comparative graph between the measured real position y, the identified $\hat{y}$ and reference y_r of Test 2

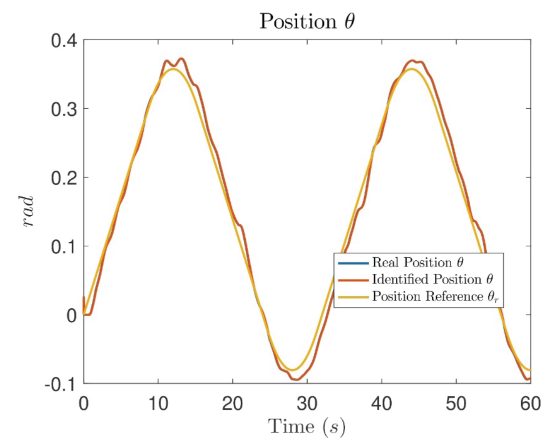

Figure 14. Comparative graph between the measured real position θ, the identified $\hat{\theta}$ and reference θ_r of Test 2

Table 3. Identification RMSE of Test 2

	RMSE
Position x	0.002336 m
Position y	0.000807 m
Position θ	0.001393 rad
Velocity v_1	0.034999 m/s
Velocity v_2	0.031415 m/s
Current i_1	0.129143 A
Current i_2	0.124773 A

orange identified signals, showing a close to zero identification error, and the tracking error between position signals and their references is small and bounded.

Table 3 shows the identification RMSEs for all the state variables of the HD2®. Moreover, Table 4 shows the tracking RMSEs, which are calculated from the identified signals and the reference signals.

Table 4. Tracking RMSE for Position x, Position y and Position θ of Test 2

	RMSE
Position x	0.037489 m
Position y	0.018244 m
Position θ	0.016095 rad

The control signals of Test 2 are shown in Figures 15 and 16, respectively.

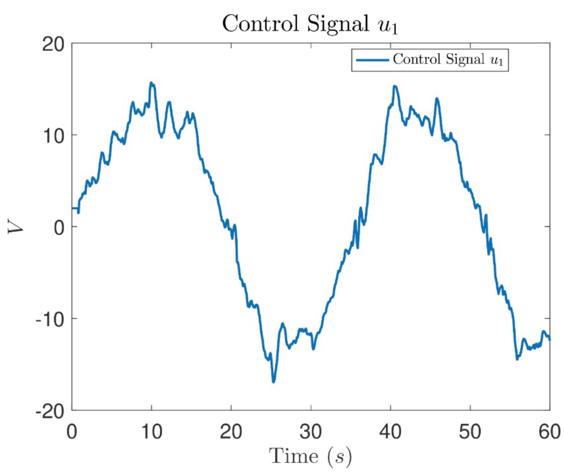

Figure 15. Control signal u_1 of Test 2

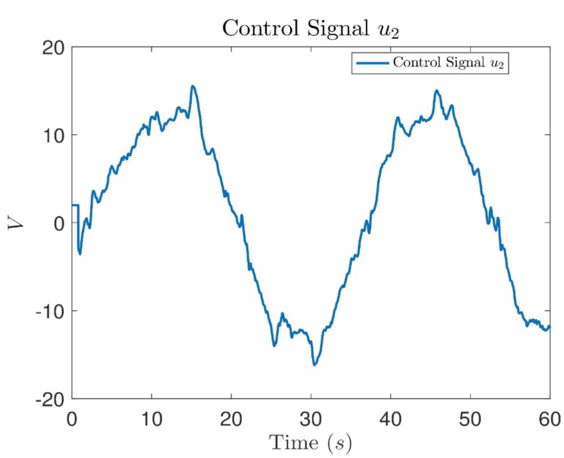

Figure 16. Control signal u_2 of Test 2

Test 3: Communication to the all-terrain tank robot through different routers

Test 3 description: Test 3 is conducted in the same way as Test 2. The difference between these two tests is the network behavior. This difference can be seen by comparing Figure 11 and Figure 17. It is a fact that the ping loop and the control systems are different programs; however, it has to be noted that they are interacting at the same time with the same devices. Figure 17 shows lost packets, which could be an indicator that in those moments, the network was more stressed.

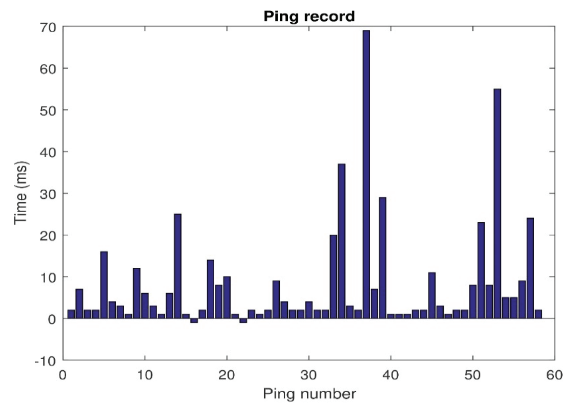

Figure 17. Ping record of Test 3. Ping statistics:
Packets: Sent = 58, Received = 56, Lost = 2 (3 % Loss).
Approximate round trip in milliseconds: Minimum = 1 ms, Maximum = 69 ms, Media = 8 ms

Figures 18, 19 and 20 show the comparison of real measured signals (blue), identified signals (orange) and reference signals (yellow) for position x, position y and position θ, respectively. In these figures, similar to Tests 1 and 2, the blue measured real signals are covered by the orange identified signals, showing a close to zero identification error, and the tracking error between position signals and their references is small and bounded.

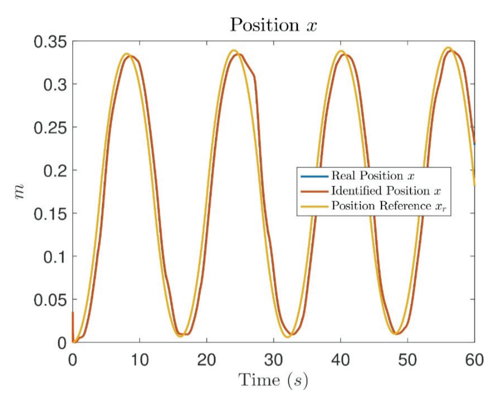

Figure 18. Comparative graph between the measured real position x, the identified $\hat{x}$ and reference x_r of Test 3

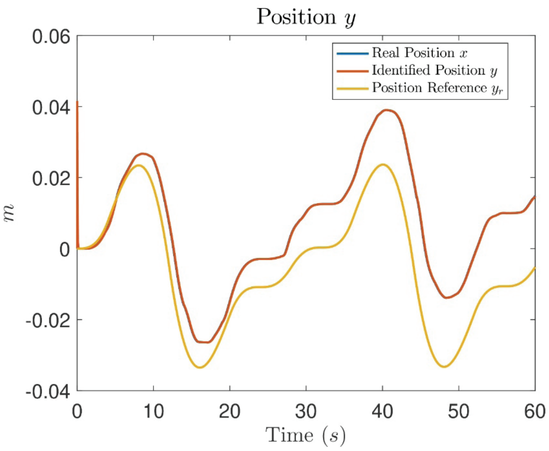

Figure 19. Comparative graph between the measured real position y, the identified $\hat{y}$ and reference y_r of Test 3

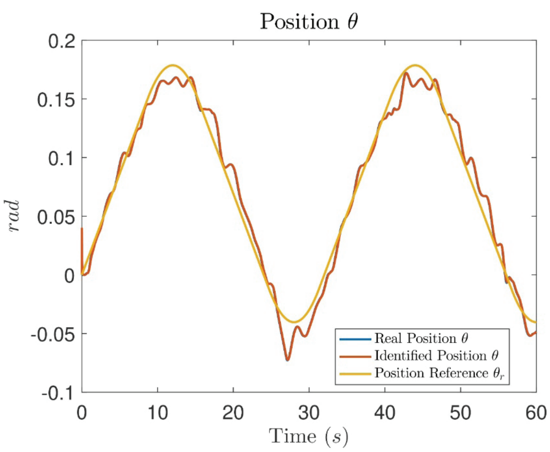

Figure 20. Comparative graph between the measured real position, the identified $\hat{\theta}$ and reference θ_r of Test 3

Table 5 shows the identification RMSEs for all the state variables of the HD2®. Moreover, Table 6 shows the tracking RMSEs, which are calculated from the identified signals and the reference signals.

Table 5. Identification RMSE of Test 3

	RMSE
Position x	0.001928 m
Position y	0.000849 m
Position θ	0.001132 rad
Velocity v_1	0.028340 m/s
Velocity v_2	0.028681 m/s
Current i_1	0.130951 A
Current i_2	0.125594 A

Table 6. Tracking RMSE for Position x, Position y and Position θ of Test 3

	RMSE
Position x	0.028207 m
Position y	0.013445 m
Position θ	0.011452 rad

The control signals of Test 3 are shown in Figures 21 and 22, respectively.

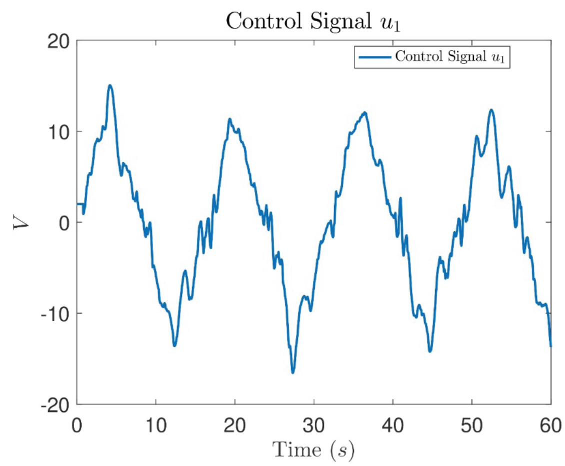

Figure 21. Control signal u_1 of Test 3

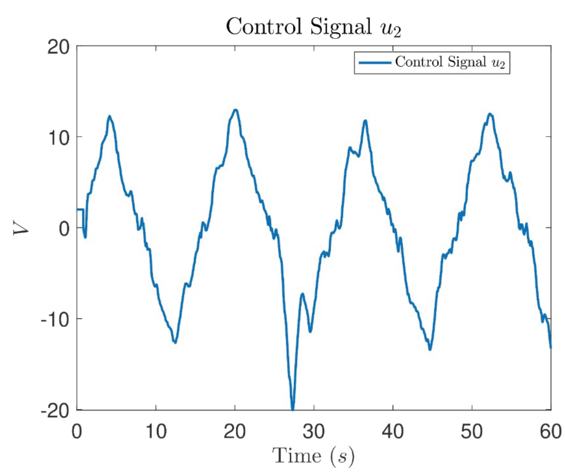

Figure 22. Control signal u_2 of Test 3

Discussion

Results presented in Sections 3.1, 3.2 and 3.3 show the effectiveness of the NIOC for trajectory tracking of unknown discrete-time nonlinear systems subject to uncertainties, disturbances, delays and packet losses, without the need of previous knowledge of models nor bounds.

As has been stated in Section 1, there are a lot of similar works reported in literature, however, most of them only have been tested with simulations, results presented in this paper are implemented in real-time for an all-terrain tracked robot controlled using a wireless network with all the communication problems that can be encounter in a real-life scenario. In Villaseñor *et al*. (2018) it is reported a NIOC for the same kind of system without consider communication problems, then the proposed NOIC represents an improvement of previous results considering communication problems introduced by the network which is a common trouble nowadays.

In this way, intelligent controllers can be a solution to alleviate this kind of problems, in fact fuzzy controllers have been used to deal with control of uncertain nonlinear systems (Aouf *et al*., 2019; Tong *el al*., 2020; Pillai & Suthakorn, 2019), however as has been stated in Jiang *et al*. (2016) neural controllers are better suited to deal with complex control task. Then it is important to remark that the use of RHONN to identify the system to be controlled allow us to use any modern control approach to deal with complex control problems as real-time trajectory tracking, this cannot be done with other intelligent controllers like fuzzy systems (Jiang *et al*., 2016).

Then in order to perform a fair comparison of the proposed controller with respect to another well-established controller that is designed with a state-space representation, that does not require previous knowledge of the exact mathematical model of the system to be controlled, we implement the real-time trajectory tracking problem for the same robot with super twisting methodology (Rios *et al*., 2017; Levant, 2011), trajectory tracking results are presented in Table 7. These results are obtained without any network problems due to this methodology do not allow us to handle this kind of problems, this issue has been encountered for the real-time implementation and they are mainly due to the chattering problem associated to sliding mode controllers, besides problems produced by communication networks do not depend of the system state therefore they cannot be compensated by the sliding mode controller. Then, the NIOC represents a better response for the problem considered in this paper.

Table 7. Tracking RMSE for Position x, Position y and Position θ.

RMSE	NIOC	Super Twisting
Position x	0.0093 m	0.03600 m
Position y	0.0069 m	0.08505 m
Position θ	0.0056 *rad*	0.0103 *rad*

This paper only considers uncertainties, disturbances, delays, and packet losses, while CPS have a lot of problems that are not considered here as: Saturation, hysteresis, backlash, friction, intrusions, attacks, faults and many others, all of them requires attention individually and as a whole system, therefore all this issues can be considered as future work, as well as implementation issues that require our attention in order to reduce time and cost for real-time applications.

Conclusion

This work presents the designing and implementation of an intelligent controller to solve on-line trajectory tracking problem of a mobile robotic system in a wireless networked environment. It is important to note, that the tests presented in this work show the performance of the NIOC scheme applied to a HD2® all-terrain robot for different network conditions, first in a direct wireless communication channel, and then, in a wireless environment with two networks. Real-time results, show that the identification performance presents errors close to zero and that the tracking performance presents small and bounded errors despite uncertainties, unknown dynamics, delays and packet losses. It is important to note that results are obtained without knowledge of the mathematical model of the HD2®, which is based on the RHONN identifier model; the control is calculated using the neural network model. In this way, the tests show how the NIOC scheme is presented as a good candidate for its robustness against unknown dynamics, unknown external and internal disturbances, unknown communication delays and packet losses.

Acknowledgments

The authors are thankful for the support of CONACYT Mexico, through Projects CB256769 and CB258068 (Project supported by Fondo Sectorial de Investigación para la Educación).

References

Alanis, A. Y. & Loukianov, A. G. (2008). *Discrete-Time high order neural control: Trained with kalman filtering*. Germany: Springer.

Alanis, A. Y., Rios, J. D., Arana-Daniel, N. & López-Franco, C. (2016). Neural identifier for unknown discrete-time nonlinear delayed systems. *Neural Computing and Applications*, 27, 2453-2464.

Alguliyev, R., Imamverdiyev, Y. & Sukhostat, L. (2018). Cyber-physical systems and their security issues. *Computers in Industry*, 100, 212-223.

Aouf, A., Boussaid, L. & Sakly, A. (2019). Same fuzzy logic controller for two-wheeled mobile robot navigation in strange environments. *Journal of Robotics*, 2019, Article ID 2465219. https://doi.org/10.1155/2019/2465219

Do, Q., Martini, B. & Choo, K-K. R. (2018). Cyberphysical systems information gathering: A smart home case study. *Computer Networks*, 138 (1), 12. https://doi.org/10.1016/j.comnet.2018.03.024

Dressler, F. (2018). Cyber physical social systems: Towards deeply integrated hybridized systems. In 2018 International Conference on Computing, Networking and Communications (ICNC), 420-424.

Farias, W. A., Freire, E. O., Givigi, S. N., Carvalho, E. A. N. & Molina, L. (2018). Comparison between fuzzy and neural controllers to cross the reality gap in evolutionary robotics. In 2018 Annual IEEE International Systems Conference (SysCon), Vancouver, BC.

Haque, S. A., Aziz, S. M. & Rahman, M. (2014). Review of cyber-physical system in healthcare. *International Journal of Distributed Sensor Networks*, 10 (4), 217415. https://doi.org/10.1155%2F2014%2F217415

Haykin, S. (2004). *Kalman filtering and neural networks. Adaptive and cognitive dynamic systems: Signal processing*, learning. Communications and Control, Wiley.

Jiang, Y., Yang, Ch. & Ma, H. (2016). A review of fuzzy logic and neural network based intelligent control design for discrete-time systems. *Discrete Dynamics in Nature and Society*, 2016, Article ID 7217364 https://doi.org/10.1155/2016/7217364

Kobayashi, K., Noritzuki, H. & Uchimura, Y. (2017). Model based predictive control for networked control system with time delay. In IECON 2017- 43rd Annual Conference of the IEEE Industrial Electronics Society, 5633-5638.

Lee E. A. (2008). Cyber physical systems: Design challenges. In 2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC), 363-369.

Levant, A. (2011). Discretization issues of high-order sliding modes. In IFAC Proceedings: 18th IFAC World Congress, 44 (1), 1904-1909. https://doi.org/10.3182/20110828-6-IT-1002.00489

Li, L., Song, J. B. & Li, H. (2017). Dynamic state aware adaptive source coding for networked control in cyberphysical systems. *IEEE Transactions on Vehicular Technology*, 66, 10000-10010. https://doi.org/10.1109/TVT.2017.2742460

López-Franco, C., Hernández-Barragán, J., Alanis, A. Y. & Arana-Daniel, N. (2018). A soft computing approach for inverse kinematics of robot manipulators. *Engineering Applications of Artificial Intelligence*, 74, 104-120.

Lv, C., Xing, Y., Zhang, J., Na, X. Li, Y., Liu, T., Cao, D. & Wang, F. Y. (2017). Levenberg-marquardt backpropagation training of multilayer neural networks for state estimation of a safety critical cyber-physical system. *IEEE Transactions on Industrial Informatics*, 1-1. https://doi.org/10.1109/TII.2017.2777460

Ochoa, S. F., Fortino, G. & Fatta, G. D. (2017). Cyber-physical systems, internet of things and big data. *Future Generation Computer Systems*, 75, 82-84.

Pillai, B. M. & Suthakorn, J. (2019). Challenges for novice developers in rough terrain rescue robots: A survey on motion control systems. *Journal of Control Science and Engineering*, 2019, Article ID 2135914. https://doi.org/10.1155/2019/2135914

Rajkumar, R., Lee I., Sha, L. & Stankovic, J. (2010). Cyber-physical systems: The next computing revolution in Design Automation Conference, 731-736. https://doi.org/10.1145/1837274.1837461

Rios, J. D., Alanis, A. Y., López-Franco, M., López-Franco, C. & Arana-Daniel, N. (2017). Real-time neural identification and inverse optimal control for a tracked robot. *Advances in Mechanical Engineering*, 9(3), 1687814017692970. https://doi.org/10.1177%2F1687814017692970

Sanchez, E. N. & Ornelas-Tellez, F. (2016). Discrete-Time inverse optimal control for nonlinear systems. EBL-Schweitzer, *CRC Press*. https://doi.org/10.1201/b14779

Selyunin, K., Ratasich, D., Bartocci, E. & Grosu, R. (2015). Deep neural programs for adaptive control in cyber-physical systems. CoRR, abs/1502.04013.

Selyunin, K., Ratasich, D., Bartocci, E., Islam, M. A., Smolka, S. A. & Grosu, R. (2015). Neural programming: Towards adaptive control in cyber-physical systems. In 2015 54th IEEE Conference on Decision and Control (CDC), 6978-6985.

Tong, M., Lin, W., Huo, X., Jin, Z. & Miao, Ch. (2020). A model-free fuzzy adaptive trajectory tracking control algorithm based on dynamic surface control. *International Journal of Advanced Robotic Systems*. https://doi.org/10.1177%2F1729881419894417

Villaseñor, C., Rios, J. D., Arana-Daniel, N., Alanis, A. Y., López-Franco C. & Hernández-Vargas, E. A. (2018). Germinal center optimization applied to neural inverse optimal control for an all-terrain tracked robot. *Applied Sciences*, 8(1).

Wiener N. (1961). *Cybernetics or control and communication in the animal and the machine*. MIT paperback series, M.I.T. Press.

Yu, Z., Ouyang, J., Li, S. & Peng, X. (2015). Formal modeling and control of cyberphysical manufacturing systems. *Advances in Mechanical Engineering*, 9(10), 1687814017725472.

Zhang, M. (2008) Artificial higher order neural networks for economics and business. IGI Global research collection, Information Science Reference.