Identification of dynamical systems through the structure of auto-regression with exogenous variable by decreasing gradient and least squares

ANDRES MOROCHO CAIZA[1], ERIK F MENDEZ GARCES[2], GABRIELA MAFLA[3], JOSEPH GUERRA[4], WILLIAMS VILLALBA[5]
[1]Escuela Superior Politecnica de Chimborazo (ESPOCH), Riobamba, ECUADOR
[2]CERN (European Organization for Nuclear Research), Geneva, SWITZERLAND
[3]Universidad Tecnica de Ambato, Ambato, ECUADOR
[4]Universidad de Cuenca, Cuenca, ECUADOR
[5]PetroEcuador, Quito, ECUADOR

Abstract: - In this article was made the identification of dynamic systems of first and second order more common in electronics such as low and high pass filters of the first order, pass-band filter and direct current motor through the structure of auto-regression with exogenous variable. The proposed dynamical systems are initially modeled by a continuous-time transfer function using physical laws. Subsequently, a step entry signal was applied and the data for the identification process was recorded in discrete time. The estimation of parameters was performed with the method of decreasing gradient and least squares. It was obtained as a result that the least squares method could not find a model for the first-order high-pass filter, but the decreasing grade method allowed to model all the proposed systems.

Key-Words: - Systems Identification, ARX, Least Squares, Decreasing Gradient, Modeling, Dynamical Systems-

Received: June 25, 2021. Revised: November 18, 2021. Accepted: December 8, 2021. Published: December 24, 2021.

1 Introduction

Dynamical systems modeling is a very vast topic to discuss and has become quite useful in the world of control. A commonly used parametric structure is auto-regression with exogenous variable (ARX) [1] and is based on least squares for the estimation of parameters, but sometimes this minimization of the mean square error does not achieve a model that fits the dynamics of the studied system. In a real system with sampled data it is not possible to achieve an ideal model but that is not the main objective because is enough with a good and simple model [2].

The ARX structure has been used in different types of modeling systems. Reference [3] presents the identification of a DC/DC converter using parametric structures. The authors note that after processing the converter signal data based on the impulse response of the input and output voltages, the ARX structure produced a better model in terms of stability for open-loop analysis. In [4] the ARX structure is applied in compression systems to make the identification from the least possible number of observations when the time-invariant linear system presents inputs with different unknown delays. The results presented in [5] and [6] demonstrate that an ARX structure is not the best option for finding a model of an electric motor. According to [7] for an inventory system of a warehouse of a distribution company, an ARX structure is also not applicable. These setbacks lead to an evaluation of the most common first and second order dynamic systems in the field of electronics [8], which is of our own interest. For this reason, the present research work studies the identification of systems such as low-pass, high-pass, band-pass and servomotor filters. Starting from known continuous models [9] of particular cases of the systems mentioned above. The data to be processed are obtained by means of the representation in state space [10] of each one of the dynamic systems.

In all the cases mentioned, the minimum square method was used for the estimation of the parameters of the structure because that method is incorporated

into the modeling software tools. This work seeks to apply another method of parameter estimation as a decreasing gradient to find a discrete-time model from the first and second-order systems initially known in continuous time. Through the comparison of experimental data, the best model in discrete time is defined.

2 Methodology

2.1 ARX Structure

In [8] is mentioned that probably the simplest input-output relationship of a system, is observed in (1) as a linear equation in differences, as well as:

$$y[n] + a_1 y[n-1] + \cdots + a_{n_a} y[n-n_a] = b_1 x[n-1] + \cdots + b_{n_b} x[n-n_b] + e[n] \quad (1)$$

Where $e[n]$ is white noise and enters as a direct error in the difference equation. A real system is not strictly causal, which means that the actual input cannot have a direct effect on the output x[n]y[n][5]. So, in modeling we work from the delayed entry. $x[n-1]$

The adjustment parameters for equation (1) are:

$$\theta = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n_a} & b_1 & b_2 & \cdots & b_{n_b} \end{bmatrix}^T \quad (2)$$

Let $\mathcal{Z}\{y[n]\} = Y(z)$, $\mathcal{Z}\{x[n]\} = X(z)$, $\mathcal{Z}\{e[n]\} = E(z)$; then when applying the transform Z to both sides of the equation (1), has:

$$\left[1 + a_1 z^{-1} + \cdots + a_{n_a} z^{-n_a}\right] Y(z) = \left[b_1 z^{-1} + \cdots + b_{n_b} z^{-n_b}\right] X(z) + E(z)$$

$$A(z) = 1 + a_1 z^{-1} + \cdots + a_{n_a} z^{-n_a}$$

$$B(z) = b_1 z^{-1} + \cdots + b_{n_b} z^{-n_b} \quad (3)$$

More compactly:

$$A(z)Y(z) = B(z)X(z) + E(z) \quad \rightarrow \quad Y(z) = \frac{B(z)}{A(z)} X(z) + \frac{1}{A(z)} E(z)$$

Or expressed similarly:

$$A(z)y[n] = B(z)x[n] + e[n] \quad \rightarrow \quad y[n] = \frac{B(z)}{A(z)} x[n] + \frac{1}{A(z)} e[n]$$

Where the following is considered:

$$G(z,\theta) = \frac{B(z)}{A(z)}, \quad H(z,\theta) = \frac{1}{A(z)} \quad (4)$$

The expression (1) is also called an ARX model (Auto-Regressive with eXogenous inputs), where AR refers to the auto-regressive part and X corresponds to the exogenous input A(z)y[n]B(z)x[n] [9][10].

Using equation (4) you can construct the following block diagram of the ARX structure.

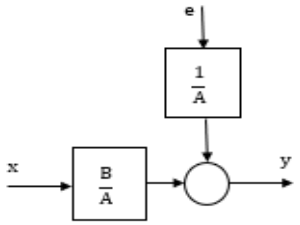

Fig.1 Structure of the ARX Model.

Fig.1 shows the structure of the ARX model where white noise passes through the dynamics of the system denominator before being added to the output. The predictor is defined by a linear regression [7], as seen in equation (5).

$$\hat{y}[n|\theta] = \theta^T \varphi[n] = \varphi^T[n]\theta \quad (5)$$

Where the vector $\varphi[n] = \begin{bmatrix} -y[n-1] & \cdots & -y[n-n_a] & x[n-1] & \cdots & x[n-n_b] \end{bmatrix}^T$.

2.2 Least Squares Method

One of the methods of parameter estimation is least squares. This method selects as a cost function the mean square error between the actual output signal and the predictor. [11]

Let $y[n]$ the output of a system in discrete time and the $\hat{y}[n]$ predictor to find the estimate of the parameters of an identification structure; then the prediction error $\varepsilon[n]$, is determined as:

$$\varepsilon[n] = y[n] - \hat{y}[n]$$

Accordingly, the cost function according to the quadratic error criterion is expressed as:

$$J = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{2} \varepsilon^2 \quad (6)$$

The least squares method is developed from linear parameterization and the quadratic criterion and its

only characteristic is to be a quadratic function in the parameter vector θ[2]. When replacing the prediction error in equation (6), it is obtained as:

$$J = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{2}[y[n] - \varphi^T[n]\theta]^2 \quad (7)$$

Equation (7) is known as the least squares criteria for the linear regression of the predictor of the ARX structure [12] and in vector form is expressed in equation (8).

Let be the vector $Y = \begin{bmatrix} y[0] \\ y[1] \\ \vdots \\ y[N] \end{bmatrix}$ and the predictor; then the cost function as a function of the vectors $\hat{Y} = \begin{bmatrix} \varphi^T[0] \\ \varphi^T[1] \\ \vdots \\ \varphi^T[N] \end{bmatrix}\theta = \Phi^T\theta e$, is represented as: $Y\hat{Y}$

$$J = \frac{1}{2N}\left(Y - \hat{Y}\right)^T \cdot \left(Y - \hat{Y}\right) = \frac{1}{2N}\left(Y - \Phi^T\theta\right)^T \cdot \left(Y - \Phi^T\theta\right) \quad (8)$$

Minimization is done by deriving the cost equation with respect to the parameter and equalizing to zero.θ

$$\frac{d}{d\theta}(J) = \frac{1}{2N}[-2\Phi^T Y + 2\Phi^T\Phi\theta] = 0 \quad \rightarrow \quad \Phi^T\Phi\theta = \Phi^T Y$$

$$\theta = \left(\Phi^T\Phi\right)^{-1}\Phi^T Y \quad (9)$$

Equation (9) allows to find the parameters of a system to be modeled under the ARX structure.

2.3 Decreasing Gradient Method

The decreasing gradient is a numerical method that aims to determine the direction of variation of the parameters that define a cost function and decreases until reaching a minimum value. The gradient algorithm can be used to find the optimal solution for quadratic optimization problems and nonlinear optimization problems. It can handle not only linear regression systems with the known information vector, but also linear and nonlinear systems with the unknown information vector. [13]

In equation (10), the decreasing gradient is defined.

$$\theta[n] = \theta[n-1] - \alpha\frac{\partial}{\partial\theta}\left(J(\theta)\right) \quad (10)$$

Where J(θ) is the cost function, is the parameter to minimize and is the learning factor, θ[n]α In equation (10) one can approximate the derivative of the cost function, that is:

$$\frac{\partial}{\partial\theta}\left(J(\theta)\right) \cong \Delta J(\theta) = J(\theta + \epsilon) - J(\theta - \epsilon)$$

Where ϵ is a very small value and sometimes they start with the same value of learning, it is the increasing cost function and it is the decreasing cost function $J(\theta + \epsilon)$ [14]. By replacing this approximation of the derivative of the cost function in equation (10); the parameter search is θ[n] defined as:

$$\theta[n] = \theta[n-1] - \alpha\,\Delta J(\theta) \quad (11)$$

The first derivative allows finding critical points, which leads to a drawback that the minimization of the cost function presents several minimums for which the initial conditions are important and thus guarantee that the best model of the system in question was obtained. It is known that an inherent characteristic of iterative search routines is that only convergence to a local solution of the problem can be guaranteed. To find the global solution, there is generally no other way than to start the iterative minimization routine at different feasible initial values and compare the results [18].

3 Results

3.1 Identification by Least Squares

A first-order system was modeled, such as the low pass filter. The input data to the system was from a step signal of amplitude 10. Fig.2 shows the actual signal and the signal modeled using the least squares method.

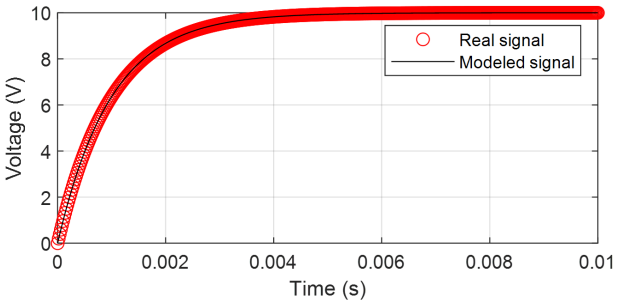

Fig.1 Output response of the model *and its corresponding signal* $G_1(s)$modeled with least squares before a stepinput.

The first order high pass filter does not have the form of equation (1) because it presents a zero in its transfer function and under this structure the least squares method did not allow to find a model that characterizes the system.

The second-order system such as the direct current servo motor was also modeled by least squares. Fig.3 shows the actual signal and the signal modeled by least squares with an input signal step of amplitude 10.

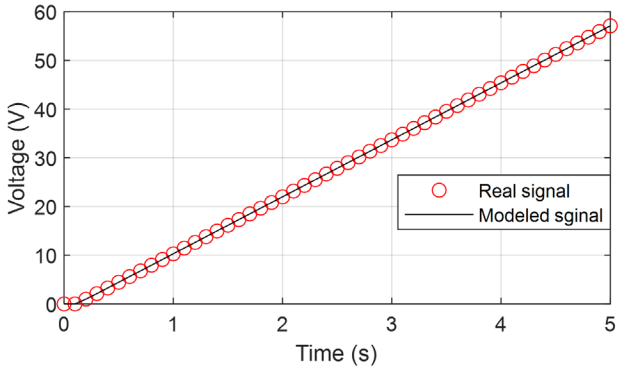

Fig.2 Output response of the model *and its corresponding signal* $G_3(s)$modeled with least squares before a stepinput.

Finally, the last second-order system corresponding to a band-passing filter is shown in Fig.4.

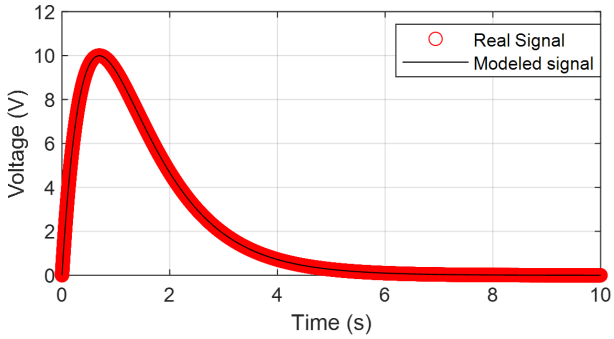

Fig.3 Output response of the model *and its corresponding signal* $G_4(s)$modeled with least squares before a stepinput.

3.2 Identification by Decreasing Gradient

The strategy for the decreasing gradient method to minimize the cost function is to establish an initial model. This model must have as small a value as possible to perform the adjustment of the parameters. This can take several minutes and even hours, the process becomes tedious when the input and output signals have been sampled at an extremely small value.

Finding an initial model with a small cost function value is essential for the algorithm to be able to work and be able to estimate the parameters of the system. The lower the number of parameters, the better for the algorithm to perform a minimization of the cost function, although it is not a rule because it can be seen in Table 1 and Table 2, the values of the initial cost function are not the same. Small enough in all cases, but still each of the systems could be modeled. Once a good initial model is achieved, the learning values "α" and the increment "ε" continue to be modified.

After modifying the values of "α" and "ε", it is checked whether the algorithm can adjust the modeled system with the actual signal. In case the model has not been adjusted, another initial model must be sought again to start the algorithm again. This whole process is carried out for each of the systems to be identified.

Fig.5 shows the actual signal along with the signal modeled using the decreasing gradient method. In addition, the evolution of the cost function can be observed and before 1600 iterations the algorithm achieves minimization.

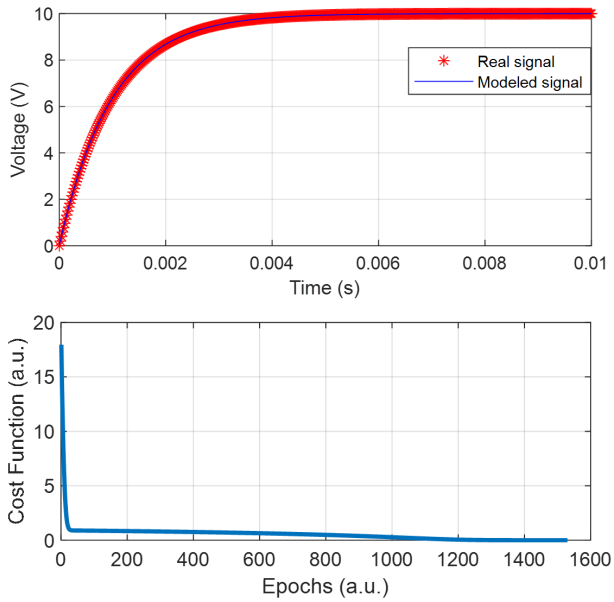

Fig.4 Output response of the model *and its corresponding signal* $G_1(s)$modeled with decreasing gradient before a step input (The upper figure is the

output response and the lower one is the evolution of the cost function).

Fig.6 shows the output of the actual signal along with the one modeled using the decreasing gradient method. This system corresponds to the high pass filter, which could be identified. In addition, you can see the evolution of the cost function and that before 100 iterations minimization is performed.

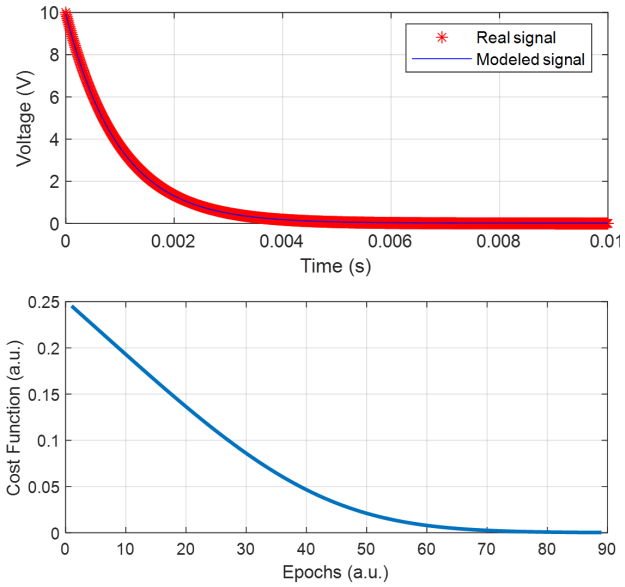

Fig.5 Output response of the model *and its corresponding signal* $G_2(s)$modeled with decreasing gradient before a step input (The upper figure is the output response and the lower one is the evolution of the cost function).

Table 1. Values for starting the decreasing gradient algorithm in first-order systems.

Model	Description	Value
$G_1(s) = \frac{1000}{s + 1000}$	Initial parameter vector	$\theta = [0.01, 0.02, 1, -0.9]$
	Learning	$\propto = 0.13$
	Increase	$\varepsilon = 10^{-5}$
	Sampling time	$t_s = 0.00002\ [s]$
	Number of iterations	1530
	Value of the initial cost function	$J_o = 19.993$
	Value of the final cost function	$J_f = 3.7(10^{-5})$
$G_2(s) = \frac{s}{s + 1000}$	Initial parameter vector	$\theta = [0.991, -0.991, 1, -0.982]$
	Learning	$\propto = 0.1$
	Increase	$\varepsilon = 10^{-5}$
	Sampling time	$t_s = 0.00001\ [s]$
	Number of iterations	89
	Value of the initial cost function	$J_o = 0.3264$
	Value of the final cost function	$J_f = 2.6(10^{-4})$

Table 1 shows the values to start the decreasing gradient algorithm for first-order systems, the same ones that correspond to a low pass and high pass filter.

In the next figure (Fig. 7) shows the output of the actual model and the output of the decreasing gradient method for the direct current servo motor. In addition, you can see the evolution of the cost function and before reaching 160 iterations you manage to minimize the cost function.

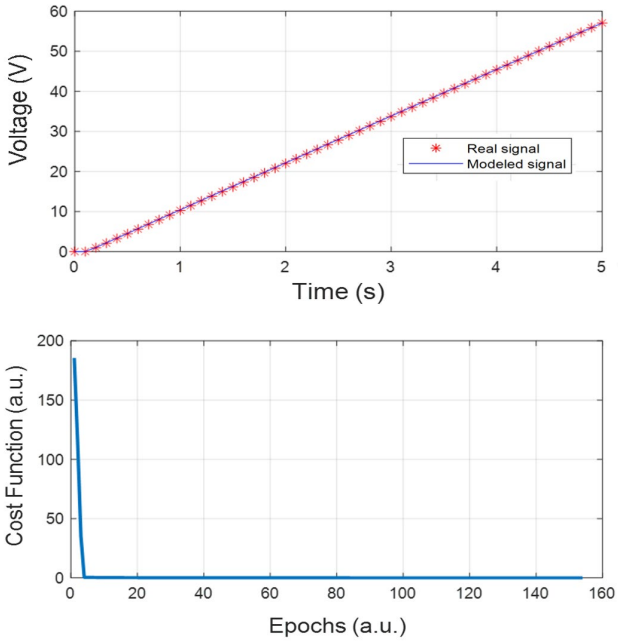

Fig.6 Output response of the model *and its corresponding signal* $G_3(s)$modeled with decreasing gradient before a step input (The upper figure is the output response and the lower one is the evolution of the cost function).

In the next figure (Fig. 8) shows the output of the actual model and the output of the decreasing gradient method for the direct current servo motor. In addition, you can see the evolution of the cost

function or in its minimization process and before reaching 300 iterations the method finds the parameters of the modeled system.

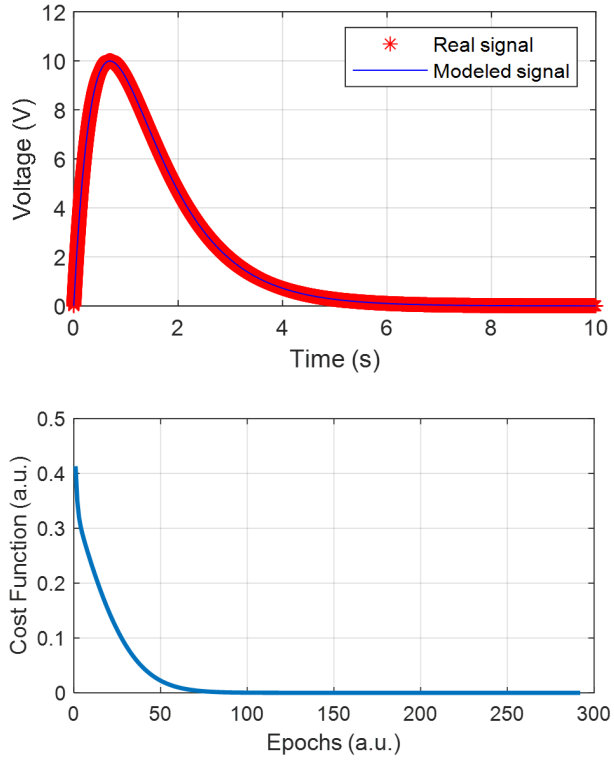

Fig.7 Output response of the model *and its corresponding signal* $G_3(s)$modeled with decreasing gradient before a step input (The upper figure is the output response and the lower one is the evolution of the cost function).

Table 2. Values for starting the decreasing gradient algorithm in second-order systems.

Model	Description	Value
$G_3(s) = \frac{1.17}{s(0.12s+1)}$	Initial parameter vector	$\theta = [0.09, -1-1, 0.16]$
	Learning	$\propto = 0.01$
	Increase	$\varepsilon = 10^{-4}$
	Sampling time	$t_s = 0.1\ [s]$
	Number of iterations	154
	Value of the initial cost function	$J_0 = 237.79$
	Value of the final cost function	$J_f = 0.006968$
$G_4(s) = \frac{4s}{s^2 + 3s + 2}$	Initial parameter vector	$\theta = [0.0004, -0.0004, -1.99969971, 0.99969974]$
	Learning	$\propto = 10^{-8}$
	Increase	$\varepsilon = 10^{-10}$
	Sampling time	$t_s = 0.0001\ [s]$
	Number of iterations	0.5614
	Value of the initial cost function	$J_0 = 0.5614$
	Value of the final cost function	$J_f = 2.77(10^{-6})$

Table 2 shows the values for starting the decreasing gradient algorithm for second-order systems, which correspond to a direct current servo motor and a band-passing filter.

Getting a very low initial cost function value in the order of the units or less, as can be seen in Table 1 and Table 2 for the models of the three filters, does not establish a rule to get a good model because the value of the initial cost function of the engine reached a very high value in the order of hundreds and the algorithm of the decreasing gradient found a consistent model.

4 Conclusions

The systems proposed are based on a transfer function in continuous time and from this, the data acquisition for the identification process was carried out. The selected structure was auto-regression with exogenous variable and the coefficients were found by means of least squares and decreasing gradient. The least squares estimation was perfect, without errors, this is because it was based on an ideal model in continuous time; For a real system, it is not common to find a perfect model in its identification process.

However, for the high pass filter it was not possible to find a model because in an ARX structure for a first order model it does not admit the presence of zeros in the transfer function because it is a causal system.

On the other hand, the decreasing gradient method allowed to characterize all the proposed systems, but with the presence of a small error. It is worth mentioning that it took a bit of difficulty to find the initial model for your programming algorithm. As the model increases in order, the difficulty of finding an initial model also increases because there are more parameters to consider.

A perfect combination would be to start modeling a real system with least squares and consider this model as a starting point for decreasing gradient and thus improve the model found.

References:

[1] S. Kolumbán and I. Vajk, "Using semidefinite programming for L1 identification of the ARX model," 11th IEEE Int. Symp. Comput. Intell. Informatics, CINTI 2010 - Proc., pp. 75–80, 2010, doi: 10.1109/CINTI.2010.5672269.

[2] L. Ljung, "Perspectives on System Identification," Annu. Rev. Control, vol. 34, pp. 1–17, 2010, [Online]. Available: http://www.control.isy.liu.se.

[3] M. A. N. Amran, A. A. Bakar, M. H. A. Jalil, M. U. Wahyu, and A. F. H. A. Gani, "Simulation and modeling of two-level DC/DC boost converter using ARX, ARMAX, and OE model structures," Indones. J. Electr. Eng. Comput. Sci., vol. 18, no. 3, pp. 1172–1179, 2020, doi: 10.11591/ijeecs.v18.i3.pp1172-1179.

[4] B. M. Sanandaji, T. L. Vincent, M. B. Wakin, R. Tóth, and K. Poolla, "Compressive System Identification of LTI and LTV ARX Models," IEEE Conf. Decis. Control Eur. Control Conf., vol. 50, pp. 791–798, 2011, doi: 978-1-61284-801-3/11.

[5] F. Piltan, S. T. Haghighi, and N. B. Sulaiman, "Comparative study between ARX and ARMAX system identification," Int. J. Intell. Syst. Appl., vol. 9, no. 2, pp. 25–34, 2017, doi: 10.5815/ijisa.2017.02.04.

[6] M. Salah-Eddine, S. Sadki, and B. Bensassi, "Microcontroller based data acquisition and system identification of a DC servo motor using ARX, ARMAX, OE, and BJ models," Adv. Sci. Technol. Eng. Syst., vol. 5, no. 6, pp. 507–513, 2020, doi: 10.25046/aj050660.

[7] S. Rachad, B. Nsiri, and B. Bensassi, "System Identification of Inventory System Using ARX and ARMAX Models," Int. J. Control Autom., vol. 8, no. 12, pp. 283–294, 2015, doi: 10.14257/ijca.2015.8.12.26.

[8] J. C. Gómez, "Identificación de Sistemas Identificación de Sistemas," p. 7, 2005.

[9] K. Uribe-Murcia and Y. S. Shmaliy, "UFIR Filtering under Uncertain One-Step Delayed and Missing Data," WSEAS Transactions on Signal Processing, pp. 191–196, 2020.

[10] C. T. Leondes, Control and Dynamic System: Advances in Theory and Applications, vol. 25, no. P1. New York: Academic Press, 1987.

[11] L. Ljung, System identification: Theory for the User, Primera. New Jersey: Prentice Hall, 1987.

[12] C. Kunusch, "Identificacion De Sistemas Dinamicos," Universidad Nacional de la Plata, 2003.

[13] K. Yankov, "Data Structure of Models in System Identification," Proc. Int. Conf. Inf. Technol., no. September, pp. 312–328, 2013.

[14] D. Abad, M. Asitimbay, and W. Navas, "Identificación y Modelado Matemático de un Motor mediante Método Grafico y estructura de modelo ARX," 2017.

[15] M. Galrinho, N. Everitt, and H. Hjalmarsson, "ARX modeling of unstable linear systems," Automatica, vol. 75, pp. 167–171, 2017, doi: 10.1016/j.automatica.2016.09.041.

[16] F. Ding, J. Pan, A. Alsaedi, and T. Hayat, "Gradient-based iterative parameter estimation algorithms for dynamical systems from observation data," Mathematics, vol. 7, no. 5, 2019, doi: 10.3390/math7050428.

[17] A. Morocho, "Desarrollo de un Regulador Neurodifuso Adaptativo para el Control de una Planta de Vuelo Vertical de National Instrument," Escuela Superior Politécnica de Chimborazo, 2018.

[18] C. T. Leondes, Control and Dynamic Systems: Advances in Theory and Applications, vol. 27, no. P3. New York, 1988.

Contribution of individual authors to the creation of a scientific article (ghostwriting policy)

All the authors have participated in the implementation and simulation of the dynamic systems, in the same way they read and approved the final manuscript, assuming full responsibility for the content of the article.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)