A Linear Identification Method Applied to a Quadcopter Drone for Stochastic Control Algorithms ⋆

Carlos E. D. Nogueira * **Antonio S. Silveira** **
Paulo V. D. Nogueira *** **Lucas de C. Sodré** ****

* *Faculdade de Engenharia Elétrica e Biomédica, Universidade Federal do Pará, PA, (e-mail: carlos.nogueira@itec.ufpa.br).*
** *Faculdade de Engenharia Elétrica e Biomédica, Universidade Federal do Pará, PA, (e-mail: asilveira@ufpa.br).*
*** *Faculdade de Engenharia Elétrica e Biomédica, Universidade Federal do Pará, PA, (e-mail: paulo.nogueira@ufpa.br).*
**** *Faculdade de Engenharia Elétrica e Biomédica, Universidade Federal do Pará, PA, (e-mail: lucas.sodre@ufpa.br).*

Abstract: This article addresses the challenge of identifying linear systems for mathematical modeling of a quadcopter drone, with the purpose of utilizing them in predictive and stochastic control algorithms. The approach involves the application of two identification methods, namely the Ordinary Least Squares for ARX systems and the Extended Least Squares for ARMAX systems. The findings indicate that the ARMAX models exhibit superior performance indices and are deemed more appropriate for controller projects.

Keywords: ARMAX Model, Quadcopter Drone, Linear Systems Identification, Stochastic Control, Extended Least Squares.

1. INTRODUCTION

Control, stabilization, and trajectory tracking of drones are areas of significant interest in science and technology, demonstrated by numerous applications in military (Springer, 1954), civil defense and disaster prevention (Daud et al., 2022), agriculture (Mogili and Deepak, 2018), and other fields . Thus, it is essential to employ various control techniques, including Model Predictive Control (MPC), which has demonstrated significant success in the literature.

To achieve their objectives, effective model-based controllers rely on accurate mathematical descriptions. Therefore, it is crucial to use robust control algorithms and appropriate identification methods to ensure that the controlled system's performance indices are not compromised. The movement dynamics of a quadcopter make it a complex system, requiring careful design of control algorithms. To design controllers based on models efficiently, a mathematically accurate description that represents the process's dynamics is necessary. Identification can also provide a better understanding of the system's dynamics, enabling the designer to use virtual simulations of various control algorithms and have a broader view of the problem's scope (Patwardhan and Shah, 2002).

Numerous scientific works have been produced worldwide to develop accurate models that describe the dynamics of quadcopter drones. In Lopes (2013), two models were developed and compared for the Parrot AR Drone: a phenomenological model (or white box model) and an Auto Regressive with Exogenous Inputs (ARX) model, both created using Ordinary Least Squares (OLS). The study aimed to represent the drone dynamics appropriately, and it was concluded that the models achieved were valid for simulations. Although there are couplings, the system can be controlled in a decoupled way in the specific case of the AR Drone.

In Hernandez (2013), the pitch and altitude dynamics were identified using an Error Prediction Method (EPM) with different sampling times for each dynamic. The study proposed some models for simulation of drones. However, the approach presented in this paper encompasses all dynamics in a single sampling period to simplify the treatment of problems related to transport delay, which can suffer fluctuations due to wireless communication. Additionally, the Extended Least Squares (ELS) algorithm was used, which performs stochastic modeling to obtain better parameterized models than in Lopes (2013).

The main objective of this paper is to explore and implement identification algorithms for quadcopters to design controllers, ranging from simple Proportional-Integral-Derivative (PID) controllers to complex Stochastic Model Predictive Controllers. This local scientific problem is addressed in this study, providing tools that can be used globally by any reader utilizing the models presented here. Additionally, the techniques demonstrated in this article allow an analysis of robustness through gain and phase

⋆ This work was carried out with the support of CNPq Proc. 140424/2022-9.

margins, predictability of dynamics, and disturbance behavior.

Moreover, this article serves as an excellent literature review, where the difference equations are applied in experimental environments, highlighting the challenges faced in the control and automation field. The study provides a realistic approach to the problems faced in practice, which cannot be entirely addressed through simulations.

2. DRONE QUADCOPTER SYSTEM

This section introduces information about the AR Drone Quadcopter and the experimental environment of the tests. Then, in sequence, the approach to identifying the model is presented.

2.1 Parrot AR Drone

The Parrot AR. Drone 2.0 is an Unmanned Aerial Vehicle (UAV) that was used in this work. This UAV has four engines coupled to propellers equally spaced from the central axis. Its data processing and transmission system has an ARM Cortex A8 32-bit 1 GHz processor, inertial navigation sensors – accelerometer, gyrometer, barometer, ultrasound (for low altitudes) and altimeter (for higher altitudes) –, cameras 720-pixel HD resolution and communication via 802.11 IEEE (WiFi™) protocols with a range of up to 50 meters without obstacles (Parrot, 2014).

In order to conduct the experiments, it is not a requirement to integrate controllers into the microprocessor of the drone. Instead, the algorithms are processed on a personal computer and only wireless signals are transmitted. The transmission and reception of signals utilize UDP (User Datagram Protocol) without the use of handshake, which is a term associated with Transmission Control Protocol (TCP) for signal reception conference. This approach prioritizes communication speed and is unconcerned with the potential loss of information, a characteristic of UDP. Thus, if hosts lose contact, the buffers located within the drone will function based on the last reference received. Additionally, the AR drone contains autonomous flight safety mechanisms that activate in the event of persistent errors.

The AR Drone version 2.0 facilitates "real-time" communication (with apologies for using a non-rigorous term) between the drone and the processing unit responsible for executing command algorithms. The term "real-time" refers to the attempt to achieve immediate signal transmission and reception from the drone. However, a complete loop of sending and receiving data occurs within a time interval of 0.065 second, which corresponds to the minimum Sampling Time T_s that should be utilized for controller design. This T_s naturally arises from the transport delay of the WiFi technology utilized in the communication between the computer and the AR Drone (Sanabria, 2013).

2.2 Quadcopter Drone Dynamics

Fig. 1 depicts the dynamics of the quadcopter, highlighting the six degrees of freedom (6-DOF) [1] associated with its movement. According to the literature by Stevens et al. (2016), the Drone's rotation around the X, Y, and Z axes corresponds to the Roll, Pitch, and Yaw angles, respectively, which are referred to as Euler Angles. Clockwise rotation of these angles yields positive angular velocities (in accordance with the right-hand rule, whereby the thumb points in the direction of the axis and the remaining fingers indicate the direction of rotation), whereas anti-clockwise rotation results in negative velocities.

[1] sideways, forwards, backwards, up and down and rotating around the 3 XYZ axes.

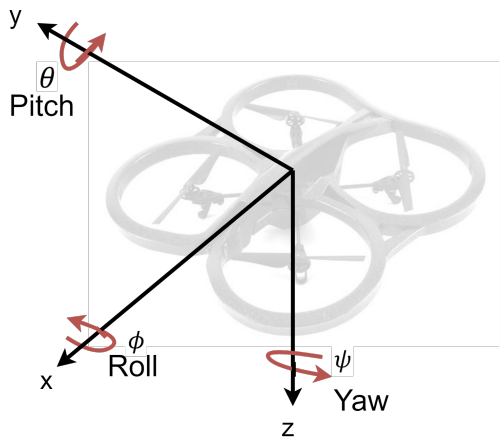

Fig. 1. Guidance of the AR Drone dynamics following the NED (North-East-Down) system.

Fig. 2. Devices used for experimental tests.

For the purpose of this work, we will utilize four degrees of freedom to identify the dynamics of Attitude and Altitude. As such, we will not be identifying the locations in the XY plane, but instead, we will solely rely on the data provided by inertial (accelerometer and gyrometer) and ultrasonic sensors.

2.3 Framework and Virtual Environment

The data acquisition involved the interconnection of three devices, as depicted in Fig. 2: a joystick that was connected to computer and controlled by the user, and was used to direct the drone to avoid indoor collisions and also to stimulate all the dynamics relevant to identification. The drone and computer were linked through WiFi to enable signal transmission and reception.

To carry out the data acquisition tests, the Toolbox AR Drone Simulink Development-Kit v1.13 provided by

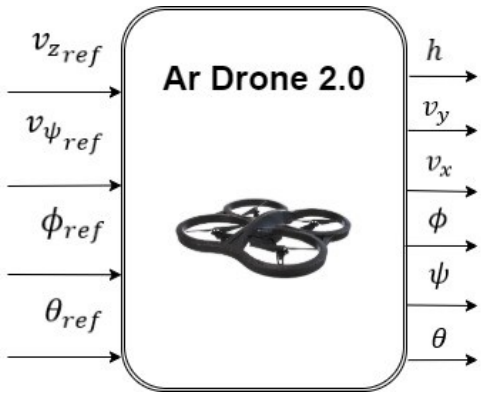

Fig. 3. Input and Output Signals for AR Drone Data Acquisition.

Sanabria (2013) is used, where the signals available for sending via WiFi are the Vertical Velocity Commands $v_{z_{ref}}$, Yaw Angle Rate Commands $v_{\Psi_{ref}}$, Pitch Angle Reference θ_{ref} and Roll Angle Reference φ_{ref}. The values coming from the drone sensors are: Altitude h, Pitch Angle θ_{ref}, Roll Angle ϕ and Yaw Angle ψ. Didactically, the signals can be visualized in Fig. 3. Note that the estimated velocity signals on the X axis v_x and Y axis v_y are not used in this work.

3. SYSTEM IDENTIFICATION

In this section, the AR Drone will be analyzed as a Linear Time Invariant (LTI) system with Multiple Inputs and Multiple Outputs (MIMO). The system's dynamics of interest can be decoupled and treated as the intersection of several Single Input Single Output (SISO) systems, allowing for the development of stochastic models of the plant that can be used to design controllers. To achieve this, the Extended Least Squares (ELS) algorithm will be utilized, which aims to eliminate polarization in the models estimated by the purely deterministic Ordinary Least Squares (OLS) algorithm. This decision is based on the need to improve the accuracy of the models.

3.1 Time Series Models

There are several families of models to represent dynamic processes. A very comprehensive example can be described by

$$A\left(q^{-1}\right)y(k) = B\left(q^{-1}\right)u(k)q^{-d} + C\left(q^{-1}\right)\frac{\zeta(k)}{\Delta} \qquad (1)$$

where $\Delta = 1$.

The aforementioned system is referred to as an ARMAX linear model, which comprises the following components: ζ, which represents the Gaussian noise; d, the discrete transport delay; $k\epsilon N$, the discrete time instant; and q^{-1}, the discrete time shift operator derived from the Transform Z^{+}. It is assumed that the system is causal for the values of k, and the initial values are null [2] (Åström and Wittenmark, 2013). The system's input and output are denoted by u and y, respectively. The model polynomials that multiply, respectively, the output, input, and noise of the system are expressed as

$$\begin{aligned} A\left(q^{-1}\right) &= 1 + a_1q^{-1} + a_2q^{-2} + \cdots + a_{n_a}q^{-n_a} \\ B\left(q^{-1}\right) &= b_0 + b_1q^{-1} + b_2q^{-2} + \cdots + b_{n_b}q^{-n_b} \\ C\left(q^{-1}\right) &= 1 + c_1q^{-1} + c_2q^{-2} + \cdots + c_{n_c}q^{-n_c} \end{aligned} \qquad (2)$$

It is crucial to consider that in a SISO LTI system, the number of roots of the polynomials $B(q^{-1})$ and $C(q^{-1})$ should not exceed the number of roots of $A(q^{-1})$ to maintain the system's causality. Failure to meet this condition may result in impairment to the causality of the system.

3.2 Ordinary Least Squares

While there is no definitive agreement on the originator of the Least Squares algorithm, it is commonly attributed to the notable contributions of Karl Friedrich Gauss in the late 18th century. This method involves minimizing the difference between estimated and actual values to reduce estimation errors, enabling the estimated linear parameters to closely align with those of the actual plant parameters (Ljung, 1987).

Let the Vector of Regressors be given in

$$\varphi^T(k) = \begin{bmatrix} -y(k-1) \\ \vdots \\ -y(k-n_a) \\ u(k-d) \\ \vdots \\ u(k-d-n_b) \end{bmatrix} \qquad (3)$$

and the Vector of real Parameters of the system in

$$\vartheta^T = [\, a_1 \;\; a_2 \;\; \ldots \;\; a_{n_a} \;\; b_0 \;\; b_1 \;\; b_{n_b} \,], \qquad (4)$$

The actual output of the system is represented by

$$y(k) = \varphi^T(k)\vartheta + e(k) \qquad (5)$$

which in matrix form is written as

$$\underbrace{\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(N-1) \end{bmatrix}}_{Y} = \underbrace{\begin{bmatrix} \varphi^T(0) \\ \varphi^T(1) \\ \vdots \\ \varphi^T(N-1) \end{bmatrix}}_{\Phi} + \underbrace{\begin{bmatrix} e(0) \\ e(1) \\ \vdots \\ e(N-1) \end{bmatrix}}_{E} \qquad (6)$$

$$Y = \Phi\vartheta + E$$

where the variable $e(k)$ is an unknown noise.

Let ϑ be the Plant's Estimated Parameters Vector, the objective is to minimize the quadratic estimation error represented in

$$\min_{\hat{\theta}} |Y - \Phi\hat{\vartheta}|^2, \qquad (7)$$

whose algebraic manipulation results in the solution in

$$\hat{\vartheta} = \left[\Phi^T\Phi\right]^{-1}\Phi^T Y. \qquad (8)$$

Therefore, the estimated output of the system can be written as

$$\hat{y}(k) = \varphi^T(k)\hat{\vartheta}(k) + \xi(k), \qquad (9)$$

where $\xi(k)$ are the residuals from the estimation.

[2] The discrete-time operator belongs to the discrete-time domain and is a simplification for the following mathematical operation: $u(k)q^{-d} = u(k-d)$.

3.3 Extended Least Squares

The ELS tries to explain some bias that can be present when the OLS fails to represent the behavior of the system well. For this, the output residuals from the estimation in

$$\xi(k,i) = y(k) - \hat{y}(k,i) \tag{10}$$

are analyzed. The presence of polarization in a system indicates a strong correlation between the system's output and its residuals. To account for such polarization, it is necessary to model the residuals and identify any Moving Average polarizations, resulting in a more accurate representation of the system's behavior. This probabilistic modeling approach aims to improve the consistency between the model and the process behavior, moving beyond the limitations of a deterministic model.

Adaptations made to the OLS algorithm for obtain ELS can be viewed in

$$\varphi(k,i) = \begin{bmatrix} -y(k-1) \\ \vdots \\ -y(k-n_a) \\ u(k-d) \\ \vdots \\ u(k-d-n_b) \\ \xi(k-1,i) \\ \vdots \\ \xi(k-n_c,i) \end{bmatrix}$$

$$\vartheta^T(i) = [\, a_1 \;\; a_2 \;\; \ldots \;\; a_{n_a} \;\; b_0 \;\; b_1 \;\; \ldots \;\; b_{n_b} \;\; c_1 \;\; \ldots \;\; c_{n_c} \,]$$

$$y(k) = \varphi^T(k,i)\vartheta(i) + e(k) \tag{11}$$

$$\underbrace{\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(N-1) \end{bmatrix}}_{Y} = \underbrace{\begin{bmatrix} \varphi^T(0,i) \\ \varphi^T(1,i) \\ \vdots \\ \varphi^T(N-1,i) \end{bmatrix}}_{\Phi} \theta + \underbrace{\begin{bmatrix} e(0) \\ e(1) \\ \vdots \\ e(N-1) \end{bmatrix}}_{E}$$

$$Y = \Phi\vartheta + E$$

$$\min_{\hat{\theta}} |Y - \Phi\hat{\vartheta}|^2$$

$$\hat{\vartheta} = \left[\Phi^T\Phi\right]^{-1}\Phi^T Y.$$

It should be noted that the algorithm includes the polynomial $C(q^{-1})$ in the Parameter Vector. In addition, the residuals resulting from the estimation error and the variable $y(k)$ are incorporated into the Regressor Vector.

Another notable difference is that the estimated output is now dependent on a parameter "i", which represents the number of times the algorithm will be computed. Specifically, in a programming loop that is repeated i-times, the Parameter Vector will be updated i-times until the algorithm converges, provided that convergence is achievable.

Subsequently, the residuals are recalculated as shown in 10 until convergence is attained. Typically, selecting a value of i within the range of 7 to 30 is satisfactory to observe convergence, as reported by Aguirre (2007).

4. RESULTS AND DISCUSSION

The obtained results were subjected to quantitative and qualitative analysis through performance indices, which included the J_{MCC} (Multiple Correlation Coefficient), graphical analysis of transient response, and examination of pole and zero locations. The expectation was that the mathematical models developed using the employed algorithms would possess adequate capacity for representing the dynamics relevant to the quadcopter. This, in turn, would enable more accurate simulation of controller implementation strategies that are closer to the system's reality. Details regarding this matter are elaborated in the following subsections.

To enable the implementation of a decentralized identification strategy, several signal acquisitions were performed. Given that the drone has four inputs, four tests were conducted, with each one exciting a single input while the others were held at zero. Once a significant amount of data was obtained, the correlation function between the input and output variables of all tests was analyzed. It was determined that the correlation was negligible, with a confidence margin of 95%. It is important to note that, while the high performance of controllers in most quadcopters is typically associated with couplings, the literature suggests that there are acceptable margins for decentralized control strategies when stabilizing the AR Drone 2.0. Furthermore, this does not imply significant losses from a control systems perspective, given the constructive nature of the quadcopter.

The results are presented separately to emphasize the unique characteristics of each dynamic. The primary focus of the comparisons was to evaluate the benefits of the stochastic identification of the system. Therefore, three key signals were showcased: the estimated output of the ARX model, the estimated output of the ARMAX model, and the actual output of the plant. Additionally, the input signal for each dynamic was plotted to determine if the data acquisition was consistent with recommended identification techniques and if the plant's response behavior matched the surveyed models.

Furthermore, the model parameters associated with the drone dynamics in the ARMAX model in (2) are shown in table 1.

Table 1. Values of the parameters of the estimated models.

Parameters	Altitude	Roll	Pitch	Yaw
d	3	3	3	3
a_1	-1.696	-1.471	-1.562	-1.664
a_2	0.6931	0.5828	0.6735	0.6669
b_0	0.012	0.03727	0.03811	0.02065
c_1	-0.01045	0.1092	0.04269	0.008062

For the selection of how many regressors can represent a good linear approximation for a plant model, was used the Akaike Information Criterion (AIC) (Akaike, 1974) shown in

$$AIC(n_\vartheta) = N ln(\sigma_e^2(n_\vartheta)) + 2n_\vartheta \tag{12}$$

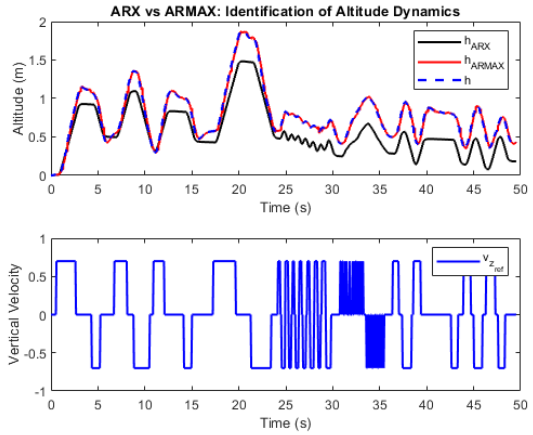

Fig. 4. A comparison between actual and estimated drone altitude dynamics responses.

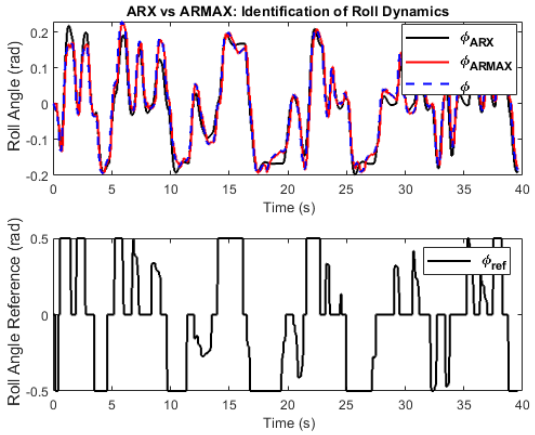

Fig. 5. A comparison between actual and estimated drone roll dynamics responses.

where n_ϑ is the number of regressors; $\sigma_e^2(\vartheta)$ is the variance of the residual which is a function of n_ϑ; and N is the size of the sampled data.

4.1 Altitude

The quadcopter's altitude is controlled through vertical velocity commands that are determined by the magnitude of the joystick's analog buttons. It is important to ensure that the drone does not ascend too high and collide with the ceiling or descend too low, which could cause turbulent non-linearities due to the wind generated by the propellers. Notably, external disturbances like wind were not encountered during the indoor tests.

The accuracy of the ARMAX model in representing the actual output of the system was confirmed by the transient response depicted in Fig. 4, and the convergence of the ELS algorithm.

4.2 Roll and Pitch

The roll and pitch dynamics of the drone are regulated by angular reference signals, where the desired angle in radians is set as a reference and transmitted to the system. The drone measures its inertial sensors to return the achieved values, as the reference and output values are in the same physical quantity.

Fig. 5 and Fig. 6 depict the estimated system responses for the roll and pitch dynamics, respectively. While the ARX model appears appropriate based on visual analysis, the cross-correlation function of the residuals with the actual output showed a polarization exceeding the 95% significance level. Thus, the ARMAX model is recommended to eliminate this bias.

4.3 Yaw

During experimental tests, it was observed that the clockwise and counterclockwise rotation commands for the drone resulted in different changes in the yaw angle magnitude, demonstrating non-linearity in the system. Fig. 7 confirms this observation, showing a clockwise tendency in the drone's movement. This non-linearity was also evident

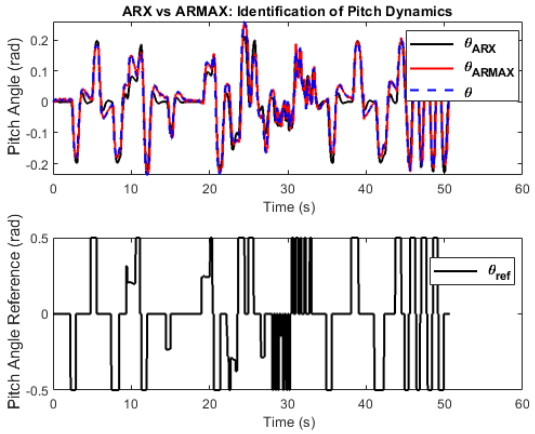

Fig. 6. A comparison between actual and estimated drone pitch dynamics responses.

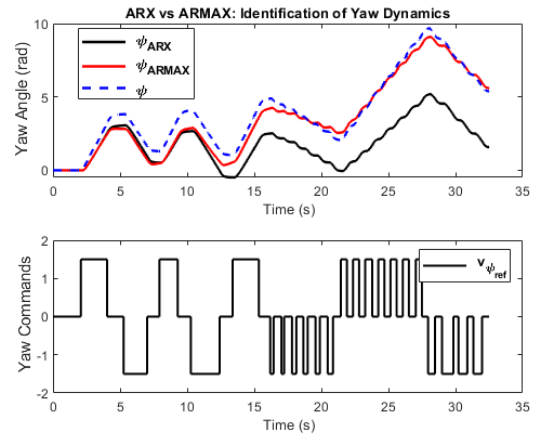

Fig. 7. A comparison between actual and estimated drone yaw dynamics responses.

in tests utilizing a Pseudo Random Binary Signal (PRBS) with zero mean, as the drone did not maintain its centralized angular position and varied significantly in both directions of rotation.

As this article's objectives do not address special treatments for non-linearities, a linear model was developed to stabilize the drone's glide. The designed linear controllers must be robust enough to accomplish this task. To achieve this, a linear identification approach was utilized, which represents an average between the two directions of rotation. However, this article opens the possibility for future studies to use non-linear controllers through other techniques, such as Artificial Neural Networks and Fuzzy Logic.

4.4 Indexes for comparison between models

To quantitatively and comparatively judge the quality of the estimated models, two indices will be used: the J_{SEQ} (Sum of Squared Error) given by

$$J_{SEQ} = \sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2 \tag{13}$$

and the J_{MCC} (Multiple Correlation Coefficient) given by

$$J_{CCM} = 1 - \frac{J_{SEQ}}{\sum_{k=1}^{N} [y(k) - \bar{y}]^2} \tag{14}$$

where $\bar{y}$ is the average actual output, $y(k)$ is the estimated output, and N is the total number of samples.

The J_{SEQ} is calculated as the sum of the squared error between the actual output and the output estimated by the OLS and ELS algorithms. A smaller J_{SEQ} value indicates that the model is closer to the real values of the system, making it more appropriate. Therefore, J_{SEQ} is used as a comparative tool to identify the algorithm that produces the closest results to the real system values.

In contrast, J_{MCC} can be used for both comparison and to evaluate the model's quality relative to the actual output. According to Coelho and Coelho (2004), for several practical applications, a J_{MCC} value between 0.8 and 1 is considered adequate. J_{MCC} is a measure of the correlation coefficient between the actual output and the model's output. A higher J_{MCC} value indicates a better correlation between the model and the actual system, demonstrating its suitability for practical use.

Table 2. Values of the indexes of the estimated models.

	Parameters	Pitch	Roll	Altitude	Yaw
ARX	J_{SEQ}	0.5550	0.3522	48.8069	3172.1
	J_{MCC}	0.9346	0.9592	0.9114	0.7107
ARMAX	J_{SEQ}	0.4922	0.2514	0.6379	184.0303
	J_{MCC}	0.9420	0.9708	0.9988	0.9832

5. CONCLUSION

The results obtained from the study were found to be satisfactory, supporting the hypothesis that moving average parcel identification leads to significant improvements in the resulting models. The decoupled identification strategy used in the study was successful in producing appropriate models for the design and simulation of controllers for the Parrot AR Drone 2.0. These results provide a reference for future researchers aiming to apply linear stochastic control algorithms to the drone.

The analysis of J_{MCC} indices showed that the ARX models may be sufficient for practical applications, but the ARMAX models produced significant improvements, leading to better stabilization solutions for the AR Drone 2.0. The use of ELS algorithm resulted in improved identification quality of linear systems compared to the classic OLS algorithm.

Moreover, all models were identified with the same sampling period, simplifying the analysis of the drone dynamics. Table 2 presents the indices of the main results obtained from the study, which support the feasibility of the proposed methodology.

REFERENCES

Aguirre, L.A. (2007). *Introdução à identificação de sistemas: Técnicas lineares e não-lineares aplicadas a sistemas reais*. Editora UFMG, Florianópolis, 3rd edition.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE transactions on automatic control*.

Coelho, A.A.R. and Coelho, L.D.S. (2004). *Identificação de Sistemas Dinâmicos Lineares*. Editora da UFSC, Florianópolis, 2nd edition.

Daud, S.M.S.M., Yusof, M.Y.P.M., C.Heo, C., Khoo, L.S., Singh, M.K.C., Mahmood, M.S., and Nawawi, H. (2022). Applications of drone in disaster management: A scoping review. *Science and Justice*, 62, 30–42.

Hernandez, A.E.A. (2013). Identification and path following control of an ar. drone quadrotor. *17th international conference on system theory, control and computing (ICSTCC). IEEE*.

Ljung, L. (1987). *System Identification: Theory for the User*. Prentice-Hall, New Jersey, 2nd edition.

Lopes, L.L.E.A. (2013). Modelagem e validação de um quadrimotor ar drone parrot. *International Micro Air Vehicle Conference and Flight Competition*.

Mogili, U.R. and Deepak, B.B.V.L. (2018). Review on application of drone systems in precision agriculture. *Procedia computer science*, 133, 502–509.

Parrot (2014). Ar drone user guide. *https://www.parrot.com/en/support/documentation/ardrone*. Accessed: 09/03/2023.

Patwardhan, R.S. and Shah, S.L. (2002). Issues in performance diagnostics of model-based controllers. *Journal of Process Control*, 12, 413–427.

Sanabria (2013). Ar drone simulink development-kit v1.1. *https://www.mathworks.com/matlabcentral/fileexchange/43719-ar-drone-simulink-development-kit-v1-1*. Accessed: 09/03/2023.

Springer, P. (1954). Military robots and drones: a reference handbook. In *Military robots and drones: a reference handbook*, volume 2. Comtemporary World Issues, Santa Barbara, California, 1st edition.

Stevens, B.L., Lewis, F.L., and Johnson, E.N. (2016). *Aircraft control and simulation: dynamics, controls design, and autonomous systems*. John Wiley and Sons, New Jersey, 3rd edition.

Åström, K.J. and Wittenmark, W.B. (2013). *Computer-controlled systems: theory and design*. Courier Corporation, New York, 3rd edition.