25th ABCM International Congress of Mechanical Engineering
October 20-25, 2019, Uberlândia, MG, Brazil

COB-2019-1952
PARAMETER IDENTIFICATION AND CONTROL DESIGN OF A QUADROTOR UAV F450

Leonardo Avelino de Lima Jacinto
Tiago Avelino Ribeiro da Silva
Beatriz Carolina Borges Pinho
Victor Hugo Santos Lima
André Murilo
Renato Vilela Lopes
Evandro Leonardo Silva Teixeira
Universidade de Brasilia
tiago.avelino1997@gmail.com

Abstract. *The present work addresses the implementation embedded controller in an Unmanned Aerial Vehicle (UAV) quadrotor type. A well-known controller in the control systems field was chosen: the Linear Quadratic Regulator (LQR). The goal is to implement that controller in a famous flight control platform known as PX4 Autopilot, ensuring that, even with all the existing applications, the controller works in a deterministic manner complying with the time constraints. For the implementation of those controllers, the physical properties of the quadrotor F450 were found theoretically and experimentally and used on the controllers.*

Keywords: *quadrotor. embedded systems.* PX4. *Parameters. F450.*

1. INTRODUCTION

The development of Unmanned Aerial Vehicles (UAVs) is growing in the last few years due to its many applications in many fields like the military, agriculture, e-commerce, and many others (Hassanalian and Abdelkefi, 2017). The quadrotor is one of the most common types of UAV and its main characteristics are the vertical takeoff/landing and the ability to hover which results in a great capacity for tasks of supervision, inspection and work in environments where space is limited (Bills *et al.*, 2011).

However, the quadrotors are multivariable, underactuated nonlinear mechatronic system, which makes the problem of stabilization and position control challenging. Most of the control algorithms rely on the quality of the mathematical models that describe the dynamics of the system which makes modeling an important question.

Because of the complex dynamics, a considerable amount of papers have been published regarding modeling and developing control strategies for quadrotors: Identification of parameters and Backstepping controller (Araar *et al.*, 2017), Linear Quadratic Regulator (LQR) (Reyes-Valeria *et al.*, 2013a), LQR using quaternions (Reyes-Valeria *et al.*, 2013b), LQR state feedback (Panomrattanarug *et al.*, 2013).

The equations that describe the quadrotor dynamics are well-established in literature and a common approach is utilizing the Euler-Lagrange formulation as can be seen in (Santana and Braga, 2008) and (Araar *et al.*, 2017) and the linearized model in (Lopes *et al.*, 2011). For a good controller performance, it is important to determine all the variables and coefficients necessary, so that the solution is closer to the actual system, this can be achieved by numerous forms that include numerical simulations and experiments. The process of modeling and identification of parameters are critical steps for a good performance of the controller.

There are several quadrotor control boards, among those the *Pixhawk* (PX4) is widely used in academic research, because is an open-hardware and open-software autopilot (Meier *et al.*, 2015). The PX4 is a well know board used in a lot of applications such as parameter identification (Yang and Liu, 2013), control design (Salunkhe *et al.*, 2016) and trajectory tracking control (Lin *et al.*, 2014) and hardware design

One of the biggest problems in quadrotor control design is to ensure the safeness of the vehicle and the people around it. To avoid accidents and the destruction of the vehicle various methods are used in the bibliography, one of these is the Software In the Loop (SIL) architecture, this kind of simulation is used to model and design a controller (Nagaty *et al.*, 2013) and create a simulator environment (Furrer *et al.*, 2016). Another simulation that can be performed is simulating all real sensors in a software, without needing to fly with the vehicle (Röck, 2011), this type of simulation is known as the Hardware In the Loop (HIL). A certain amount of papers are published using the HIL platform to develop embedded

control strategies (De Farias *et al.*, 2019), and in UAV quadrotors (Wu *et al.*, 2018) and (Aljehani and Inoue, 2019), among many others.

A different way of ensuring the safety of the embedded control in a quadrotor is through a test with the vehicle in flight but attached to a platform, the test bench. Some papers work with this test, such as (Khan and Kadri, 2014), test bench with control comparison between PID and LQR controllers (Bouabdallah *et al.*, 2004), six degree of freedom test bench for quadrotor (Yu and Ding, 2012) and control stabilization of aircraft quadrotor with quaternion-based feedback control (Tayebi and McGilvray, 2006).

To finalize the validation step a flight test is performed in many articles (Goel *et al.*, 2009), UAV with tilling propellers (Ryll *et al.*, 2013) and fixed-wing UAVs (Gunarathna and Munasinghe, 2018). Although many articles describe one step at a time, none of those describe the complete methodology for embedded a controller in a UAV.

In this paper, the intention is to describe a complete and robust methodology for the implementation of an LQR attitude controller for a quadrotor helicopter DJI F450. In particular, this paper describes a strategy for the identification of the parameters needed for the model of the quadrotor and a full procedure for the embed controller, which involves a SIL and HIL simulations, a test bench and a flight test.

This paper is organized as follows: in the first section is described the mathematical model of the quadrotor. The following section shows the methodology used for the parameters identification and the process of embedding the LQR controller. In section 4 is presented the discussion of the results obtained in the modeling part, SIL, HIL simulations and the flight test. The last section presents the conclusion of the work.

2. SYSTEM MODEL E CONTROL DESIGN

The quadrotor is a dynamic system and its motion is regulated by four rotors that act like actuators generating thrust. The flight of the vehicle relies on having a pair of rotors spinning clockwise and the other pair spinning counterclockwise. In Fig.1 it is possible to observe the forces (f_1, f_2, f_3, f_4), torques ($\tau_{M_1}, \tau_{M_2}, \tau_{M_3}, \tau_{M_4}$) and angular velocities ($\omega_1, \omega_2, \omega_3, \omega_4$) that act in the system.

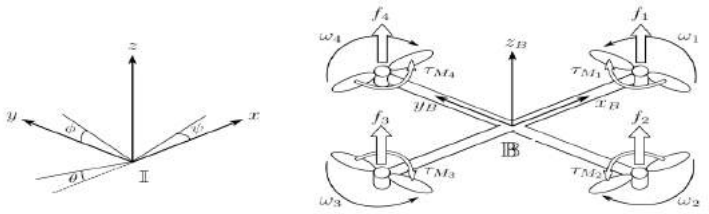


Figure 1: Diagram of quadrotor dynamics with inercial and body reference frames. Adapted from: (Luukkonen, 2011).

The equations that describe the system model witch are obtained from the Euler-Lagrange formulation are defined by (Santana and Braga, 2008) and represented in Eq.1, where Φ represents the roll angle, Θ the pitch angle and Ψ the and yaw angle. τ_x, τ_y and τ_z are the torques and I_{xx}, I_{yy} and I_{zz} are the inertias on axis x,y and z, respectively.

$$\ddot{\phi} = \frac{I_{yy} - I_{zz}}{I_{xx}}\dot{\psi}\dot{\theta} + \frac{\tau_x}{I_{xx}}, \qquad \ddot{\theta} = \frac{I_{zz} - I_{xx}}{I_{yy}}\dot{\psi}\dot{\theta} + \frac{\tau_y}{I_{yy}}, \qquad \ddot{\psi} = \frac{I_{xx} - I_{yy}}{I_{zz}}\dot{\theta}\dot{\phi} + \frac{\tau_z}{I_{zz}}. \tag{1}$$

The torques τ_x, τ_y and τ_z from Eq.1 are results of the thrust and angular velocities as can be seen in the following equations (Araar *et al.*, 2017) :

$$T = b\Omega^2, \qquad M = d\Omega^2, \tag{2}$$

where b is the lift factor, d is the drag factor and ω is the angular velocity. This model can be approximated to a linear one through a Taylor expansion and rewritten in the state of spaces representation as can be seen in Eq.3 (Lopes *et al.*, 2011). Here ξ is the attitude state vector and is given by $\xi = [\phi\ \dot{\phi}\ \theta\ \dot{\theta}\ \psi\ \dot{\psi}]$ and u is the control input $u = [\Omega_1\ \Omega_2\ \Omega_3\ \Omega_4]^T$.

$$\dot{\xi} = A_c\xi + B_c u. \tag{3}$$

In order to implement in an autopilot, which is a digital system, this model is discretized as can be seen in Eq. 4(Lopes *et al.*, 2011). The sampling period in this model is 50ms.

$$\xi(k+1) = A\xi(k) + Bu(k), \qquad y(k) = C\xi(k). \tag{4}$$

After the system modeling process, an optimal control method that results in optimum feedback gain was chosen

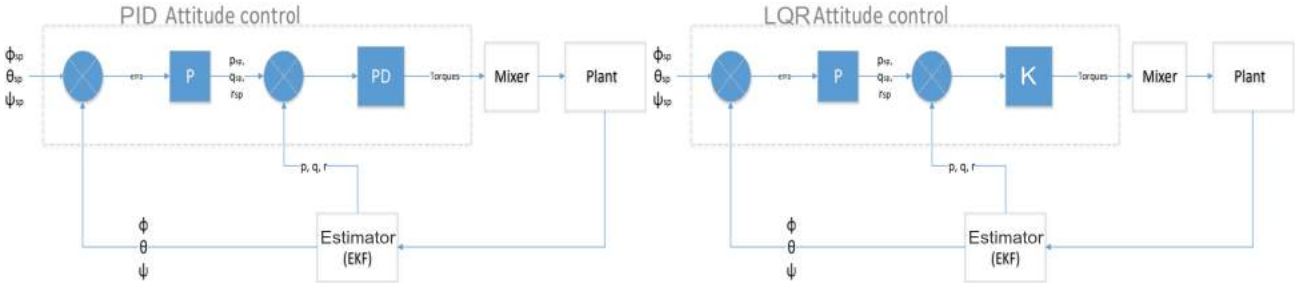
because of the linear quadratic regulator, as can be seen in Fig. 2b. With a linear time-invariant system model and a quadratic cost function, the optimization problem has an analytical solution trough the Ricatti equation, yielding a constant gain feedback formulation. The solution is calculated as in (Bouabdallah *et al.*, 2004) and can be seen in Eq.6.

$$J(u) = \int_0^{\inf} ((\xi - \xi^d)^T Q(\xi - \xi^d) + u^T R u\ ) dx. \tag{5}$$

Here, $\xi^d = [\phi_d\ \dot{\phi}_d\ \theta_d\ \dot{\theta}_d\ \psi_d\ \dot{\psi}_d]$ is the desired state. The matrix $[Q]$ is the weighting matrix of the states ξ and the control matrix $[R]$ is positive definite *weighting matrices*, these are set according to control objectives. A higher pondering on $[Q]$ will impose higher costs on orientation errors, may result in high control commands on u. On the other hand, by increasing the components in the $[R]$ matrix, the controller will prioritize minimizing the command efforts. The LQR strategy is particularly useful for MIMO systems since it's an optimal way to set controller gains.

In order to change the embedded control, the control design step was performed applying the controller in the PX4 control architecture. Initially, the PX4 control architecture provides the attitude Proportional Integral Derivative (PID) controller as can be seen in Fig. 2a. The attitude controller receives the attitude angles (roll, pitch and yaw) trough the position controller. Since the position controller only provides set points for attitude angles, the angular velocities had to be calculated through a proportional controller, getting the full attitude state vector.

Following the proportional controller, the angles are estimated by an Extended Kalman Filter (EKF) (PX4, 2019), allowing feedback loop control and calculation of a new attitude angle setpoint, as can be seen in Fig. 2b.



(a) PID Attitude Control. (b) LQR attitude control.

Figure 2: In Fig.2a shows the control architecture of PX4. In Fig.**??** shows the chosen LQR controller to embedded in the PX4 architecture.

3. METHODOLOGY

The methodology for this paper involved five stages: parameters identification, SIL simulations, HIL simulations, a test bench and a flight test. These phases are graphically represented in the diagram of Fig. 3. For the parameters identification, the first step involved obtaining the inertias of the quadrotor through a CAD model, then two experiments were carried out to identify the drag and lift factors. After an established model, SIL and HIL simulations were performed towards obtaining an optimum gain. Then, a test bench was conducted to ensure the safeness of the quadrotor. For the final step adjustment, a flight test was executed.

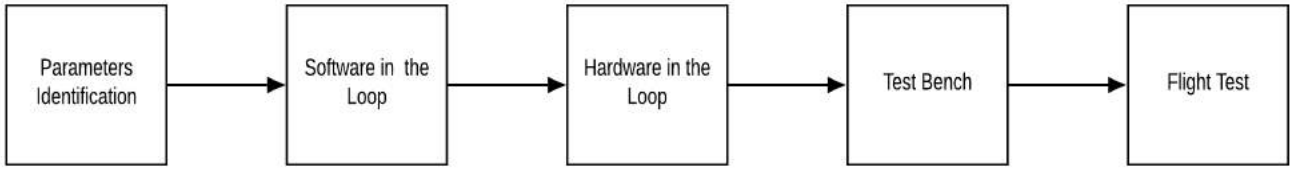


Figure 3: Diagram of the stages of the controller development.

3.1 Parameter Identification

It is possible to observe from Eq.1 that the system dynamics relies on the inertias and torques of the quadrotor. The estimation of the moments of inertia in x, y and z (I_{xx}, I_{yy} and I_{zz}) was achieved through a CAD model, with the assumption that the quadrotor is symmetric. This CAD model was made in *SolidWorks*® and can be seen in Fig.4.

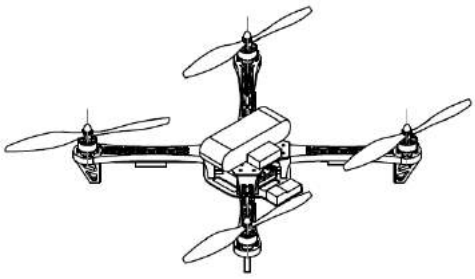
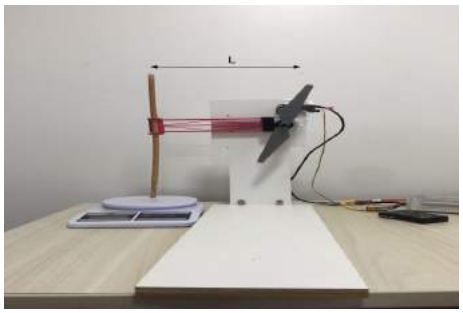


Figure 4: F450 CAD model employed in the acquisition of the inertias.

In order to determinate the lift and drag coefficients, two experiments were performed based on the work of (Araar *et al.*, 2017). To determine the thrust relation with the angular velocity the experiment consists of a system in which one of the rotors was attached to weights heavier than its thrust power and placed over a scale, as can be seen in Fig.5b.Then, the velocity of the rotor was modified and for each value, the scale indicated the mass dislocated due to the thrust. Simultaneously, the angular velocity was measured with a digital tachometer Ininipa MDT-2238A that was positioned above the structure as observed in Fig. 5a.

(a) Experiment for the thrust factor acquisition.



(b) Experiment for the drag factor acquisition.

Figure 5: In Fig.5a a rotor is attached to a weight over a scale for acquiring the thrust. In the upper part of the image, a tachometer was being used for measuring the angular velocity. In Fig.5b a rotor is attached perpendicular to the scale that is being used for measuring the force produced by the angular velocity. Where "L" is the lever arm of the force.

For the d factor, the rotor was fixed perpendicular to the ground in a structure that is shown in Fig.5b. Part of this structure was designed in *SolidWorks*® and 3D printed, which was important to ensure that the structure was not generating any friction that could disturb the revolving of the rotor. With this setup, the mass dislocated by the angular velocity of the rotor was measured with a scale for different velocities. This mass multiplied by the gravity acceleration and the lever arm, which is represented in Fig.5a by "L", resulted in the linear moment. The tachometer measured once more the angular velocity.The drag factor, contrariwise as the lift factor is calculated punctually by $d = \frac{M}{\Omega^2}$.

3.2 PX4 Architecture

To embedded the LQR controller in the PX4 architecture is necessary to understand the PX4 flight stack. The flight stack has some modules sensors, position and attitude estimators, position and attitude controllers, navigator and radio controller, mixer and actuator. Those modules provides an overview of the system operation as can be seen in Fig 6, and each module is briefly described bellow.

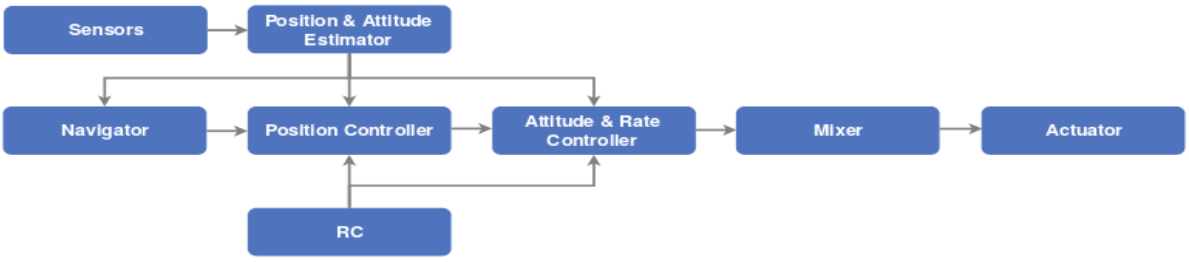


Figure 6: Block diagram of flight control embedded on *PX4*. Available from: (autopilot, 2019).

1. Sensors and Position & attitude Estimator: The PX4 has basically 3 sensors: the gyroscope, magnetometer and the accelerometer, which are used to estimate all quadrotor states.

2. Position Controller and Attitude & Rate Controller: To control the quadrotor, the firmware is executed through two controller blocks: position and attitude. The position block sends the setpoints of the angles roll, pitch and yaw to the attitude block that sends torque commands to the engines, through a mixer. This command guarantees different torques for each engine, making the drone follow the received trajectory.

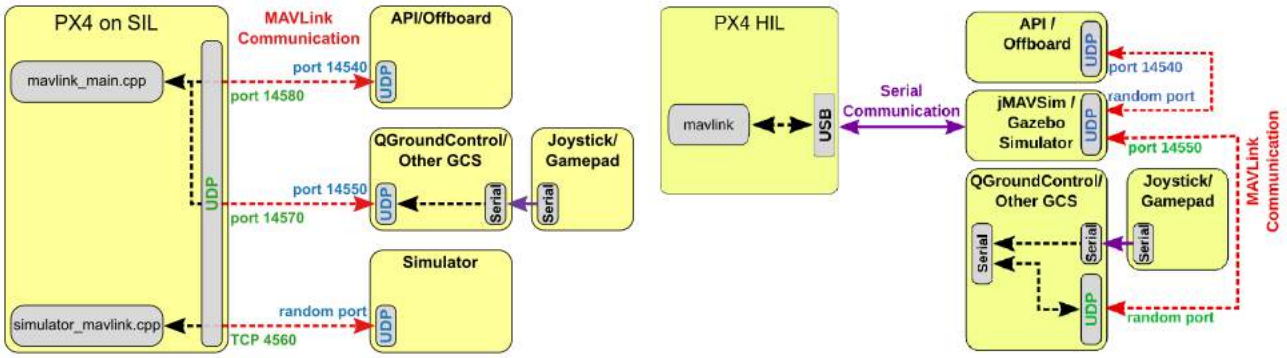
3. Navigator and RC: The position controller requires an initial setpoint, the setpoint is passed by a navigator with a radio controller (RC).

4. Mixer and Actuator: The mixer block receives the information in torque and translates this information into a pulse width modulation (PWM) input for the motors. The controller used the Micro Object Request Broker (uORB) communication for transmitting between hardware and software, with functions overwriting the sensor data periodically and then writing new values in the actuators.

3.3 Software In the Loop

After the controllers are simulated in *MATLAB* to obtain the optimum gain, is performed the software in the loop architecture. The SIL allows the vehicle to pass to several tests, including the application of torque commands, quadrotor inertia variation, among others. An advantage of working with the SIL is the debugging tools that allow correction of larger errors in a shorter time due to the higher processing power of computers when compared to the *Pixhawk* system. Another advantage is to allow a simulation of the firmware with the attitude controller without the necessity for a flight test.

Additionally, SIL simulation is the secure implementation because of the several tests that can be performed in a software. The software *Gazebo* was used to create a simulation environment for the SIL approach. The *Gazebo* is an open-source software that allows testing the implemented algorithm in 3D scenarios, making several tests cited above. The communication interface used to connect with the trajectory software *QGroundControl* is shown in Fig. 7a, where the setpoints are sent to the embedded firmware. The protocol used for data transfer is the MAVlink also shown in Fig. 7a.



(a) SIL architecture. (b) HIL architecture.

Figure 7: In Fig 7a is presented the PX4 SIL architecture for gazebo and jMAVSIM simulators, available from: (PX4, 2019b). In Fig 7b is presented the HIL architecture for gazebo and jMAVSIM simulators, available from: (PX4, 2019a).

The *PX4* firmware provides different quadrotors models for a 3D environment, that simulate all the actuators and sensors. Among those, the Iris model was chosen because of the proximity to the F450. The inertias, the drag and lift coefficients of the model were replaced by those obtained in this work for a better approximation.

3.4 Hardware In the Loop

When the gains were finally adjusted in SIL, then a Hardware In the Loop (HIL) simulation was proceeded, receiving data from a simulated system through the *Pixhawk 1* board responsible for embedded the controller. The HIL simulation enables a real time response of the controller in the embedded system, allowing to check the response time and the commands generated by the controller, thus validating the controller performance while simulating the sensors and the actuators, without need to perform a flight test. One feature that should be emphasized is that HIL simulation allows to verify the control modules, diagnosing failures and correcting them before damage the vehicle (De Farias *et al.*, 2019).

The *jMAVSIM* simulator allowed the verification of the hardware response so that final adjusts in the LQR gain were obtained. At the moment of this work, the *Gazebo* was not able to execute HIL simulation, but the new versions of the simulator already allow HIL. The *jMAVSIM* interface communication with the *Pixhawk* 1 board can be seen in Fig. 7b. The biggest difference between the SIL interface and HIL is in direct communication with the board in HIL while in SIL approach simulates the board operation.

3.5 Test Bench

Subsequently, the test bench was executed to ensure the safeness of the quadrotor and can be seen in Fig. 8. The bench was built by (Magalhães *et al.*, 2014) to a helicopter and adapted to fix the quadrotor to the bench, which influences the dynamics of the vehicle. Nevertheless, the platform is useful to evaluate single commands and the attitude angles of the quadrotor. The test was performed by applying torque commands with an RC and creating some disturbances in the platform to verify the quadrotor response, the data acquisition can be seen in Fig 8.

Figure 8: Test bench of LQR controller. Its possible to see the axis attached to the platform and data acquisition.

3.6 Flight Test

The test was performed by taking the quadrotor to a test field with the controller embedded in the board. The quadrotor went through variations in the command of the radio control, performing some simple maneuvers. Subsequently, an attitude angle analysis was performed to make slight adjustments to the LQR controller gains.

Even with all the previous steps of validation the embedded controller it is necessary to emphasize that in the first flight test some restrictions should be followed, such as avoiding flying near objects such as trees and benches and prevent days when the wind is more intense and avoiding places with people around as can be seen in (Avelino, 2017).

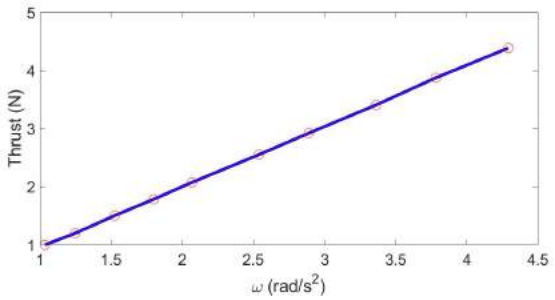
4. RESULTS

4.1 Parameter Identification

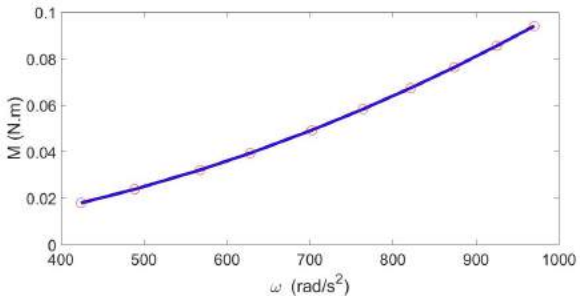
The drag coefficient (d), lift coefficient (b) and mass of the drone were acquired experimentally. The angular velocity of the rotor was altered with the modification of the PWM duty cycle signal.

For the lift factor, were performed three measurements for each fixed duty cycle value. The duty cycle ranged from $1230us$ to $1500us$, which were being acquired at each variation of $30us$. It is possible to observe in Fig.9a the graph of the thrust in respect to the angular velocity, in which the lift coefficient is the angular coefficient of the curve.

In order to obtain the drag factor, once more were performed three measurements for each fixed duty cycle value ranging from $1250us$ to $1750us$ with the acquisition of $50us$. For this coefficient lower duty cycle values resulted in inconsistent data, so the values below $1350us$ were discarded. The drag factor differs from then lift factor as it is calculated punctually. The experimental results for the yaw torque concerning to the angular velocities are shown in Fig. 9b, in which d is the mean value of the ratio between the yaw moment and the squared angular velocity.



(a) Graph of total thrust versus rotor angular velocity.



(b) Graph of total yaw moment versus rotor angular velocity.

Figure 9: In Fig.9a the angular coefficient of this curve is equal to the lift coefficient.From Fig.9b the drag coefficient is the average from the ratio between the yaw moment and and the angular velocity.

The inertias from the quadrotor were obtained through a CAD model made in *SolidWorks*®. The value of all parameters are shown in the table below:

Mass with the battery	1.340 kg	I_{xx}	0.01079714 (kgm^2)
b	1.04457 *10^{-5} Ns^2/rad^2	I_{yy}	0.01183289(kgm^2
d	1.602491 *10^{-7} Nms^2/rad^2	I_{zz}	0.02157329 (kgm^2)

4.2 Software In the Loop

Initially was developed the LQR controller for SIL, setpoints was passed by *QGroundControl* to *Gazebo* simulator, what has already been shown in Fig. 7a. In some moments, a few disturbances were applied in the quadrotor via software, to ascertain the quadrotor response. Data acquisition is done at $250Hz$ at all steps SIL, HIL, test bench, and flight test because it is the time it takes to PX4 EKF estimators to transmit data. This simulation obtained good results as can be seen in Fig.10, it is possible to see the simulated attitude angles of the quadrotor.

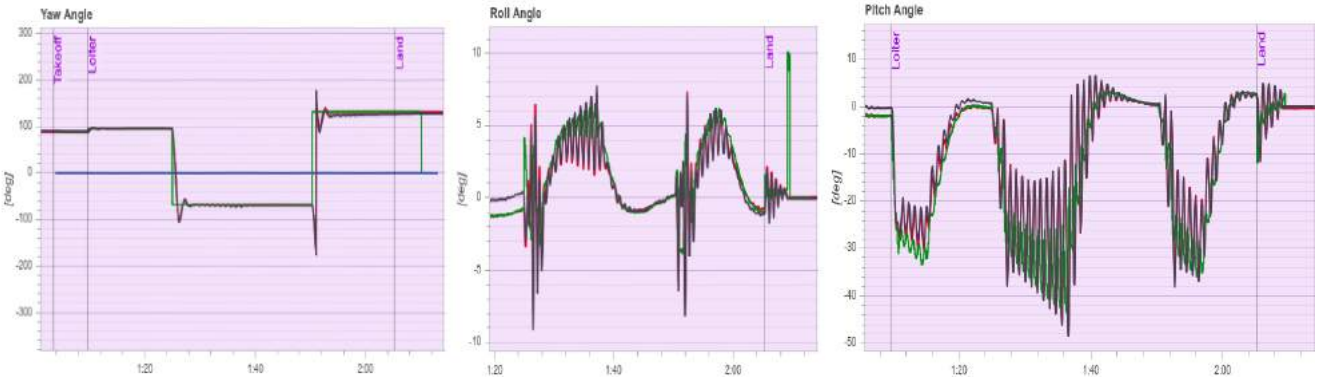


Figure 10: LQR Controller response for yaw, roll and pitch angles respectively, in SIL. In red estimated angles, in green the setpoint and in black the ground truth.

To assess whether the projected LQR gains were well adjusted, a comparison was made with the original PID embedded into the PX4 firmware. The results are shown in Fig. 11 show that LQR performed better, with faster response time than the original PID.

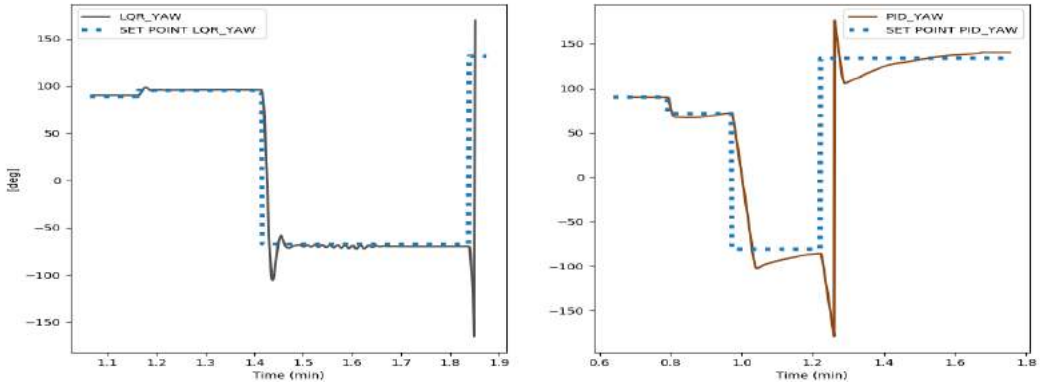


Figure 11: Comparison between the yaw angle of *PX4* PID controller and designed LQR controller in SIL.

4.3 Hardware In the Loop

In the HIL step, commands are not passed through a simulator, but through an RC device. This device sends the position setpoints and the vehicle is simulated in an environment with varying wind speeds to validate the attitude controller performance. The gains used in HIL showing good results in attitude angles, the quadrotor follows the setpoint reference, as can be seen in Fig.12.

Figure 12: LQR Controller response for yaw, roll and pitch angles respectively, in HIL. In red estimated angles and in green the setpoint.

A new comparison was made between the original PID embedded in PX4 firmware and the designed LQR controller for HIL. This comparison was made with different setpoints, but LQR shows a better response for overshooting and presented a more stabilized result, this comparison can be seen in Fig.13.

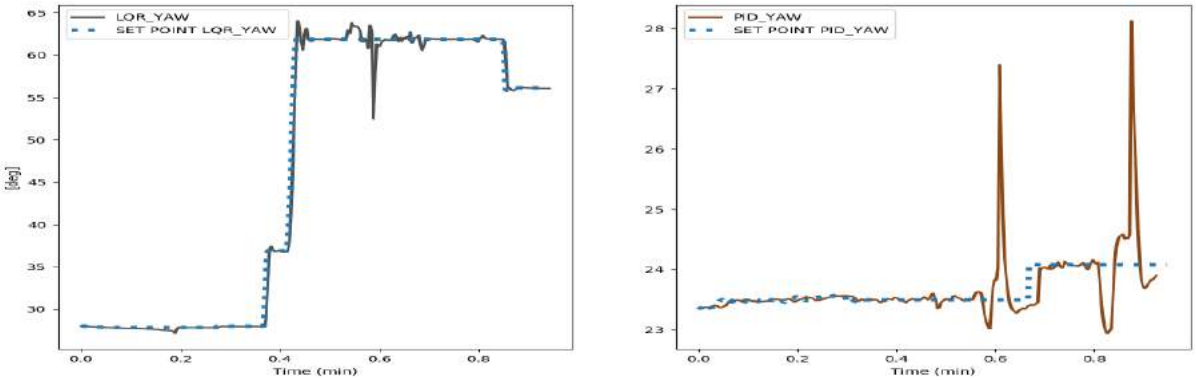


Figure 13: Comparison between the yaw angle of *PX4* PID controller and designed LQR controller in HIL.

4.4 Flight Test

For the flight test, the quadrotor was taken to an environment without people and away from objects as presented in the methodology, some quadrotor movements were performed to validate the controller. Through this procedure, the gain K was refined obtaining the last value for a safer flight test. The gain K obtained was K = [1.2746, 2.3401, 0, 0, 0, 0 ; 0, 0, 1.5693 2.9010, 0, 0; 0, 0, 0, 0, 0.5941, 1.8834], these values are generated by the prioritization of the angles φ, Θ and Ψ on the diagonal of the matrix [Q].

The setpoint and the LQR response for attitude angles can be seen in Fig.14. It is possible to see that in-flight tests the results are noisier than in other steps. Nevertheless, the response follows the setpoint as good as the others.

Figure 14: LQR Controller response for roll angle in flight test.

Another comparison was made between the *PX4* firmware and the designed LQR controller for the flight test. This comparison was made with different setpoints, but LQR showed a better response for overshoot, velocity, and stabilization for yaw angle as can be seen in Fig.15.

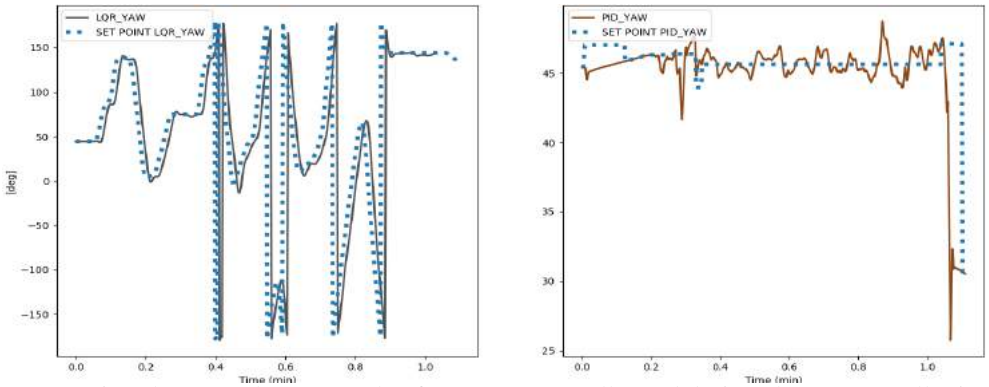


Figure 15: Comparison between the yaw angle of *PX4* PID controller and designed LQR controller in flight test.

5. CONCLUSION

With the aim to develop a complete methodology for the design and implementation of a controller architecture it was essential to obtain a system model with accurate parameters, which made the identification crucial. The conducted experiments and a CAD model of the F450 allowed an estimation of the drag and lift factors, mass, and inertias of the system. For the LQR controller design, the gains calculated by the *MATLAB* software allowed a good starting point for the calculation of the optimal gain. With the optimal gain, small empirical changes were necessary to guarantee the stability and speed of the response. Thus was possible to guarantee a real-time system of the embedded firmware with attitude controller LQR. The comparison between the *PX4* PID controller showed that the designed controller presented a better response for overshoot, velocity, and stabilization. It is important to comment that the difference between the simulators in SIL and HIL steps influences the gain due to the difference in the models used in each simulator. Although, those differences can be mitigated by an accurate model and the robustness of the LQR controller.

6. ACKNOWLEDGEMENTS

The authors thank the National Council for Scientific and Technological Development (CNPq), the Federal District Research Support Foundation (FAP-DF), University of Brasília - Gama College (UnB-FGA) and CAPES for their partial support to research activities.

7. REFERENCES

Aljehani, M. and Inoue, M., 2019. "Performance evaluation of multi-uav system in post-disaster application: Validated by hitl simulator". *IEEE Access*, Vol. 7, pp. 64386–64400. doi:10.1109/ACCESS.2019.2917070.

Araar, O., Minouni, M.Z., Fella, K. and Osmani, H., 2017. "Identification control of a multirotor uav in the presence of actuator asymmetry". In *25th Mediterranean Conference on Control and Automation(MED)*. Valleta, Malta.

autopilot, P., 2019. "Px4 architectural overview". PX4 autopilot. 25 Jul. 2019 <https://dev.PX4.io/en/concept/architecture.html>.

Avelino, L., 2017. "Lqr controller embedded in pixhawk hardware". URL `https://www.youtube.com/watch?v=g66_KfxAI_Q&feature=youtu.be`.

Bills, C., Chen, J. and Saxena, A., 2011. "Autonomous mav flight in indoor environments using single image perspective cues". In *2011 IEEE International Conference on Robotics and Automation*. pp. 5776–5783.

Bouabdallah, S., Noth, A. and Siegwart, R., 2004. "Pid vs lq control techniques applied to an indoor micro quadrotor". In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*. Vol. 3, pp. 2451–2456 vol.3. doi:10.1109/IROS.2004.1389776.

De Farias, A.B.C., Rodrigues, R.S., Murilo, A., Lopes, R.V. and Avila, S., 2019. "Low-cost hardware-in-the-loop platform for embedded control strategies simulation". *IEEE Access*, Vol. 7, pp. 111499–111512. doi:10.1109/ACCESS.2019.2934420.

Furrer, F., Burri, M., Achtelik, M. and Siegwart, R., 2016. *Rotors – A Modular Gazebo MAV Simulator Framework*, Vol. 625, pp. 595–625. ISBN 978-3-319-26054-9. doi:10.1007/978-3-319-26054-9_23.

Goel, R., Shah, S., Gupta, N. and Ananthkrishnan, N., 2009. "Modeling, simulation and flight testing of an autonomous quadrotor".

Gunarathna, J.K. and Munasinghe, R., 2018. "Development of a quad-rotor fixed-wing hybrid unmanned aerial vehicle". In *2018 Moratuwa Engineering Research Conference (MERCon)*. pp. 72–77. doi:10.1109/MERCon.2018.8421941.

Hassanalian, M. and Abdelkefi, A., 2017. "Classifications, applications, and design challenges of drones: A review". *Progress in Aerospace Sciences*, Vol. 91, pp. 99–131.

Khan, H.S. and Kadri, M.B., 2014. "Position control of quadrotor by embedded pid control with hardware in loop simulation". In *17th IEEE International Multi Topic Conference 2014*. pp. 395–400. doi:10.1109/INMIC.2014.7097372.

Lin, Q., Cai, Z., Yang, J., Sang, Y. and Wang, Y., 2014. "Trajectory tracking control for hovering and acceleration maneuver of quad tilt rotor uav". In *Proceedings of the 33rd Chinese Control Conference*. pp. 2052–2057. doi:10.1109/ChiCC.2014.6896946.

Lopes, R.V., Borges, G.A., Ishihara, J.Y. and Santana, P.H.R.Q.A., 2011. "Model predictive control applied to tracking and attitude stabilization of a vtol quadrotor aircraft".

Lopes, R.V., Santana, P., Borges, G. and Ishihara, J., 2011. "Model predictive control applied to tracking and attitude stabilization of a vtol quadrotor aircraft". In *21st International Congress of Mechanical Engineering*. pp. 176–185.

Luukkonen, T., 2011. "Modelling and control of quadcopter". Espoo, Finland.

Magalhães, J., Zoé, R. and Lopes, R.V., 2014. "Desenvolvimento de uma plataforma de teste para controle de atitude de helicópteros de pequena escala". Brasilia, Brasil.

Meier, L., Honegger, D. and Pollefeys, M., 2015. "Px4: A node-based multithreaded open source robotics framework for deeply embedded platforms". In *2015 IEEE International Conference on Robotics and Automation (ICRA)*. pp. 6235–6240.

Nagaty, A., Saeedi, S., Saeedi, C., Seto, M. and H.Li, 2013. "Control and navigation framework for quadrotor helicopters". *Journal of Intelligent Robotic Systems*, Vol. 70. doi:10.1007/s10846-012-9789-z.

Panomrattanarug, B., Higuchi, K. and Mora-Camino, F., 2013. "Attitude control of a quadrotor aircraft using lqr state feedback controller with full order state observer". In *The SICE Annual Conference 2013*. Nagoya, Japan.

PX4, A., 2019. "Ecl/ ekf overview tuning". ECL/ EKF Overview Tuning · PX4 v1.9.0 User Guide. 24 Jul. 2019 <https://docs.px4.io/v1.9.0/en/advanced_config/tuning_the_ecl_ekf.html>.

PX4, A., 2019a. "Hil simulation". HITL Simulation · PX4 v1.9.0 Developer Guide. 25 Jul. 2019 <https://dev.px4.io/v1.9.0/en/simulation/hitl.html>.

PX4, A., 2019b. "Simulation". Simulation · PX4 v1.9.0 Developer Guide. 24 Jul. 2019 <https://dev.px4.io/v1.9.0/en/simulation/>.

Reyes-Valeria, E., Enriquez-Caldera, R., Camacho-Lara, S. and Guichard, J., 2013a. "Identification control of a multirotor uav in the presence of actuator asymmetry". In *CONIELECOMP 2013, 23rd International Conference on Electronics, Communications and Computing (CONIELECOMP)*. Cholula, Mexico.

Reyes-Valeria, E., Enriquez-Caldera, R., Camacho-Lara, S. and Guichard, J., 2013b. "Lqr control for a quadrotor using unit quaternions: Modeling and simulation". In *CONIELECOMP 2013, 23rd International Conference on Electronics, Communications and Computing*. pp. 172–178. doi:10.1109/CONIELECOMP.2013.6525781.

Ryll, M., Bülthoff, H.H. and Giordano, P.R., 2013. "First flight tests for a quadrotor uav with tilting propellers". In *2013 IEEE International Conference on Robotics and Automation*. pp. 295–302. doi:10.1109/ICRA.2013.6630591.

Röck, S., 2011. "Hardware in the loop simulation of production systems dynamics". *Production Engineering*, Vol. 5, pp. 329–337. doi:10.1007/s11740-011-0302-5.

Salunkhe, D.H., Sharma, S., Topno, S.A., Darapaneni, C., Kankane, A. and Shah, S.V., 2016. "Design, trajectory generation and control of quadrotor research platform". In *2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA)*. pp. 1–7. doi:10.1109/RAHA.2016.7931876.

Santana, P.H.R.Q.A. and Braga, M., 2008. "Concepção de um veículo aéreo não-tripulado do tipo quadrirrotor".

Tayebi, A. and McGilvray, S., 2006. "Attitude stabilization of a vtol quadrotor aircraft". *IEEE Transactions on Control Systems Technology*, Vol. 14, No. 3, pp. 562–571. doi:10.1109/TCST.2006.872519.

Wu, Y., Hu, K. and Sun, X., 2018. "Modeling and control design for quadrotors: A controlled hamiltonian systems approach". *IEEE Transactions on Vehicular Technology*, Vol. 67, No. 12, pp. 11365–11376. doi:10.1109/TVT.2018.2877440.

Yang, L. and Liu, J., 2013. "Parameter identification for a quadrotor helicopter using pso". In *52nd IEEE Conference on Decision and Control*. pp. 5828–5833. doi:10.1109/CDC.2013.6760808.

Yu, Y. and Ding, X., 2012. "A quadrotor test bench for six degree of freedom flight". *Journal of Intelligent Robotic Systems*, Vol. 68. doi:10.1007/s10846-012-9680-y.