MODELING, CONTROL AND HARDWARE-IN-THE-LOOP SIMULATION OF A GENERIC TILTING MULTI-ROTOR

F. Machini*, I. O. Tarifa, R. M. F. Neto, L. Sanches
***Universidade Federal de Uberlândia, Uberlândia-MG, Brazil**

Abstract

In general, standard multi-copters are classified as an underactuated system since their number of control inputs are insufficient to allow the control of position and orientation independently. In this context, this paper deals with the dynamical modeling of a tilted rotor multi-copter aerial vehicle and the project of a trajectory tracking controller using modern control techniques. The dynamic model is developed using Newton Euler Laws and it is assumed that each rotor is capable of two different movements (tilt laterally and longitudinally) introducing more control inputs to the system. Then, the equations of motion are linearized around the trimmed operating conditions (based on mission applications). Finally, linear modern MIMO (Multi-Input Multi-Output) control techniques are applied in order to allow the aircraft to follow a pre-defined trajectory. The control law is validated by a hardware-in-the-loop implementation using a real microcontroller integrated with a numerical real time simulation.

1 Introduction

Multi-rotors are included in the category of vertical take-off and landing (VTOL) vehicles having more than two propellers. The number of propellers or rotors defines the resulting thrust force and, consequently, the payload capacity of the aircraft. Over the past decades, the control capacity of these unmanned aerial vehicles (UAVs) has attracted the attention of researchers. This fact is due to their mechanical simplicity, simple dynamics and simple/low-cost maintenance; thus, they are considered as being ideal robotics platform for the development and testing control strategies [1].

However, standard multi-rotors UAVs possess a limited mobility due to their inherent underactuactuation [2]. A quadrotor, for instance, has 4 independent control inputs (the 4 propellers spinning velocities) and, on the other hand, 6 degrees of freedom (DOFs) which represents the system position/orientation in space. Thus, for quasi-hover conditions, a horizontal translation necessarily implies a change in the attitude, and the quadrotor can hover in place only when being in horizontal position with respect to the inertial coordinate frame.

Many authors have investigated different solutions for the underactuation problem [2], [3], [4]. Each work purposes a novel modification on the aircraft that increases the system's number of DOFs and control inputs. [3] propose an actuation concept for a quadrotor UAV in which the propellers are allowed to tilt about the axes perpendicular to the arms. [2] and [5] developed a quadrotor UAV with eight control inputs that allow its independent position and attitude control by tilting the propellers around the axes connecting them to the main body frame. Lastly, some projects consider a different multi-rotor architecture where the propellers are tilted reorienting the whole vehicle before flight without the need of any additional hardware ([6] and [7]).

Further, in general, the UAVs construction may be costly and time consuming and safety is a primary issue when conducting actual flight tests in such manner that the experimental team and the aircraft integrity must be preserved. Thus, the UAV hardware-in-the-loop simulation is an effective way to detect and prevent unnecessary malfunctions of the hardware, software and automatic control systems allowing the researchers to effectively evaluate the reliability of the overall UAV system. This method consists on a real-time simulation in which the UAV platform is reacting the same way as it in the real experiment while receiving input commands from an onboard hardware device. For instance, on [8] and [9] a real-time hardware-in-the-loop simulation framework was designed and implemented on a helicopter system UAV. The design was successfully exploited for several flight tests simulations including basic flight motions, full-envelope flight and multiple UAV formation flight.

The focus of this paper is therefore: develop a generic dynamic model for a tilt rotor multi-rotor considering the aircraft with *n* rotors and two possible tilt directions, project a MIMO controller for trajectory tracking using modern control theories, evaluate the system response via hardware-in-the-loop simulations.

2 Dynamical Modeling

The concept of this section is to derive the dynamical model of a generic multi-copter with n rotors such that each rotor is capable of tilting in two different directions (laterally and longitudinally with respect to the rotor's arm) introducing more control inputs to the system. Some considerations must be taken before the model development such as: the aircraft structure and propellers are supposed to be rigid, all the rotors and propeller blades are the same.

2.1 Kinematic Relations

Three different reference frames will be used in order to model the aircraft dynamics coupled with the rotor tilting as illustrated on Figure 1. The first, an Earth fixed reference frame Inertial Coordinate System denoted by $ICS:\{O_E; x_E, y_E, z_E\}$ used to represent the absolute position of the aircraft. Secondly, a body fixed coordinate frame represented by $BCS:\{O_B; x_B, y_B, z_b\}$ attached to the aircraft. The origin of the body fixed reference frame coincides with the aircraft center of gravity (CG), and its translational velocity and angular velocity vectors are denoted by $\vec{v}=[u \quad v \quad w]^T$ and $\vec{\omega}=[P \quad Q \quad R]^T$, respectively, such that P, Q and R are the angular velocities around the x_B, y_B and z_B axis.

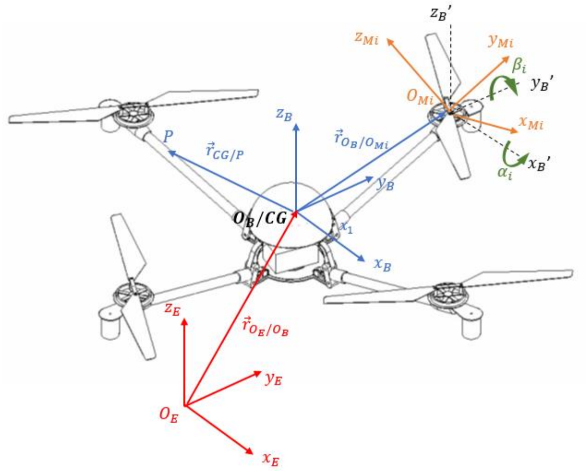


Figure 1. Multi-copter *ICS, BCS* and *MCS* axes configuration.

The aircraft attitude can be defined with respect to the *BCS* using the Euler angles, which are represented by $\Theta=[\phi \; \theta \; \psi]^T$ corresponding to the roll, pitch and yaw angles, respectively. The Euler angles angular velocities are expressed as the time rate change of the Euler angles $\dot{\Theta}=[\dot{\phi} \; \dot{\theta} \; \dot{\psi}]^T$. It must be remarked that $\omega \neq \dot{\Theta}$ since the ω vector points in the rotation axis, while $\dot{\Theta}$ only represents the time derivative of the attitude angles. However, these two vectors are correlated by a kinematic relation:

$$\omega = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \cos\phi\cos\theta \\ 0 & -\sin\phi & \cos\theta\cos\phi \end{bmatrix} \dot{\Theta} \tag{1}$$

Any vector, defined at the body-fixed frame (BCS), can be expressed at the *ICS* by using the following rotation matrix R_B^I [10],

$$\vec{r}^{\,ICS} = R_B^I \cdot \vec{r}^{\,BCS} \tag{2}$$

with, $R_B^I = R_Z(\psi)R_Y(\theta)R_X(\phi)$ and $R_Z(\psi)$, $R_Y(\theta)$ and $R_X(\phi)$ are rotations around z_E, y_E and x_E axis, respectively.

The motor reference axis $MCS:\left\{O_{Mi}; x_{m_i}, y_{m_i}, z_{m_i}\right\}$, $i = 1 \ldots n$ is the frame associated to each of the i^{th} propulsive group, with x_{m_i} representing the laterally tilting actuation axis, y_{m_i} the longitudinally tilting actuation axis and z_{m_i} the propeller actuated spinning axis that is coincident with the thrust force direction [2]. The lateral and longitudinal tilt angles are denoted by α_i and β_i, respectively. Further, the i^{th} propulsion system position w.r.t the BCS is denoted by $\vec{r}_{CG/P} = \left[l\cos\gamma_i \quad l\sin\gamma_i \quad z_{CG}\right]^T$, where l represents the multi-rotor arm length, γ_i the angle between the rotor's arm and the x_B direction and z_{CG} the distance of the rotor center of gravity to the x_B / y_B plane. Also, any vector in the MCS reference frame can be written on the BCS by a rotational matrix, following the rotation sequence around z_M, y_M and x_M, which can be represented by the matrices:

$$R_M^B(\gamma_i) = \begin{bmatrix} \cos\gamma_i & -\sin\gamma_i & 0 \\ \sin\gamma_i & \cos\gamma_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$R_M^B(\alpha_i) = \begin{bmatrix} \cos\alpha_i & 0 & \sin\alpha_i \\ 0 & 1 & 0 \\ -\sin\alpha_i & 0 & \cos\alpha_i \end{bmatrix} \quad (4)$$

$$R_M^B(\beta_i) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\beta_i & -\sin\beta_i \\ 0 & \sin\beta_i & \cos\beta_i \end{bmatrix} \quad (5)$$

Consequently, a vector on the MCS can be written on the BCS using the rotational matrix R_M^B represented by the multiplication of (3), (4) and (5) matrices, respectively, as follows:

$$\vec{q}_B = R_M^B(\gamma).R_M^B(\alpha_i).R_M^B(\beta_i).\vec{q}_M \quad (6)$$

2.2 Equations of Motion

The multi-rotor equations of motion are derived using the Newton-Euler formulation for a generic six degrees of freedom rigid body system. Thus, the equations for linear and angular body motion, written on the BCS, are:

$$\begin{bmatrix} \Sigma F_X \\ \Sigma F_Y \\ \Sigma F_Z \end{bmatrix} = m \begin{bmatrix} \dot{u} - Rv + Qw \\ \dot{v} - Pw + Ru \\ \dot{w} - Qu + Pv \end{bmatrix} \quad (7)$$

$$\begin{bmatrix} \Sigma M_X \\ \Sigma M_Y \\ \Sigma M_Z \end{bmatrix} = \begin{bmatrix} I_{XX}\dot{P} + QR\left(I_{ZZ} - I_{YY}\right) - I_{YZ}(\dot{R} + PQ) \\ I_{YY}\dot{Q} + PR\left(I_{XX} - I_{ZZ}\right) + I_{XZ}(P^2 + Q^2) \\ I_{ZZ}\dot{R} + PQ\left(I_{YY} - I_{XX}\right) + I_{XY}(QR + \dot{P}) \end{bmatrix} \quad (8)$$

where $F = \left[F_X \quad F_Y \quad F_Z\right]$ and $M = \left[M_X \quad M_Y \quad M_Z\right]$ are the external force and moment applied at the center of mass of the vehicle. I_{xx}, I_{yy}, I_{zz}, I_{xy}, I_{yz} and I_{xz} are the components of the rotational inertia matrix of the vehicle with respect to the body coordinate frame.

Concerning the external forces, it is mainly composed of thrust, drag and gravitational components. Assuming the most commonly type of propulsion system used for UAVs (DC motors), the thrust force generated by the propeller can be considered as proportional to the propeller angular speed [11]:

$$T = \left(\frac{K_\tau K_v \sqrt{2\rho\pi r_{prop}^{\ 2}}}{K_t}\Omega\right)^2 = k\Omega^2 \quad (9)$$

with K_t a proportional constant relating the torque produced by the electric motor and its electrical current, K_v relates the motor voltage and its angular velocity, K_τ is relative to the motor torque and thrust, r_{prop} is the blade disc radius, ρ the density of the surrounding air and Ω the rotational velocity of the motor shaft.

Once the thrust T acts in the z direction of the MCS, the force generated by each rotor can be written on the BCS using the matrix transformation presented in Eq. (6).

As presented in Marques (2017), a drag force due to the viscosity of the vehicle surrounding air is considered in the model which be simplified by expressing this force as proportional to the linear velocity of the aircraft and always acting in the opposite direction of the body movement. Hence, the drag force on the BCS is written as:

$$\vec{F}_D^{BCS} = -k_d \vec{v}_B = \begin{bmatrix} -k_{d\,x} u \\ -k_{d\,y} v \\ -k_{d\,z} w \end{bmatrix} \quad (10)$$

where k_d is a friction constant which can be split in three directions of the BCS (x_B, y_B and z_B).

Also, a constant gravitational force pointing always to the z-direction of the *ICS* acts on the aircraft center of gravity. The components of this force at the *BCS* is obtained from the rotation matrix (Eq. (2)), as follows:

$$\vec{F}_{grav}^{BCS} = \left(R_B^I\right)^T . F_{grav}^{ICS} = \begin{bmatrix} mg\sin\theta \\ -mg\sin\phi\cos\theta \\ -mg\cos\phi\cos\theta \end{bmatrix} \quad (11)$$

The moments acting in the *BCS* are mainly generated by propeller system actuation and how they are distributed on the UAV center of mass. Hence, the torque produced by the spinning propeller is obtained by the following relation:

$$\vec{\tau}_T = \vec{r}_{CG/P} \times F_T^{BCS} \quad (12)$$

being $\vec{F}_T^{BCS}$ the thrust force vector written on the *BCS* and $\vec{r}_{CG/P} = \begin{bmatrix} l\cos\gamma_i & l\sin\gamma_i & z_{CG_i} \end{bmatrix}^T$ the motor position vector with respect to the multi-rotor center of gravity.

Afterward, since the mass of the aircraft is generally small, the gyroscopic effect of the blades must be accounted in the dynamic model. This torque is an outcome of the propellers angular momentum direction change along the flight [13], and is calculated with respect to the multi-rotor center of gravity considering the two tilting directions (α_i and β_i). First, the motor angular velocity with respect to the aircraft *CG* is written as:

$$\vec{\omega}_{prop} = \begin{bmatrix} \cos(\alpha_i)\dot{\beta}_i + P & \dot{\alpha}_i + Q & -\sin(\alpha_i)\dot{\beta} + R \end{bmatrix}^T \quad (13)$$

where $\dot{\alpha}_i$ and $\dot{\beta}_i$ are each rotor angular velocity rate of change.

Assuming that the propeller blades have the same moment of inertia (I_M) and are spinning around the z_M direction of the *MCS*, the gyroscopic effect is obtained from by the cross product between the motor angular velocity (Eq. (13)) and the angular momentum generated by the n propellers ($\vec{H} = \begin{bmatrix} 0 & 0 & \sum_{i=1}^{n} J_{Mi} \vec{\Omega}_i \end{bmatrix}^T$):

$$\vec{\tau}_G = \left(\dot{\vec{H}}\right)_{ICS} = \left(\frac{d\vec{H}}{dt}\right)_{BCS} + \omega_B \times \vec{H} \quad (14)$$

Considering that the propeller angular speed variation is negligible and the propeller moment of inertia is constant, then $\left(d\vec{H}/dt\right)_{BCS} = 0$. Further, the fan torque is also an external torque component due to the air aerodynamic drag on the propeller blades cross section that acts on the rotor spinning axis (z_M) and can be modeled as [14]:

$$\vec{\tau}_{F_{MCS}} = \begin{bmatrix} 0 & 0 & \sum_{i=1}^{n} -b\Omega_i^2 \end{bmatrix}^T \quad (15)$$

where b is a proportional constant relating the resulting drag torque with the propeller angular speed. Equation (15) can be written on the *BCS* using the relation on Eq. (6).

Thus, the total forces and toques applied to the aircraft are:

$$\begin{aligned} \vec{F} &= \vec{F}_T^{BCS} + \vec{F}_D^{BCS} + \vec{F}_{grav}^{BCS} \\ \vec{M} &= \vec{\tau}_T + \vec{\tau}_G + \vec{\tau}_F \end{aligned} \quad (16)$$

The mathematical model of a generic tilt multi-rotor with *n* rotors is obtained substituting Eq. (16) in Eqs. (7) and (8) also considering the kinematic relations presented on Eq. (1) forming a nine degree of freedom dynamical system. The number of inputs depends on the number of motors and tilting actuators.

In order to apply modern control techniques, such as the Linear Quadratic Regulator (LQR), the derived equations of motion must be linearized in order to form a LTI (Linear Time Invariant) system [12]. The procedure of linearization is based on the small perturbation theory similarly to the application of fixed wing aircraft models as developed in [13]. Once the equations were linearized, the first order differential equations can be written in the state space format:

$$\begin{aligned} \dot{\vec{x}} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x} + D\vec{u} \end{aligned} \tag{17}$$

with $\vec{x} = \{u \quad v \quad w \quad P \quad Q \quad R \quad \varphi \quad \theta \quad \psi\}$ being the state vector, $\vec{y}$ the output measured signals, A the dynamic matrix and B the input matrix. Since the measured signals are considered to be exactly the state vector, then C is an identity matrix and D is a zero matrix.

In most applications concerning trajectory tracking problems, the position of the aircraft with respect to the *ICS* is the desired control variable [10]. In order to include the vehicle position, the state vector is expanded and three new kinematic relations are added to the problem. They concern the linear velocities ($\vec{V}_x$, $\vec{V}_y$ and $\vec{V}_z$) written as a function of the temporal derivatives of (x_E, y_E and z_E) [15].

3 Modern Control

In terms of autonomous systems development, modern control theory has revealed to be a valuable control technique for multi-copter autonomous flight applications as presented in [1] and [12]. Once this system configuration is classified as *MIMO* (Multiple Input Multiple Output), as presented on the dynamical equations, the numbers of control states and inputs in the multi-copter dynamical model make classical control techniques as PID controllers a hard-working task since the controller is based on successively closed loops. Moreover, for multi-variable or problems, modern control techniques can be more efficient since the control loop gains are calculated simultaneously while the stability condition are guaranteed.

Therefore, on this work, the Linear Quadratic Regulator (LQR) with state feedback is applied to guarantee stability and good signal tracking capability. The LQR strategy control strategy consists in making zero order closed loop system that forces all the states to the equilibrium position. Mainly concerned with trajectory tracking characteristics, the tracking error ($\vec{e}(t)$) is defined as:

$$\vec{e}(t) = \vec{r}(\mathrm{t}) - \vec{\mathrm{x}}(\mathrm{t}) \tag{18}$$

with $\vec{r}(\mathrm{t})$ the tracking command vector containing the reference values for the states desired to be tracked which can be represented as a *p* order differential equation:

$$\overset{(p)}{r} = \sum_{i=1}^{p} a_i \overset{p-i}{r} \tag{19}$$

Furthermore, the LQR problem can be extended to the LQT (*Linear Quadratic* Tracking) problem increasing the system order following the internal model principle, where the error (Eq.(18)) is driven to zero. The problem can be treated as a servomechanism design which contains the reference model (Eq.(19)) written in the state space form as [16]:

$$\overset{(p)}{r} = \sum_{i=1}^{p} a_i \overset{p-i}{r} \tag{20}$$

where the vector μ represents the plant input vector and z is the expanded state vector containing the plant and p error derivatives ($\dot{z} = \begin{bmatrix} e & \dot{e} & \cdots & \overset{(p-1)}{e} & x \end{bmatrix}^T$). Further, assuming that the control law is represented by:

$$u(t) = -K_c z \tag{21}$$

The feedback gain matrix $K_c = \begin{bmatrix} K_p & K_{p-1} & \cdots & K_1 & K_x \end{bmatrix}$ is obtained solving a cost-function minimization problem in a manner that the closed loop matrix system ($\tilde{A} - \tilde{B}K_c$) is stable (Burns, 2001):

$$J = \int_0^\infty \left((z)^T Q_k (z) + \mu^T R_k \mu \right) dt \tag{22}$$

where $Q_k = Q_k^{\ T} \geq 0$, $R_k = R_k^{\ T} \geq 0$ and $\left(\tilde{A}, Q_k^{\ 1/2} \right)$ are detectable.

Therefore, for a given state space LTI system, there is a gain matrix K_c which can minimize the cost function and bring all states (including the error) to zero simultaneously. The cost function minimization problem (Eq. (22)) can be solved using the Riccati equations [17].

4 Hardware-in-the-loop Simulation

According to [8], the hardware-in-the-loop (HIL) technique is a real-time simulation method or framework, in which the UAV is reacting the same way as it would in the real experiment. This technique has been widely applied by researchers since it can be an effective way to detect and prevent unnecessary malfunctions of hardware, software and unexpected behavior of the aircraft when executing an experiment.

For instance, in [18], a Matlab®/Simulink interface was developed in such manner that factors such as wind, sensor noise and actuator slew were included in the model to facilitate a more realistic simulation environment. In general, the benefits of the HIL is that it can reduce the experiment’s cost, increase its safety, and it can be less time consuming. Using such method, the reliability of the overall UAV system can be evaluated principally the performance of designed automatic flight control algorithms.

In this sense, a HIL framework was established in order to join in the aircraft hardware to a Matlab® flight control module which executes the automatic control algorithms and integrates the aircraft equations of motion over time. This procedure enables the LQR/LQT control law to be tested using the real hardware preserving the MAV integrity.

The HIL simulation framework is depicted in Fig. 2. Briefly, the integrated framework works as follows: the first step is a handshake connection between the microcontroller and the Matlab® so the task commands (reference signal), initial states and number of iterations are defined. The BeagleBone microcontroller receives the states from the algorithm and, using the compiled control law, calculates the motor/actuators input signal (PWM). Later, the PWM signal returns to the Matlab® algorithm and goes to the aircraft actuators/motors. The equations of motion are integrated in the software using a ODE45 function being the defined time step as the simulation time interval. The output from the Matlab® function are the states which are transferred to the microcontroller so that the loop is completed. Simultaneously, the actuators/motors receive the PWM signal, while the aircraft is fixed, and the rotors speeds are monitored using a tachometer.

The definition of the time step depends on the time communication between the Matlab® software and the BeagleBone microcontroller. The standard procedure is based on serial communication on which the controller connects directly to the software. However, for this purpose, the connection between the microcontroller and the software via TCP/IP has proved to be faster, reducing the time communication in approximately 20 times. Thus, the sample time for the HIL application could be set as 20 ms.

Notwithstanding, the time for the Matlab® routine to execute the equations of motion integration function and calculate the states is variable. Differently from a real problem where they are obtained instantly. Though, from simulations experiments was verified that the calculation time does not exceed 4 ms. Hence, setting the sample time to 20 ms, there is time enough for the software to execute the algorithm and the microcontroller to calculate the input signal. The time difference between the sample time and the software calculations is the stand-by time for the microcontroller to send the output signal back to the software. In conclusion, the proposed HIL procedure is performed as close as a real-time simulation.

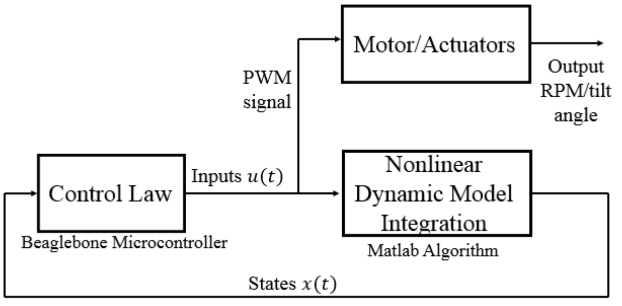


Figure 2. Hardware-in-the-loop framework.

5 Simulations and Results

Once the multi-rotor dynamical model equipped with a tilting mechanism was derived and a control law based on the linear state-space model was proposed, numerical simulations using the hardware-in-the-loop framework presented in Section 4 are presented in this section in order to illustrate the autonomous flight application.

As research object, a quadcopter configuration containing 4 longitudinal tilting mechanisms was chosen in order to exemplify the control application. In this case, the tilting mechanism are considered to actuate independently.

Further, the reference input for the three space positions is a unitary step signal represented by the following differential equation:

$$\dot{r} = 0 \tag{23}$$

with $p = 1$ and $a_1 = 0$ on Eq. (20).

The position and attitude responses are presented on Figs. 3 and 4, respectively. From the figures, it can be inferred that the aircraft position converges to the reference value on the three directions simultaneously however the system has higher oscillations on x_E direction. The oscillations are also observed on the attitude position illustrated on Fig. 4. Furthermore, regarding the yaw angle response (ψ) on Fig. 4, it is observed that the system does not converge to the equilibrium position through the simulation. This phenomenon can be attributed to the system response efficiency due to the propeller drag [19].

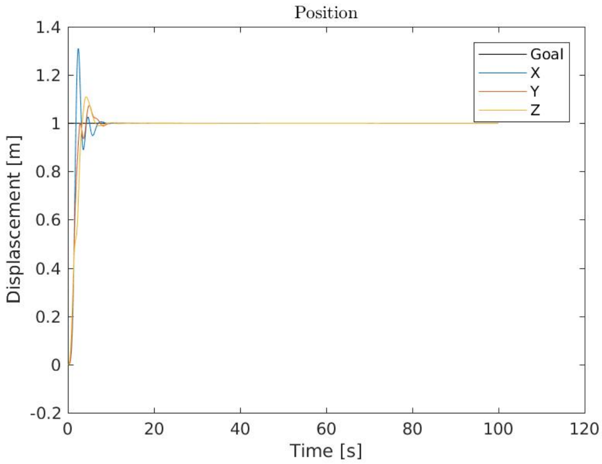


Figure 3. Quadcopter position for a unit step reference input signal.

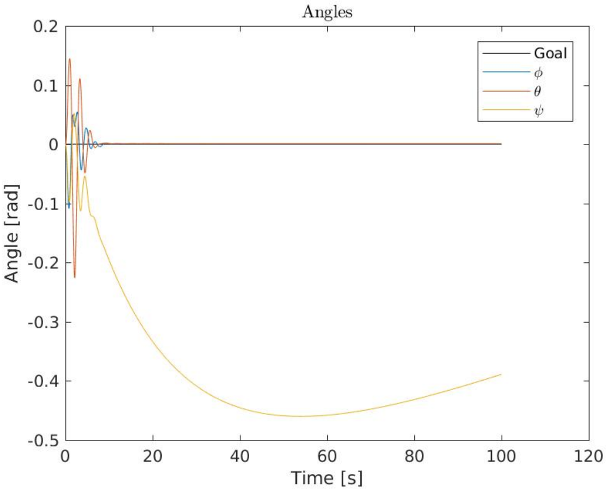


Figure 4 Quadcopter Euler angles for a unit step reference input signal.

Figures 5 and 6 show the input signal from the microcontroller to the rotors Ω (rad/s) and tilt actuator α (rad) on the numerical simulator, respectively, where one can observe that the control inputs tends to a steady state value which is not the same as the initial condition. Thus, it can be inferred that the system is at a new equilibrium condition as soon as the tilting angles are not zero (Fig. 6).

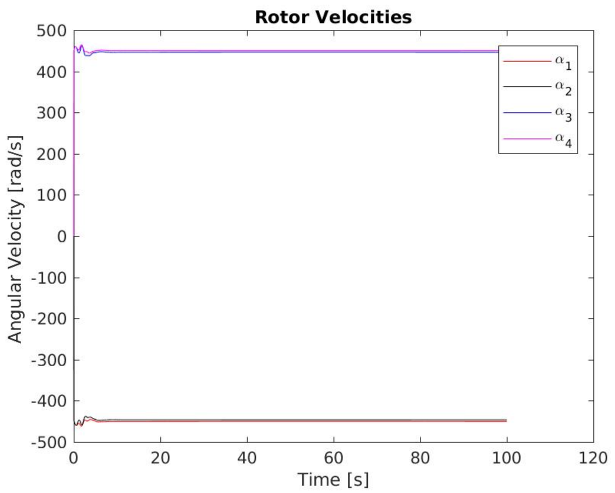


Figure 5. Quadcopter rotor angular velocities for a unit step reference input signal.

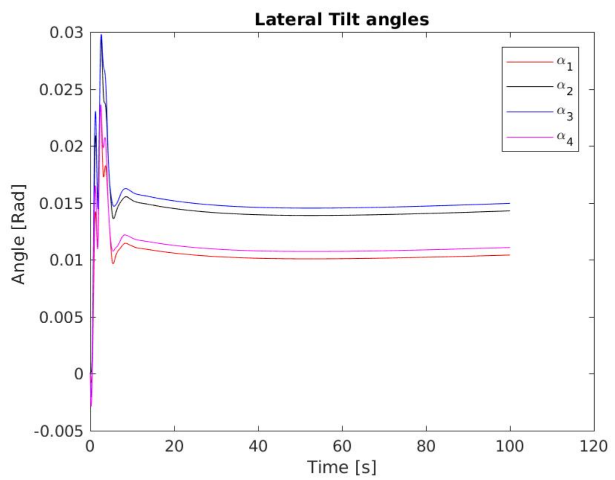
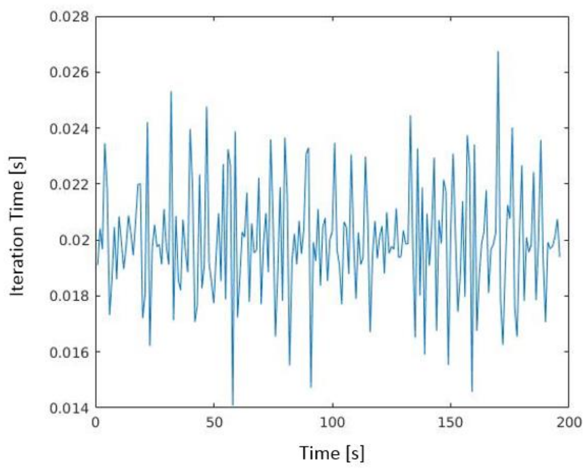


Figure 6. Quadcopter longitudinal tilting angles for a unit step reference input signal.

Once the hardware-in-the-loop simulation was performed, the performance of the system processing and communication was test to guarantee that the time elapsed between each numerical integration is enough to guarantee that the states and control inputs are updated at each time step.

Considering the time sampling, Fig. 7 shows the variation of the time calculation over the simulation. One can state that there are some peaks which exceed the required sampling time (0.02s) since it is a non-real time system however the mean value is guaranteed. Hence, in real flight operations, the transmission conditions are improved assuming that the system does not have connection delay.



The sampling time is mainly composed of three parts: plant calculation (Matlab® numerical integration, controller calculation and communication time. Ideally, the plant calculation and communication time should be very small compared to the controller calculation such that the simulated model can approximate the real system. The communication time was obtaining eliminating all the calculation and considering only the time expend for changing data, the value obtained was 5e-5s with a standard deviation of 5e-6 s, equivalent of approximately 0.5% of all the processing time as presented on Table 1.

Table 1. Processing and communication time.

Process	Percentage of total time
Plant Numerical Integration	5%
Communication	0.5%
Controller Calculation	94.5%

Also, considering the table, one can deduce that the communication time is negligible compared to the controller calculation time. Therefore, the hardware-in-the-loop procedure permits to evaluate the controller behavior on real flight conditions and to test its robustness and performance for a specific mission.

6 Conclusions

In summary, the objective of this present work was to derive the dynamical model of a generic multi-rotor UAV equipped with a tilting rotor mechanism either on the longitudinal or lateral directions. The addition of the tilting mechanism to the system increases the number of signal inputs, hence the underactuation problem can be solved.

The equations of motion were then derived using Newton-Euler formulation considering the system as a rigid body with six degrees of freedom (3 translations and 3 rotations). The dynamic equations were lately linearized so that modern control techniques could be applied. An LQR controller using servomechanism approximation for trajectory tracking was designed in a manner that the closed loop system stability is held and the plant is able to follow a desired tracking command.

The dynamical model and designed control were tested via hardware-in-the-loop simulations in order to evaluate the microcontroller performance. This analysis is a one step procedure for future experimental analysis and has given preliminary results for the controller behavior avoiding human and equipment risks.

However, for future work, other effects could be investigated for the proposed problem such as: the weighting matrix for different scenarios, the introduction of parameter uncertainties and external perturbations to the model, other control techniques application, the controller can be tested for different reference signals and input trajectories and, finally, the real experimental tests can be done.

7 References

[1] Suiçmez E. C., *Trajectory Tracking of a Quadrotor Unmanned Aerial Vehicle (UAV) Via Attitude and Position Control*. Tubitak: Middle East Technical University, 2014.

[2] Ryll, M., Bulthoff, H. H. and Giordanno, P. R., “Modeling and control of a quadrotor UAV with Tilting Propellers”, International Conference on Robotics and Automation (IEEE), Saint Paul, Minesota, USA, 2012.

[3] Badr, S., Mehrez, O. and Kabeel, A. E., “A Novel Modification for a Quadrotor Design”, International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, 2016.

[4] Oosedo, A., Abiko, S., Narasaki, S., Kuno, A., Konno, A. and Uchiyama, M., “Large attitude change flight of a quad tilt rotor unmanned aerial vehicle”, Advanced Robotics, 30:5, 325-337, 2016.

[5] Hintz, C., Torno, C. and Carrillo, L. R. G, “Design and Dynamic Modeling of a Rotary Wing Aircraft with Morphing Capabilities”, International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, Orlando, Florida, USA, 2014.

[6] Rajappa, S., Ryll, M., Bulthoff and H. H., Franchi, A., “Modeling, Control and Design Optimization for a Fully-actuated Hexarotor Aerial Vehicle with Tilted Propellers”, International Conference on Robotics and Automation (ICRA), Seattle, Washington, USA, 2015.

[7] Mehmood, H., Nakamura, T. and Johnson, E. N., “A Maneuverability Analysis of a Novel Hexarotor UAV Concept”, International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, Arlington, VA, USA, 2016.

[8] Cai G., Chen, B. M., Lee, T. H. and Fong, M., “Design and Implementation of a Hardware-in-the-loop Simulation System for Small-cale UAV Helicopters”, IEE 2008 International Conference on Automation and Logistics, Qingdao, China, 2009.

[9] Pizetta, I. H., Brandão, A. S., Sarcinelli-Filho, M.,"A hardware-in-loop platform for rotary-wing unmanned aerial vehicles", Unmanned Aircraft Systems (ICUAS) 2014 International Conference on, pp. 1146-1157, 2014

[10] Valavanis, K. P. *Advances in Unmanned Aerial Vehicles*. Dordrecht, Netherlands: Springer, 2007.

[11] Huang, H., Hoffmann, G.M. and Waslander, S. L., Tomlin, C. J. "Aerodynamics and Control of Autonomous Quadrotor Helicopters in Aggressive Maneuvering", International Conference on Robotics and Automation, IEEE, 2009.

[12] Marques F. M. M., *Modeling, Simulation and Control of a Multi-rotor Aircraft*, Monography, Federal University of Uberlândia, Uberlândia – MG, Brazil, 2016.

[13] Roskam, J. *Airplane Flight Dynamics and Automatic Flight Controls*. Lawrence, KS: DAR Corporation, 2001.

[14] Asselin, M., 1965. An Introduction to Aircraft Performance. Reston, Virginia: AIAA.

[15] Etkin, B., *Dynamics of Flight. Stability and Control*, 3rd edition. Wiley, New York, 1985.

[16] Lavretsky, E. e Wise, K. A. *Robust and Adaptive Control*, London: Springer, 2013

[17] Burns, R. S. *Advanced Control Engineering*. Oxford: Plant a Tree, 2001.

[18] Blaauw, D., Flight Control System for a Variable *Stability Blended-wing-body Unmanned Aerial Vehicle*, Master Thesis, 187 pp, University of Stellenbosch, Matieland, South Africa, 2009.

[19] MARQUES F. M. M., *Modeling, Simulation and Control of a Generic Til-rotor Multi-copter*. Master

Thesis, 99 pp, Federal University of Uberlândia, Uberlândia – MG, Brazil, 2018.

Copyright Statement