Development and Validation of a LQR-Based Quadcopter Control Dynamics Simulation Model

Article *in* International Journal of Aerospace Engineering · November 2021

DOI: 10.1061/(ASCE)AS.1943-5525.0001336

5 authors, including:

Alessandro Minervini
Politecnico di Torino
2 PUBLICATIONS **1** CITATION
SEE PROFILE

Simone Godio
Technology Innovation Institute (TII)
10 PUBLICATIONS **49** CITATIONS
SEE PROFILE

Giorgio Guglieri
Politecnico di Torino
162 PUBLICATIONS **1,902** CITATIONS
SEE PROFILE

Fabio Dovis
Politecnico di Torino
366 PUBLICATIONS **3,687** CITATIONS
SEE PROFILE

Development and Validation of a LQR-Based Quadcopter Control Dynamics Simulation Model

Alessandro Minervini[1]; Simone Godio, S.M.ASCE[2]; Giorgio Guglieri, Ph.D.[3]; Fabio Dovis, Ph.D.[4]; and Alfredo Bici[5]

Abstract: The growing applications involving unmanned aerial vehicles (UAVs) are requiring more advanced control algorithms to improve rotary-wing UAVs' performance. To preliminarily tune such advanced controllers, an experimental approach could take a long time and also be dangerous for the vehicle and the onboard hardware components. In this paper, a simulation model of a quadcopter is developed and validated by the comparison of simulation results and experimental data collected during flight tests. For this purpose, an open-source flight controller for quadcopter UAVs is developed and a linear quadratic regulator (LQR) controller is implemented as the control strategy. The input physical quantities are experimentally measured; hence, the LQR controller parameters are tuned on the simulation model. The same tuning is proposed on the developed flight controller with satisfactory results. Finally, flight data and simulation results are compared showing a reliable approximation of the experimental data by the model. Because numerous state-of-the-art simulation models are available, but accurately validated ones are not easy to find, the main purpose of this work is to provide a reliable tool to evaluate the performance for this UAV configuration. **DOI: 10.1061/(ASCE)AS.1943-5525.0001336.**

Author keywords: Quadcopter; Flight controller; Model-based design; Linear quadratic regulator (LQR) controller; Modelling; Simulation; Model validation; Experimental measurements.

Introduction

Unmanned aerial vehicles (UAVs) have been playing a key role for the last decades in several civilian applications. In particular, rotary-wing UAVs have been employed in several applications and in the coming years, companies are planning to extend their use even more. Agricultural activities make large use of UAVs for field mapping, plant stress detection, or chemical spraying (Hassler and Baysal-Gurel 2019): in some cases, drones are already guaranteeing best performance from old machinery with a significant reduction of operating times (Daponte et al. 2019). Also, aerial surveillance and surveying activities (Shahmoradi et al. 2020) have been using UAVs recently: drones can be equipped with remote sensing cameras (Baluja and Diago 2012) to detect waters in dry areas or find water leaks in underground water pipes. Some are also used to collect information from areas affected by natural disasters. In less than 10 years, UAVs could be largely employed in delivery too: exploiting the vertical dimension, delivery times would be drastically reduced and streets would also be less busy (Gulden 2017; Semsch et al. 2009).

Flight controllers for UAVs play a crucial role in guaranteeing proper flight performance and high stability for these vehicles. In particular, rotary-wing UAVs are intrinsically statically and dynamically unstable vehicles, so they need proper control algorithms to maintain self-leveling flight or perform maneuvers. Most of the commercial flight controllers for rotary-wing UAVs make use of a proportional-integrative-derivative (PID) control algorithm that can be easily tuned and implemented on a microcontroller. A PID control works properly in guaranteeing flight stability, but the increasing demand for UAVs has led many researchers to develop and test more advanced control strategies. Capello et al. (2013) proposed an L_1 adaptive controller as the inner loop autopilot of a fixed-wing UAV. In Kyaw and Gavrilov (2017), the L_1 adaptive controller was employed for the attitude control of a quadcopter, and an intuitive procedure to design the controller was provided. Model predictive control (MPC) has been largely tested on simulation models. This control strategy shows the advantage of not being based on model parameters, and it is highly effective to deal with nonlinearities and model uncertainties. Islam et al. (2017) tested a linear MPC successfully in a simulation environment for the trajectory control of a quadcopter UAV. Due to the high nonlinear effects and the vulnerability to disturbances of a quadcopter, Suhail et al. (2019) discussed the implementation of a novel control approach called active disturbance reaction (ADR). Results in the paper demonstrate the capability of the ADR to provide a faster response to disturbance than a PID controller. Also, neural network (NN) approaches have been proposed for the flight controller's inner loop design by Heryanto et al. (2017). Furthermore, in Koch et al. (2018), an inner-inner loop autopilot for a quadcopter based on reinforcement learning was developed to control the UAV's attitude.

Unlike the PID controller, such advanced control algorithms can require a long time to be tuned through experimental tests that

[1]Dept. of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi, 24, Torino 10129, Italy. Email: s265729@studenti.polito.it

[2]Ph.D. Student, Dept. of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi, 24, Torino 10129, Italy (corresponding author). ORCID: https://orcid.org/0000-0002-4474-8454. Email: simone.godio@polito.it

[3]Professor, Dept. of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi, 24, Torino 10129, Italy. Email: giorgio.guglieri@polito.it

[4]Professor, Dept. of Electronics and Telecommunications, Politecnico di Torino, Corso Duca degli Abruzzi, 24, Torino 10129, Italy. Email: fabio.dovis@polito.it

[5]Dept. of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi, 24, Torino 10129, Italy. Email: s257337@studenti.polito.it

Note. This manuscript was submitted on May 4, 2021; approved on June 7, 2021; published online on August 24, 2021. Discussion period open until January 24, 2022; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Aerospace Engineering***, © ASCE, ISSN 0893-1321.

can also be dangerous for the vehicles or onboard hardware components. Several studies have been carried out to develop a simulation model of a quadcopter and follow a model-based approach to identify the controller design. Valid results in modeling the quadcopter dynamics have been obtained though neural networks as in Al Mahasneh et al. (2017). Pairan and Shamsudin (2017) proposed a method based on a radial basis function neural network trained with a minimal resource allocating network (MRAN) algorithm: this methodology demonstrates the capability of the model to properly approximate the experimental data. This approach is justified considering the nonlinear and strongly coupled dynamics concerning quadcopters and the inability of a physics-based model to properly simulate these effects. Furthermore, to design a control algorithm, the eigenvalues and frequency domain response need to be considered, especially when filters and compensators must be added to the control scheme. A neural network–based model does not allow such an approach and the design of the control algorithm can thus be far more difficult.

At the state of art, simplified and poorly validated simulation models are mostly available for this UAV configuration. Furthermore, how to properly obtain model input data, such as aircraft inertia or engine thrust coefficient curves as the throttle varies, is often not clarified. For these reasons, this paper aims at providing a simulation tool, complete with the entire procedure of input calculation, mathematical analysis, and results, where a model-based controller design for quadcopters can be performed. The controller design is identified on the simulation model and implemented on a flight controller developed in Minervini (2021). Flight data are collected and compared with the simulation results to demonstrate the capability of the developed model to properly approximate the quadcopter attitude dynamics.

In "Quadcopter Mathematical Model" section, a nonlinear mathematical model of a quadcopter is provided and a methodology to introduce the pulse-width modulation (PWM) signal amplitude into dynamics equations is proposed. In "Model Implementation" section, the nonlinear model developed in the previous section is implemented and the linear quadratic regulator (LQR)–based controller scheme is introduced. In "Experimental Measurements" section, experimental measurements are carried out: the thrust provided by one of the propellers is evaluated and the drone's moments of inertia are measured following the procedure described in Capello et al. (2020). In "Developed Flight Controller for Model Validation" section, the developed flight controller used for the model validation is shown and its main features described. In "Model Validation and Results" section, the model validation is performed comparing the simulation results and the flight data.

Quadcopter Mathematical Model

Assumptions

In this section, a nonlinear mathematical model of the quadcopter is developed. The following assumptions are made before developing the model:

- The quadcopter is a rigid body; thus, vibrations and deformations during the flight are neglected.
- Forces and torques generated by the wind are neglected.
- The thrust provided by motors is constant with respect to the environmental conditions (air density, temperature) and altitude.

Nonlinear Model

Considering the north-east-down (NED) reference system defined as in DeRuiter et al. (2013), and the body frame as in Fig. 1, the NED and body frame are identified by $x-y-z$ and $x_b-y_b-z_b$ axes, respectively, while $\hat{\boldsymbol{e}}_{\mathbf{x}}, \hat{\boldsymbol{e}}_{\mathbf{y}}, \hat{\boldsymbol{e}}_{\mathbf{z}}$ and $\hat{\boldsymbol{e}}_{\mathbf{1}}, \hat{\boldsymbol{e}}_{\mathbf{2}}, \hat{\boldsymbol{e}}_{\mathbf{3}}$ are the three unit vectors for each frame. Euler angles are defined as in Stevens et al. (2015), and the $\boldsymbol{R}$ matrix [Eq. (1)] is found by a 3-2-1 Euler rotation sequence

$$\boldsymbol{R} = \begin{bmatrix} \cos(\theta)\cos(\psi) & \sin(\phi)\sin(\theta)\cos(\psi) - \cos(\phi)\sin(\psi) & \cos(\phi)\sin(\theta)\cos(\psi) + \sin(\phi)\sin(\psi) \\ \cos(\theta)\sin(\psi) & \sin(\theta)\sin(\phi)\sin(\psi) + \cos(\phi)\cos(\psi) & \cos(\phi)\sin(\theta)\sin(\psi) - \sin(\phi)\cos(\psi) \\ -\sin(\theta) & \sin(\phi)\cos(\theta) & \cos(\phi)\cos(\theta) \end{bmatrix} \tag{1}$$

Hence, defining $\boldsymbol{V} = [\dot{x}, \dot{y}, \dot{z}]^T$, the velocity of the aircraft in the inertial frame, and $\boldsymbol{V_B} = [u, v, w]^T$, the velocity of the aircraft in the body frame, the following kinematics equation [Eq. (2)] can be written:

$$\boldsymbol{V} = \boldsymbol{R} \cdot \boldsymbol{V_B} \tag{2}$$

As in Stevens et al. (2015), considering the vector $\boldsymbol{\omega} = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$ whose components are the angular rates involved in the 3-2-1 Euler rotation sequence, and the angular rate of the body frame with respect to the NED frame $\boldsymbol{\omega_B} = [p, q, r]^T$, Eq. (3) can be written as

$$\boldsymbol{\omega} = \boldsymbol{T} \cdot \boldsymbol{\omega_B} \tag{3}$$

where the $\boldsymbol{T}$ matrix is defined by Eq. (4)

$$\boldsymbol{T} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \dfrac{\sin(\phi)}{\cos(\theta)} & \dfrac{\cos(\phi)}{\cos(\theta)} \end{bmatrix} \tag{4}$$

Eqs. (2) and (3) are the kinematics equations of the model.

Newton's law [Eq. (5)] and Euler's equation [Eq. (6)] provide the translational and rotational dynamics of a six-degrees-of-freedom rigid body. The two equations are written as follows:

$$\boldsymbol{F_B} = m(\boldsymbol{\omega_B} \wedge \boldsymbol{V_B} + \dot{\boldsymbol{V}}_B) \tag{5}$$

$$\boldsymbol{M_B} = \boldsymbol{I} \cdot \dot{\boldsymbol{\omega}}_{\boldsymbol{B}} + \boldsymbol{\omega_B} \wedge (\boldsymbol{I} \cdot \boldsymbol{\omega_B}) \tag{6}$$

where $\boldsymbol{F_B} = [F_x, F_y, F_z]^T$ is the vector containing the total force applied to the aircraft in body frames; m = mass of the aircraft; $\boldsymbol{I}$ = diagonal inertia matrix; and $\boldsymbol{M_B} = [M_x, M_y, M_z]^T$ is the vector containing the total torque applied to the body in the body frame. The $\boldsymbol{I}$ matrix is defined by Eq. (7)

$$\boldsymbol{I} = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{7}$$

where I_x, I_y, and I_z = moment of inertia along the three body axes. The $\boldsymbol{I}$ matrix is assumed to be diagonal due to the symmetry of the quadcopter's frame with respect to the body reference system.

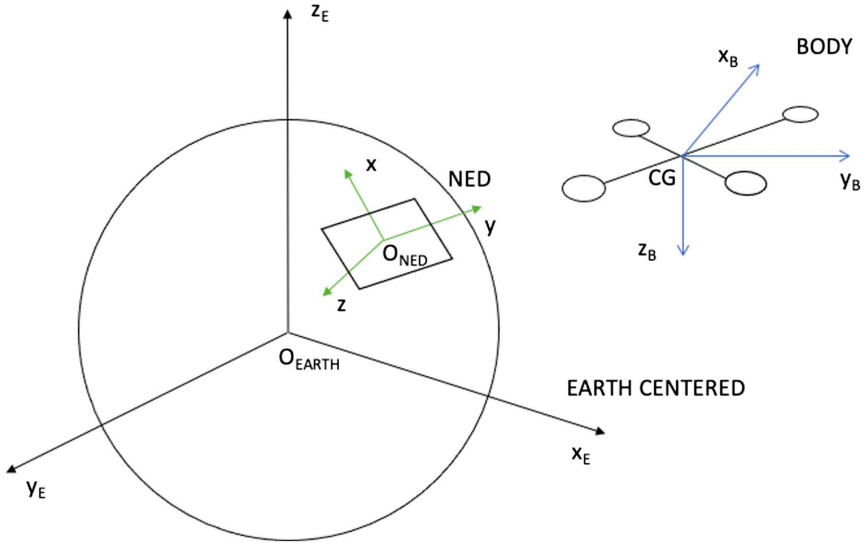


Fig. 1. NED and body reference systems.

The vector $\boldsymbol{F_B} = [F_x, F_y, F_z]^T$ contains the total forces applied to the quadcopter along the three body axes and it is given by Eq. (8)

$$\boldsymbol{F_B} = mg\boldsymbol{R}^T \cdot \hat{\boldsymbol{e}}_\boldsymbol{z} - F_t\hat{\boldsymbol{e}}_\boldsymbol{3} \tag{8}$$

where $\hat{\boldsymbol{e}}_\boldsymbol{z}$ = unit vector in the inertial z-axis; g = module of the gravitational acceleration; and F_t = module of the total thrust provided by the four motors. The forces due to the wind or other disturbances are neglected in this treatment. The module of the total thrust provided by the four motors can be written as in Eq. (9)

$$F_t = T_1 + T_2 + T_3 + T_4 \tag{9}$$

where T_1, T_2, T_3, T_4 = values of the thrust of each motor.

The vector $\boldsymbol{M_B} = [M_x, M_y, M_z]^T$ contains the total torques applied to the quadcopter in the body frames and it is given by Eq. (10)

$$\boldsymbol{M_B} = \boldsymbol{\tau_B} + \boldsymbol{g_m} \tag{10}$$

where $\boldsymbol{\tau_B} = [\tau_x, \tau_y, \tau_z]^T$ is the vector containing the control torques provided by the quadcopter's motors; and $\boldsymbol{g_m} = [g_{mx}, g_{my}, g_{mz}]^T$ is the vector containing the gyroscopic torques caused by the combined rotation of the four rotors and the aircraft body. Torques due to the wind or other disturbances are neglected in this treatment.

The vector $\boldsymbol{\tau_B}$ is found referring to Fig. 2 by Eq. (11)

$$\begin{aligned}\tau_x &= (T_1 - T_2 - T_3 + T_4)L\sin(\theta)\\ \tau_y &= (T_1 + T_2 - T_3 - T_4)L\cos(\theta)\\ \tau_z &= -C_1 + C_2 - C_3 + C_4\end{aligned} \tag{11}$$

where L = length from the rigid body's center of gravity to the motors; θ is defined as in Fig. 2; and C_1, C_2, C_3, C_4 = contrast torques provided by each rotor.

The vector $\boldsymbol{g_m}$ is given by the following relation [Eq. (12)]:

$$\boldsymbol{g_m} = \sum_{i=1}^{4} J_p(\boldsymbol{\omega_B} \wedge \hat{\boldsymbol{e}}_\boldsymbol{3})(-1)^{i+1}\Omega_i \tag{12}$$

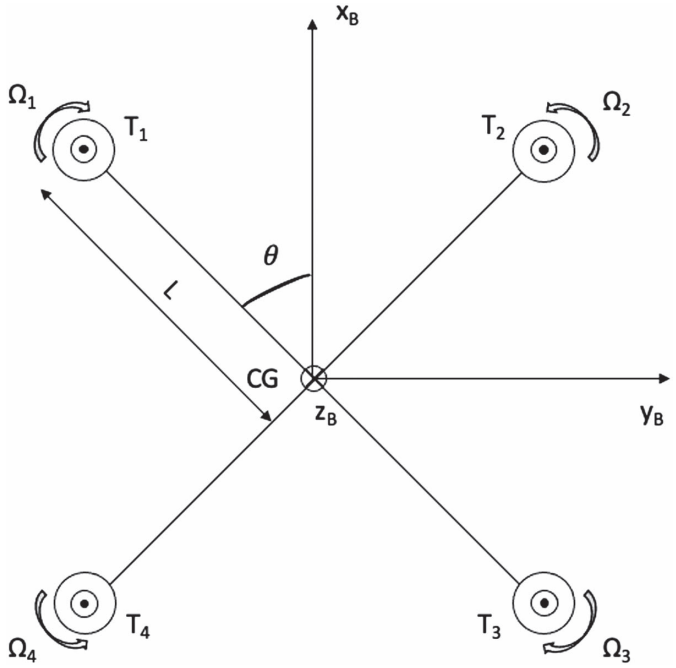


Fig. 2. Top view of the quadcopter.

where J_p = moment of inertia of the propellers; $\hat{\boldsymbol{e}}_\boldsymbol{3}$ = unit vector in the body z-axis; and Ω_i = angular rate of the ith rotor.

The mathematical model described previously gives the quadcopter's attitude and the position of the applied torques and forces are known. Such relations cannot be employed for a model-based approach because the command inputs should be a PWM pulse amplitude instead of a control torque. Therefore, a methodology to convert the control torque in the equivalent PWM pulse amplitude is proposed.

The variation of the thrust vector $\boldsymbol{\Delta T} = [\Delta T_x, \Delta T_y, \Delta T_z]$ contains the variation of the thrust that generates a torque along the three body axes. Referring to Fig. 2, Eq. (13) can be found

$$\Delta T_x = \frac{\tau_x}{4L\sin(\theta)}$$
$$\Delta T_y = \frac{\tau_y}{4L\cos(\theta)}$$
$$\Delta T_z = \frac{\tau_z}{4} \tag{13}$$

Assuming a linear relation between the propeller's thrust $\boldsymbol{T}$ and the PWM pulse width, Eq. (14) can be written

$$\tau_x = 4n_T L\sin(\theta)\Delta\text{PWM}_x$$
$$\tau_y = 4n_T L\cos(\theta)\Delta\text{PWM}_y$$
$$\tau_z = 4n_C\Delta\text{PWM}_z \tag{14}$$

where n_T = slope of the linear relation between the propeller's thrust and the PWM pulse width; and n_C = slope of the linear relation between the motor's contrast torque and the PWM pulse width.

Finally, the mathematical model that allows a model-based approach to design a quadcopter controller is proposed in Eq. (15)

$$\begin{aligned}
\dot{x} &= w[\sin(\phi)\sin(\psi) + \cos(\phi)\cos(\psi)\sin(\theta)] \\
&\quad - v[\cos(\phi)\sin(\psi) - \sin(\phi)\cos(\psi)\sin(\theta)] + u[\cos(\psi)\cos(\theta)] \\
\dot{y} &= v[\cos(\phi)\cos(\psi) + \sin(\phi)\sin(\psi)\sin(\theta)] \\
&\quad - w[\cos(\psi)\sin(\phi) - \cos(\phi)\sin(\psi)\sin(\theta)] + u[\cos(\theta)\sin(\psi)] \\
\dot{z} &= w[\cos(\phi)\cos(\theta)] - u[\sin(\theta)] + v[\cos(\theta)\sin(\phi)] \\
\dot{\phi} &= p + r[\cos(\phi)\tan(\theta)] + q[\sin(\phi)\tan(\theta)] \\
\dot{u} &= rv - qw - g\sin(\theta) \\
\dot{v} &= pw - -ru + g\sin(\phi)\cos(\theta) \\
\dot{w} &= qu - pv + g\cos(\theta)\cos(\phi) - F_t \\
\dot{\theta} &= q[\cos(\phi)] - r[\sin(\phi)] \\
\dot{\psi} &= r\frac{\cos(\phi)}{\cos(\theta)} + q\frac{\sin(\phi)}{\cos(\theta)} \\
\dot{p} &= \frac{I_y - I_z}{I_x}rq + \frac{4n_T L\sin(\theta)\Delta\text{PWM}_x}{I_x} \\
\dot{q} &= \frac{I_z - I_x}{I_y}pr + \frac{4n_T L\cos(\theta)\Delta\text{PWM}_y}{I_y} \\
\dot{r} &= \frac{I_x - I_y}{I_z}pq + \frac{4n_C\Delta\text{PWM}_z}{I_z}
\end{aligned} \tag{15}$$

Model Implementation

Assumptions

The following assumptions have been made to implement the mathematical model in a simulation environment:

- All quadcopter's states (attitude and position) are exactly known from the mathematical model: the navigation block is not implemented.
- Direct current (DC) motor dynamics are simulated through a first-order transfer function to introduce delays in the response.
- The control action is performed at 400-Hz frequency, which is the same refresh frequency of the flight controller developed to validate the model.
- The quadcopter is controlled by an attitude signal for roll and pitch, while an angular rate signal is used to control the yaw dynamics.

Simulation Model

The developed simulation model is shown in the block diagram of Fig. 3. It consists of three main blocks:

- Controller: the LQR controller is implemented in this block and the PWM signal amplitude is given as output.
- DC motor: DC motor dynamics are simulated.
- Plant: the kinematics and dynamics model described by Eq. (15) is implemented.

The LQR controller consists of a $\boldsymbol{K}$ gain matrix obtained by the MATLAB version R2018a function $lqr(\boldsymbol{A},\boldsymbol{B},\boldsymbol{Q},\boldsymbol{R})$, where $\boldsymbol{A}$ and $\boldsymbol{B}$ are respectively the state and input matrix of the linearized state-space representation of Eq. (15) shown in the Appendix, while $\boldsymbol{Q}$ and $\boldsymbol{R}$ are respectively defined as the state and control weighting matrices. The $\boldsymbol{K}$ gain matrix is found by minimizing a cost function defined by Eq. (16)

$$J(\boldsymbol{x},\boldsymbol{u}) = \frac{1}{2}\int_0^\infty \boldsymbol{x}^T(\boldsymbol{Q} + \boldsymbol{K}^T\boldsymbol{R}\boldsymbol{K})dt \tag{16}$$

where $\boldsymbol{u}$ and $\boldsymbol{x}$ vectors are defined in the Appendix.

The DC motors are simulated by first-order transfer functions [Eq. (17)]

$$\text{PWM} = \frac{1}{\tau s + 1}\text{PWM}_{\text{ideal}} \tag{17}$$

where $\text{PWM}_{\text{ideal}}$ = PWM signal without the delays introduced by the DC motors. The τ value is the constant time of the motors, and it is assumed to be 0.02 s from similar works, while s is the complex variable from Laplace transformation in the frequency domain.

Such a feedback-control configuration is used to allow the plant to track a reference signal: pitch and roll dynamics follow

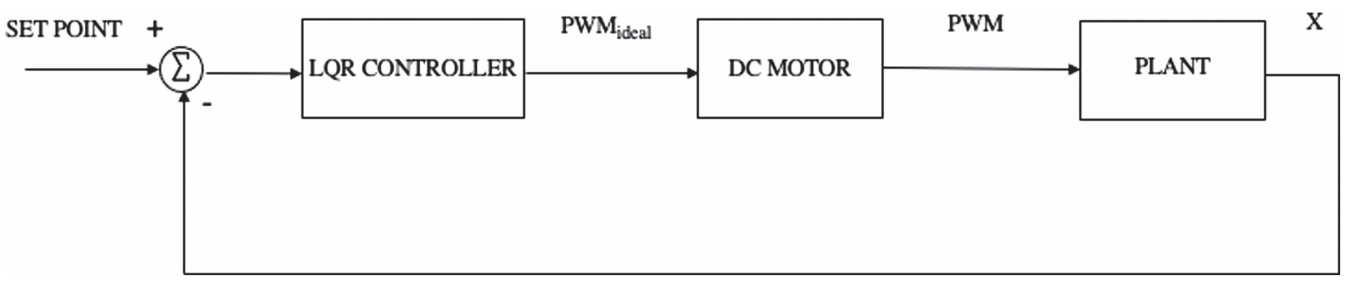
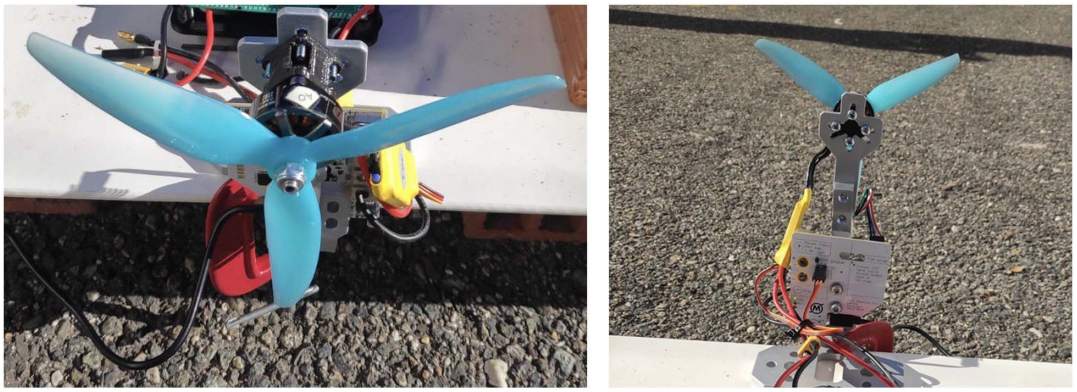


Fig. 3. Model architecture implemented in Simulink.



(a) (b)

Fig. 4. Test bench for propeller's thrust measurements.

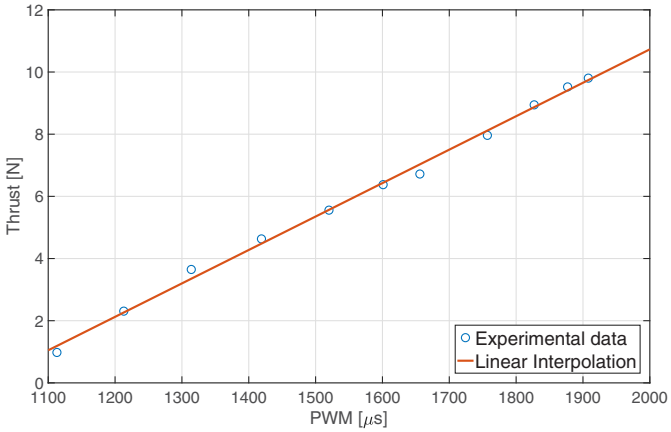


Fig. 5. Linear interpolation of thrust experimental data.

a reference angle, while the yaw angle is controlled by commanding an angular rate to make drone driving more intuitive.

Experimental Measurements

Experimental measurements have been carried out to find physical quantities related to roll and pitch dynamics. These quantities are given as input to the simulation model and a comparison between simulation results and flight data is performed.

Propeller tests have been performed to evaluate the n_T coefficient of Eq. (15). The RCBenchmark (Gatineau, Canada) Series 1520 thrust stand (Fig. 4) is a small-size propeller test stand, and it was used to collect data about the propellers. RCBenchmark provides dedicated software to collect the trust provided by the propellers when the PWM signal varies. Fig. 5 shows the interpolation of the experimental data and the linear relation is described by Eq. (18)

$$T = n_T \mathrm{PWM} + q \tag{18}$$

where $n_T = 0.010757\ \mathrm{N}/\mu\mathrm{s}$ and $q = -10.784300\ \mathrm{N}$ are found from the interpolation.

The moments of inertia of the quadcopter are measured following the procedure shown in Capello et al. (2020). The test bench consists of a pendulum where the drone is built at the extremity (Fig. 6). Evaluating the kinetic and potential energy of the

Fig. 6. Pendulum used to measure I_x and I_y.

pendulum and applying the Lagrangian equation, the moment of inertia of a built on body can be obtained by Eq. (19)

$$I = \frac{T_I^2}{4\pi^2}\left(m_1 g \frac{l_1}{2} + mg(l_1 + d)\right) - \frac{m_1 l_1^2}{4} - m(l_1 + d)^2 - I_{\mathrm{rod}} \tag{19}$$

where I = moment of inertia of the drone; T_I = period of oscillation; m_1 = rod mass; l_1 = distance between the joint and the center of gravity (CoG) of the rod; d = distance between the CoG of the bar and the CoG of the drone; and I_{rod} = rod's moment of inertia, which is referred to as the rotation axis of the pendulum.

Fig. 7. Flight setup: the quadcopter and the developed flight controller.

The measured periods are $T_{Ix} = 1.880$ s for I_x and $T_{Iy} = 1.876$ s for I_y. By using Eq. (19), the drone's moments of inertia are found as $I_x = 0.0231$ kg · m^2 and $I_y = 0.0282$ kg · m^2. As expected, I_y results are grater than I_x due to the battery and other hardware devices that contribute to increase the moment of inertia with respect to y_b-axis.

Developed Flight Controller for Model Validation

An open-source flight controller (Fig. 7) has been developed to perform the model validation and test the controller design identified on the simulation model. The developed flight controller software runs on an STM-32 Arduino microcontroller (Genève) at 400 Hz to provide a fast response to disturbances and commands. An MPU-6050 (Sunnyvale, California) provides the drone's angular rates and acceleration in the body frame by its micro-electromechanical system (MEMS) gyroscope and accelerometer. A sensor fusion between these data is performed and a complementary filter is applied to obtain a noise-free and accurate attitude from the sensor. The LQR controller is implemented with the same scheme as in Fig. 3. Data are collected by an SD card and then they are compared with the simulation model.

Before testing the quadcopter during flight, the controller design identified on the simulation model was tested on a bench (Fig. 8). Roll, pitch, and yaw dynamics could be performed thanks to a joint that constrains the drone translations.

Model Validation and Results

Once the input physical quantities were measured, the LQR controller was designed through a trial-and-error procedure. Starting from the identity matrix, $\boldsymbol{Q}$ and $\boldsymbol{R}$ were identified to perform a fast and stable step response with low overshoots. In this phase, it was considered that the $\boldsymbol{Q}$ matrix acts to optimize the error between observed states and set-point values, while the $\boldsymbol{R}$ matrix acts to optimize the command sent to motors. By increasing the diagonal components of the $\boldsymbol{Q}$ matrix, the angle and angular rate errors are penalized by the algorithm. Similarly, by acting on diagonal components of the $\boldsymbol{R}$ matrix, the actuators' effort can be reduced. Because the starting design with identity matrices showed a very slow response, all the $\boldsymbol{R}$ matrix components were decreased up to 10^{-6} order of magnitude. This design resulted in a faster response but with higher power consumption. Therefore, the angle and angular rate error have been optimized by acting on the $\boldsymbol{Q}$ matrix. Especially, the components have been decreased up to 10^{-2} order of magnitude to obtain low overshoots and errors as regards the first three diagonal components, and up to 10^{-3} as regards the other three components. The results are shown in Fig. 9.

The same LQR design was tested on the flight controller described previously. Roll and pitch dynamics perform satisfactory results with the model-based design as shown in Fig. 10. Two different flight tests are reported: the first shows the autostabilizing flight is performed properly by the quadcopter, while the second shows that even fast commands can be tracked properly by the vehicle. Finally, a comparison of the simulation results and flight tests is provided. In Fig. 11, the roll and pitch angles are compared and the corresponding absolute error is shown. The model approximates the experimental data with an absolute error under 3° for most of the test. The error increases when fast commands are required, such as from about 57 to 70 s for roll dynamics. However, it remains lower than 5°, except for some peaks. The pitch matching

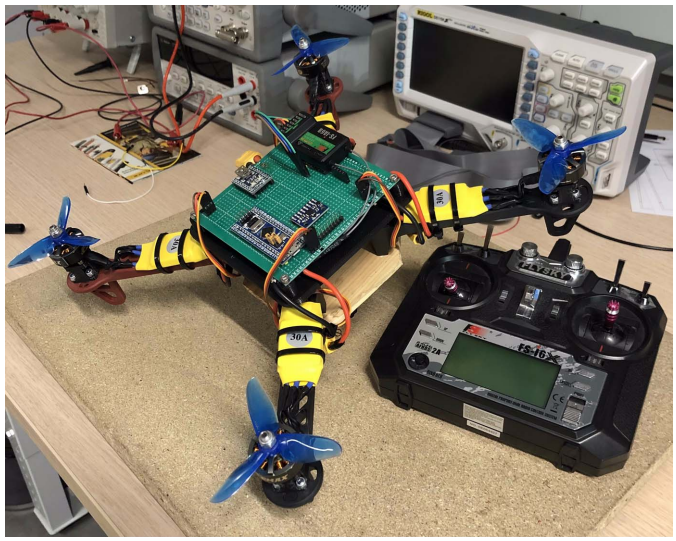


Fig. 8. Test bench for preliminary tuning of the controller.

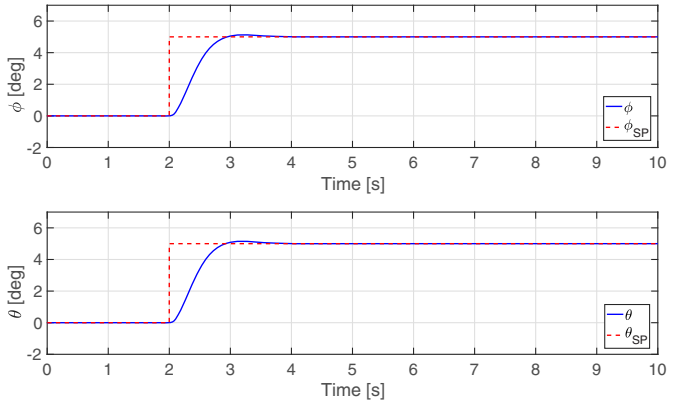


Fig. 9. Simulation model: roll and pitch step response.

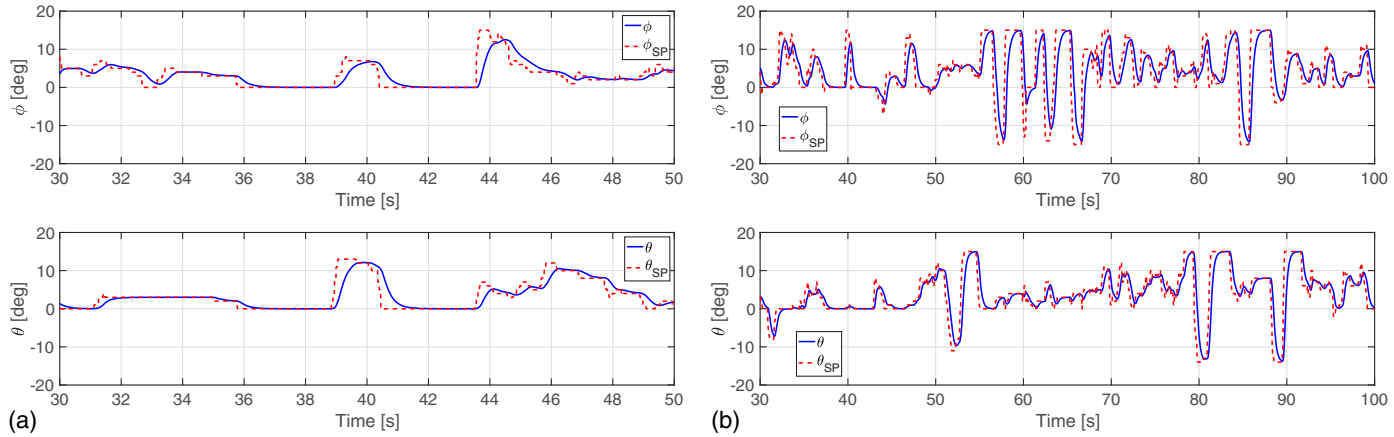


Fig. 10. Flight data: roll and pitch angles.

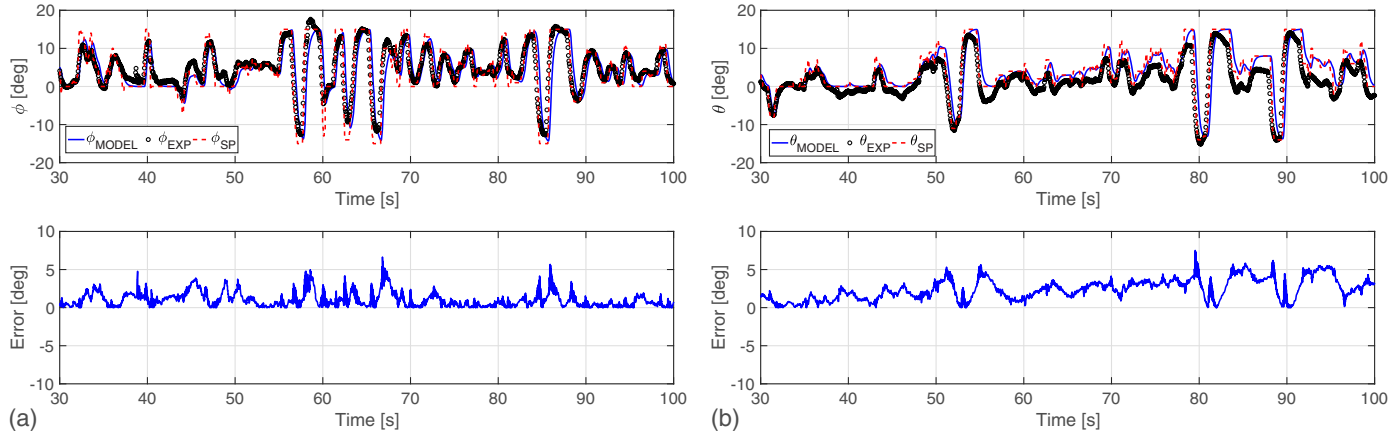


Fig. 11. Comparison of experimental data and simulation results.

Table 1. Comparison of rising times

Angle	Simulation model (s)	Experimental data (s)	Error (%)
Roll	0.69	0.62	10
Pitch	0.60	0.57	5

seems to be a bit worse than the roll one: probably the inertial measurement unit (IMU) was not built on the drone at exactly 0°; hence, the true pitch angle could be closer to the simulation results. Furthermore, the approximation remains satisfactory, especially considering the comparison of the rising time between the two responses. In Table 1, the rising times of the simulation results and experimental data are collected. An error of 10% for roll and 5% for pitch dynamics shows the capability of the model to properly evaluate this time-domain specification, which is the most demanding one to make the drone controllable.

Conclusions and Further Developments

Due to the increasing demand for more and more advanced UAV performances, the experimental approach based on the trial-and-error procedure needs to be overcome.

In this paper, a simulation model of a quadcopter UAV has been developed and validated from the comparison of the simulation results and experimental data. Despite the highly nonlinear and strongly coupled dynamics, the simulation model can be successfully employed to design the control algorithm with a model-based approach.

In this paper, an experimental procedure to obtain physical parameters is shown in detail and validated from the final comparisons. The experiement proves that the quadcopter dynamics can be successfully approximated by using Newton's and Euler's equations, neglecting wind disturbances and other nonlinear effects. As shown by the results, the error between simulation and experimental data increases when fast and large-amplitude commands are required. Nevertheless, the rising time can be estimated with a reduced error by the developed model, and control parameters can be identified properly in the simulation environment to fulfill this very demanding specification.

It is not common to find simulation models for this aircraft configuration with solid validation data. For these reasons, the authors developed this work in hopes of providing a comprehensive and useful design and performance estimation tool for the scientific community for this UAV configuration.

Future works will focus on the validation of the procedure for other quadcopter UAVs that make use of the well-known PX4 autopilot (Zurich, Switzerland). Also, the proposed mathematical model can be easily modified to simulate other rotary-wing UAV configurations (i.e., optacopter, hexacopter). Afterward, more advanced control algorithms that cannot be tuned with an experimental approach can be tested to improve flight performance.

Appendix. Linear Model

The system introduced by Eq. (15) can be linearized around an equilibrium point and the state-space representation can be written [Eq. (20)]

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y} &= \boldsymbol{C}\boldsymbol{x}(t) \end{aligned} \tag{20}$$

where $\boldsymbol{x}(t) = [x, y, z, \phi, \theta, \psi, u, v, w, p, q, r]^T$ is the state vector, $\boldsymbol{u}(t) = [F_t, \Delta\text{PWM}_x, \Delta\text{PWM}_y, \Delta\text{PWM}_z,]^T$ is the control signal, and $\boldsymbol{y}(t)$ is the controlled output. The symbols $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ are defined as follows:

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -g & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & g & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{21}$$

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{1}{m} & 0 & 0 & 0 \\ 0 & \frac{4n_T L \sin(\theta)}{I_x} & 0 & 0 \\ 0 & 0 & \frac{4n_T L \sin(\theta)}{I_y} & 0 \\ 0 & 0 & 0 & \frac{4n_C}{I_z} \end{bmatrix} \tag{22}$$

$$\boldsymbol{C} = \boldsymbol{I}_{12} \tag{23}$$

where $\boldsymbol{I}_{12} \in \mathbb{R}^{12,12}$ and it is the identity matrix.

Data Availability Statement

Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments

The research program is shared with PIC4SeR: Politecnico di Torino Interdepartmental Center for Service Robotics (https://pic4ser.polito.it/). Contributions to this paper were as follows: Conceptualization, A.M. and S.G.; methodology, A.M. and S.G.; software, A.M. and A.B.; validation, A.M., A.B., and S.G.; formal analysis, S.G. and G.G.; investigation, A.M. and S.G.; resources, G.G. and F.D.; data curation, A.M., A.B., and S.G.; original draft preparation, A.M.; review and editing, S.G. and G.G.; visualization, A.M. and S.G.; supervision, G.G. and F.D.; project administration, G.G. and F.D.; funding acquisition, G.G. and F.D. All authors have read and agreed to the published version of the paper.

References

Al Mahasneh, A. J., S. G. Anavatti, and M. Garratt. 2017. "Nonlinear multi-input multi-output system identification using neuro-evolutionary methods for a quadcopter." In *Proc., 9th Int. Conf. on Advanced Computational Intelligence (ICACI)*, 217–222. New York: IEEE.

Baluja, J., and M. Diago. 2012. "Assessment of vineyard water status variability by thermal and multispectral imagery using an unmanned aerial vehicle (UAV)." *Irrig. Sci.* 30 (6): 511–522. https://doi.org/10.1007/s00271-012-0382-9.

Capello, E., G. Guglieri, and F. Quagliotti. 2013. "Design and validation of an L1 adaptive controller for Mini-UAV autopilot." *J. Intell. Rob. Syst.* 69 (1): 109–118. https://doi.org/10.1007/s10846-012-9717-2.

Capello, E., H. Park, B. Tavora, G. Guglieri, and M. Romano. 2020. "Modeling and experimental parameter identification of a multicopter via a compound pendulum test rig." In *Proc., 2015 Workshop on Research, Education and Development of Unmanned Aerial Systems*. New York: IEEE.

Daponte, P., L. De Vito, L. Glielmo, L. Iannelli, D. Liuzza, F. Picariello, and G. Silano. 2019. "A review on the use of drones for precision agriculture." *IOP Conf. Ser.: Earth Environ. Sci.* 275 (1): 012022. https://doi.org/10.1088/1755-1315/275/1/012022.

DeRuiter, A., C. Damaren, and J. Forbes. 2013. *Spacecraft dynamics and controls*. New York: Wiley.

Gulden, T. 2017. *The energy implications of drones for package delivery: A geographic information system comparison*. Santa Monica, CA: RAND.

Hassler, S., and F. Baysal-Gurel. 2019. "Unmanned aircraft system (UAS) technology and applications in agriculture." *Agronomy* 9 (10): 618. https://doi.org/10.3390/agronomy9100618.

Heryanto, M., H. Suprijono, B. Y. Suprapto, and B. Kusumoputro. 2017. "Attitude and altitude control of a quadcopter using neural network based direct inverse control scheme." *Adv. Sci. Lett.* 23 (5): 4060–4064. https://doi.org/10.1166/asl.2017.8328.

Islam, M., M. Okasha, and M. Idres. 2017. "Dynamics and control of quadcopter using linear model predictive control approach." *IOP Conf. Ser.: Mater. Sci. Engi.* 270 (1): 012007. https://doi.org/10.1088/1757-899X/270/1/012007.

Koch, W., R. Mancuso, R. West, and A. Bestavros. 2018. "Reinforcement learning for UAV attitude control." *ACM Trans. Cyber-Phys. Syst.* 3 (2): 1–21. https://doi.org/10.1145/3301273.

Kyaw, M. T., and A. I. Gavrilov. 2017. "Designing and modeling of quadcopter control system using L1 adaptive control." *Procedia Comput. Sci.* 103 (Jan): 528–535. https://doi.org/10.1016/j.procs.2017.01.046.

Minervini, A. 2021. "Development of an open-source flight controller for rotary wing UAVs." M.S. thesis, Dept. of Mechanical and Aerospace Engineering, Politecnico di Torino.

Pairan, M., and S. Shamsudin. 2017. "System identification of an unmanned quadcopter system using MRAN neural." *IOP Conf. Ser.: Mater. Sci. Eng.* 270 (1): 012019. https://doi.org/10.1088/1757-899X/270/1/012019.

Semsch, E., M. Jakob, D. Pavlicek, and M. Pechoucek. 2009. "Autonomous UAV surveillance in complex urban environments." In Vol. 2 of *Proc., 2009 IEEE/WIC/ACM Int. Conf. on Intelligent Agent Technology*, 82–85. New York: IEEE.

Shahmoradi, J., E. Talebi, P. Roghanchi, and M. Hassanalian. 2020. "A comprehensive review of applications of drone technology in the mining industry." *Drones* 4 (3): 34. https://doi.org/10.3390/drones4030034.

Stevens, B., F. L. Lewis, and E. N. Johnson. 2015. "Modeling the aircraft." In *Aircraft control and simulation: Dynamics, controls design, and autonomous systems*. New York: Wiley.

Suhail, S. A., M. Bazaz, and S. Hussain. 2019. "Altitude and attitude control of a quadcopter using linear active disturbance rejection control." In *Proc., 2019 Int. Conf. on Computing, Power and Communication Technologies (GUCON)*, 281–286. New York: IEEE.