



Simulation Study of Quadcopter Altitude Control

Muhammad 'Azim¹, Ramhuzaini Abd Rahman^{1*}

¹Faculty of Mechanical and Manufacturing Engineering (FKMP),
 Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Johor, MALAYSIA.

*Corresponding Author

DOI: <https://doi.org/10.30880/paat.2022.02.02.001>

Received 14 November 2022; Accepted 26 December 2022; Available online 31 December 2022

Abstract: In this paper, virtual simulation of a quadcopter altitude control using MATLAB Simscape is presented. In the simulation, a PID controller is utilized and its performance in commanding the quadcopter to change its altitude is analyzed. Both time response and frequency response analysis have been performed and evaluated. Analysis in time response has shown that the controller managed to achieve steady-state condition in 3.91s with zero error and 13.9% overshoot for a step input of 1 meter. In terms of frequency response, the controller achieved a bandwidth of up to 10 rad/s. Hence, it can be concluded that the utilized controller can perform altitude control and achieved moderate performance. Nevertheless, performance of the controller can be further improved with proper tuning of the controller gains using a more systematic tuning method such as Ziegler-Nichols.

Keywords: Quadcopter, altitude control, PID controller, MATLAB Sims cape

1. Introduction

Control system engineering is a branch of engineering that employs concepts from control theory to construct a system that produces the desired behavior under controlled conditions. A well-calibrated control system is essential to achieving a desirable result in the system. The field of unmanned aerial vehicles (UAVs), in which quadcopters are included, is one application of control engineering systems. The performance of the quadcopter's control system has been enhanced throughout the years despite its extensive use. Common uses for quadcopters include agriculture, medicine, and the military [1][2][3]. It is crucial that this study's control system design yields correct results to ensure that the quadcopter does not crash while in use. Not employing simulation when creating a control system can lead to inability to simulate different flight condition and the needs of conducting costly, time-consuming, and inefficient experiments on a small scale. This research will focus on an investigation of the vertical movement of a quadcopter. In recent years, many potential uses for quadcopters have been suggested. Research on altitude controllers has covered a variety of topics, including the effectiveness of attitude and altitude stabilization under disturbance conditions [6], the wide variety of position control applications for unmanned vehicles [4], and Proportional Integral Derivative and Integral Backstepping techniques used to ensure stable vertical position stability [5]. According to previous studies [7], if the values of the Euler angles (θ , ψ , and ϕ) are unknown, the quadcopter will be unable to achieve its altitude in an uncontrolled response. To attain the hovering altitude required by the quadcopter, this study aims to design a PID controller to take input signals and transfer them to all the rotors. The remaining parts of the paper are structured as follows: In Section 2, the architecture of a quadcopter altitude control system is described. Section 3 provides information on the quadcopter simulator. Simulation evaluation is discussed in Section 4, along with the results and discussion, and a conclusion is presented in Section 5.

*Corresponding author: huzaini@uthm.edu.my

2. Architecture of Altitude Control System of Quadcopter

The controller subsystem is connected to the input signal that has been set for the desired altitude value for the quadcopter's rotor to spin. The controller's function is to minimize the difference between the desired value of the system and the actual value gained throughout the system. Fig. 1 below shows the altitude control system.

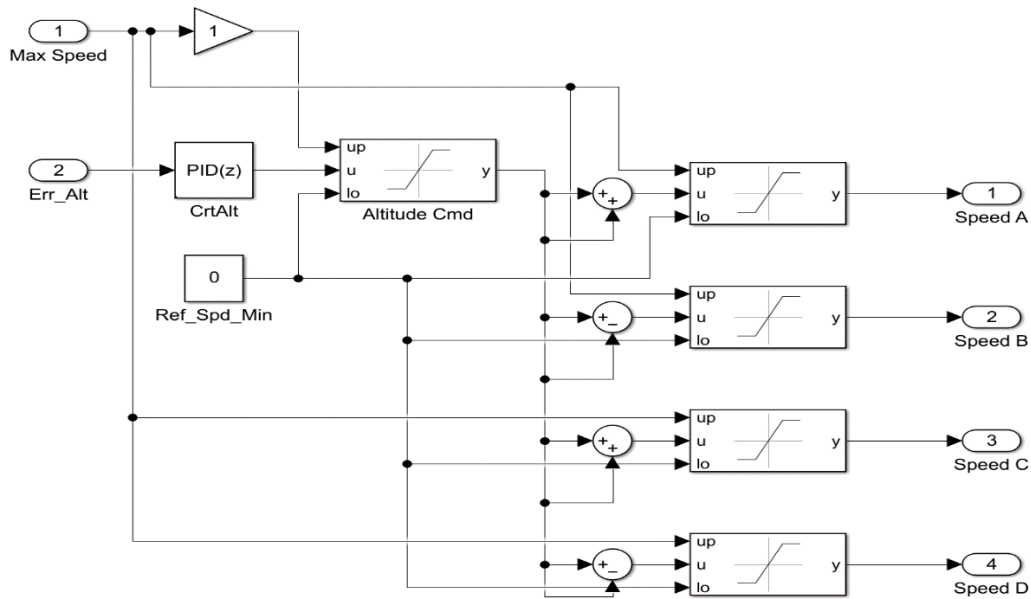


Fig. 1 - Altitude control system in quadcopter

A PID controller is used in this research due to its simplicity for implementation in a system. A fully tuned PID controller is needed for the quadcopter's rotors to spin according to the desired values. The values of K_p , K_i and K_d are used in this research due to its simplicity for implementation in a system. A fully tuned PID controller is needed for the quadcopter's rotors to spin according to the desired values. The values of and give it a better performance in its vertical motion. The PID controller improved the steady-state accuracy by decreasing the steady-state error in the system, thus improving the stability of the system. The maximum overshoot of the system is controlled by the PID controller, and the PID controller speeds up the slow response of an overdamped system.

The input is multiplied by a fixed amount in the Gain block. The gain value demonstrates the link between the input signal's and output signal's magnitudes at steady state. The constant value for the minimum speed is 0. The saturation dynamic block is connected to all rotors and the PID controller. This block generates an output signal that is the value of the input signal bound to the saturation values between the lower and upper limit values.

The sum input in this controller is put on every rotor's connector. The sum functions as an algebraic sum of the inputs, which follows the polarity of the inputs given. The end of the control subsystem is connected to the rotors of the quadcopter. Fig. 2 shows the controller subsystem connected to quadcopter models' rotors.

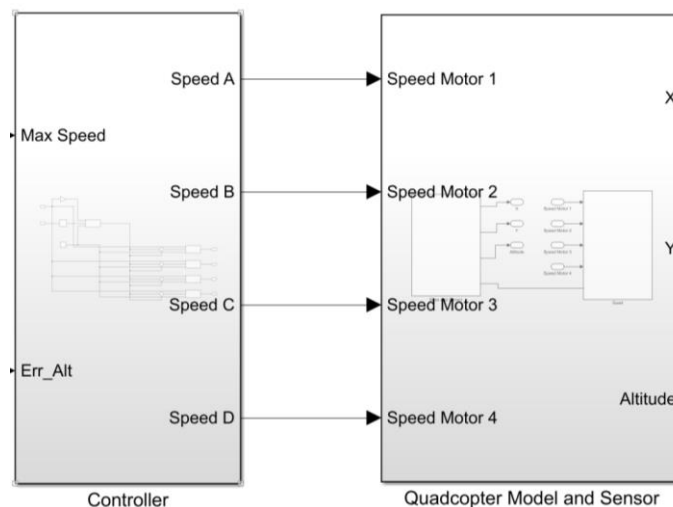


Fig. 2 - Controller connected to quadcopter's rotors

3. Quadcopter Simulator

Solidworks [8] is used to create the model necessary for the simulation, which is the first step in this investigation. The altitude controller simulation in MATLAB [11] will require a 3D model. To use MATLAB Simulink with a quadcopter model, the XML file must be exported. Figure 3 is a visual representation of the process of converting a Solidworks assembly file to an XML file. After obtaining the XML file, the "smimport" command must be used to import the file into the MATLAB Simulink software. Fig. 4 shows the modification to the block diagram and illustrates the arrangement of the block diagrams into the various subsystems. Fig. 5 depicts the tuning procedure and PID block diagram integration into the subsystem. To tune the PID controller, a PID tuning module was used.

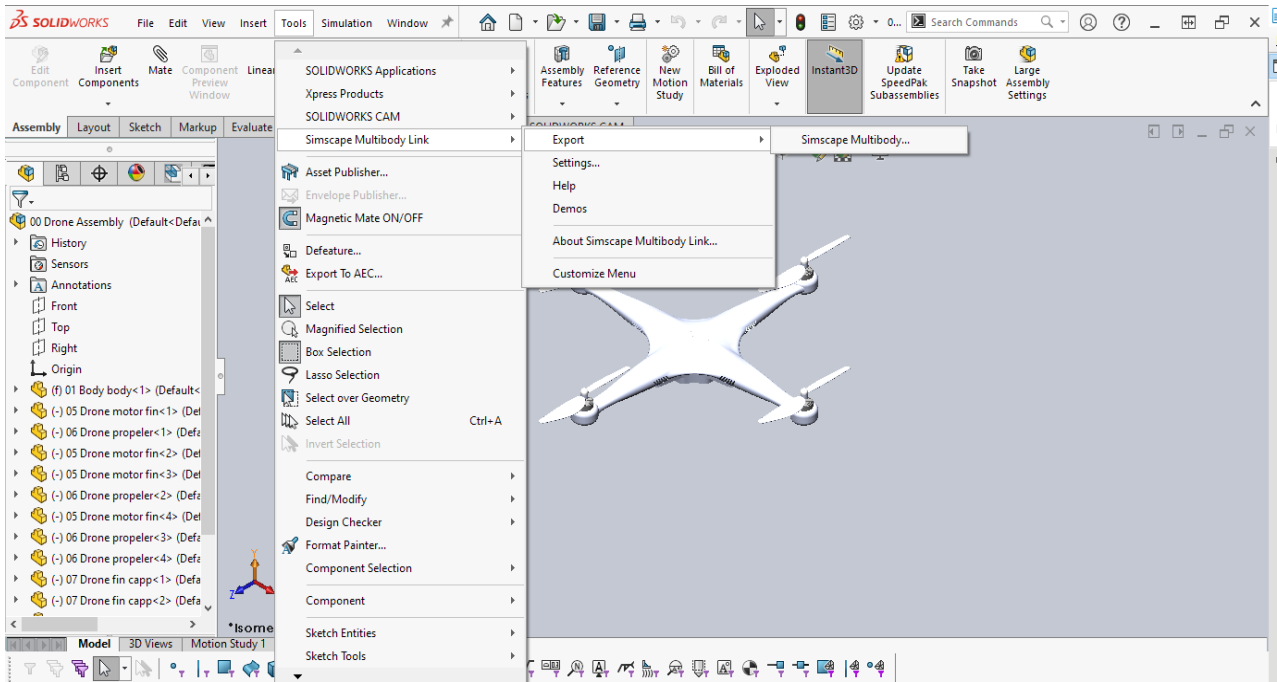


Fig. 3 - Exporting solid works assembly file into XML file

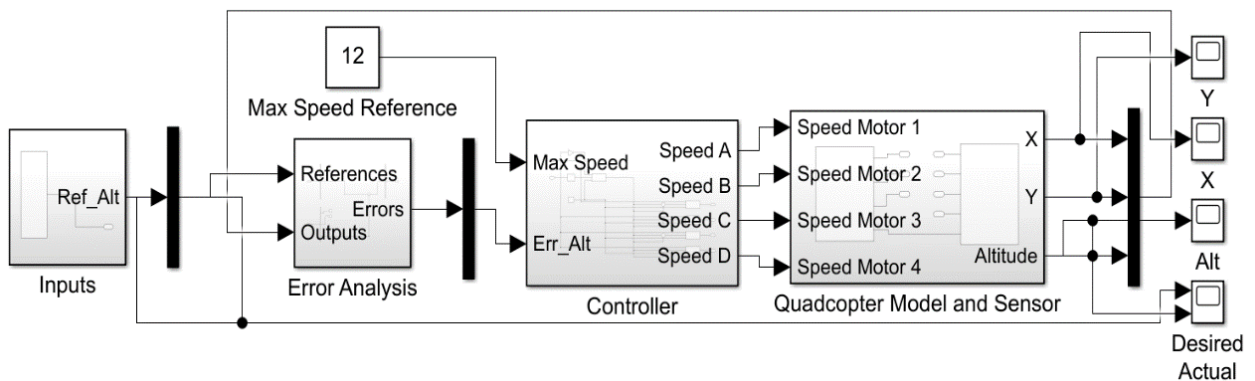


Fig. 4 - Adjusting block diagram in Simulink

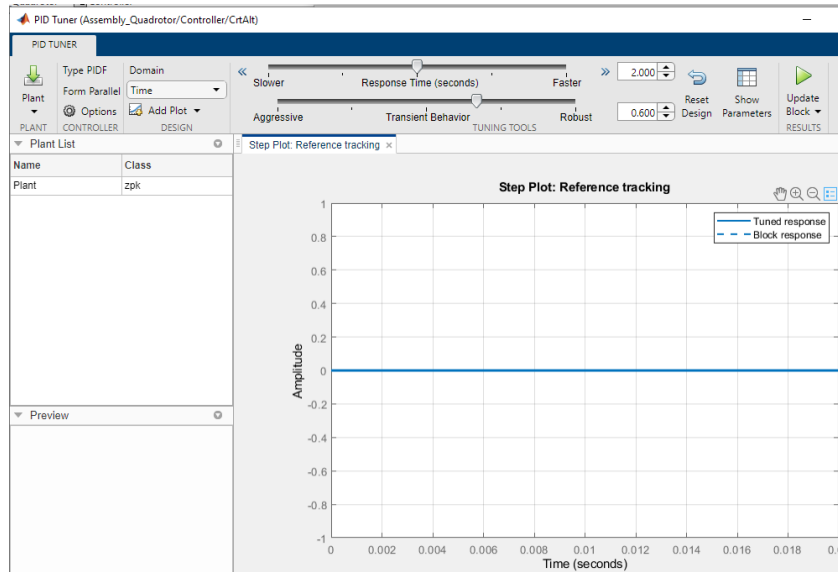


Fig. 5 - Adjusting PID controller with PID tuning module.

4. Simulation Evaluation

The performance of the controller was examined, and a graph of desired and actual altitude versus time is depicted in Fig. 6. As shown in Fig. 6, the controller was able to maintain the desired altitude within 4 s. Fig. 7 depicts the transient response of the control system. The transient response time took place between $t = 0$ s and $t = 3.91$ s. Again, it is evident that the transient response decays over time at approximately 3.91 s. Fig. 8 depicts the settling time achieved from this study. The obtained value for the settling time is 3.91 s. Fig. 9 depicts the peak response obtained in this research. At a duration of 0.46s, the peak amplitude value obtained is 1.07. Fig. 10 depicts the overshoot of frequency response. At a frequency of 4.15 rad/s, the peak amplitude is 1.12db and the overshoot percentage is 15.4%. In this research, the steady state error acquired is zero since the output of the actual signal and the desired signal are identical.

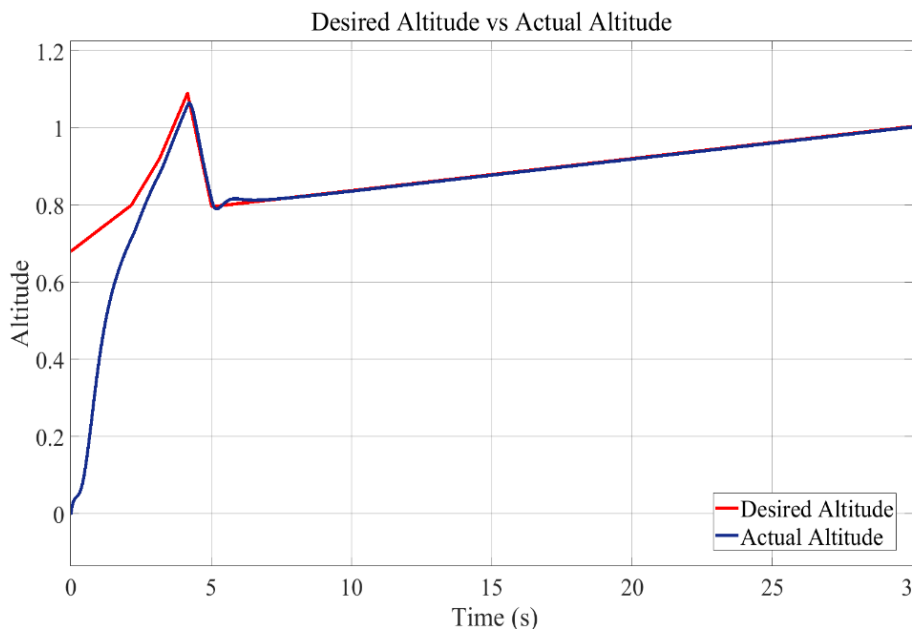


Fig.6 - Desired altitude vs actual altitude against time graph

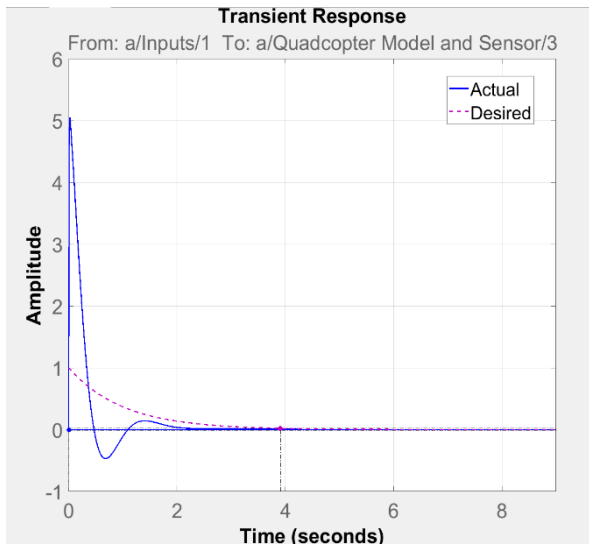


Fig. 7 - The transient response of the system

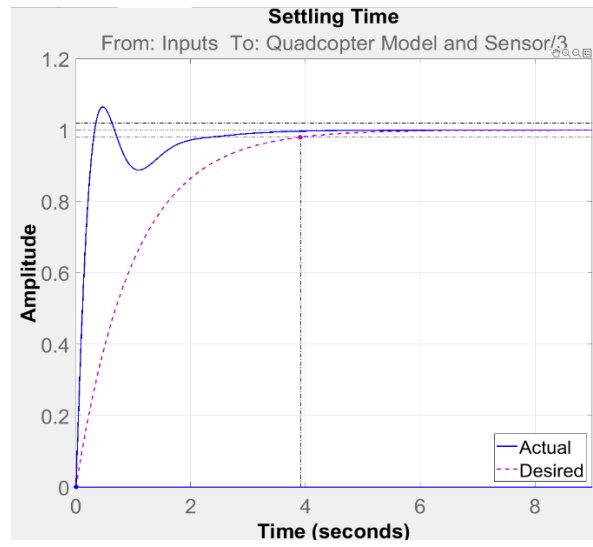


Fig. 8 - The settling time of the system

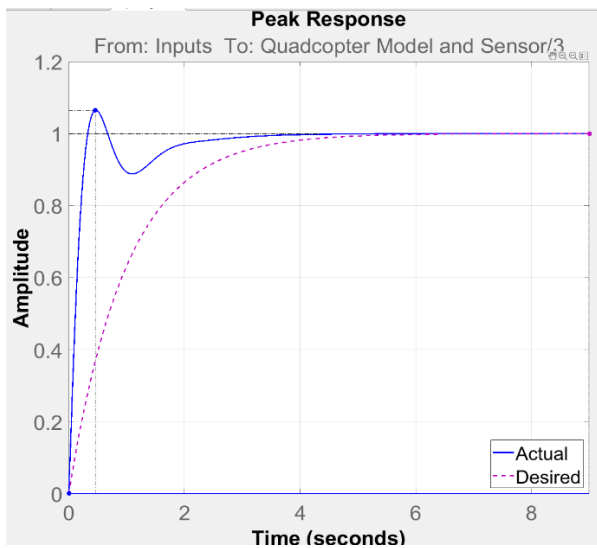


Fig. 9 - The peak response

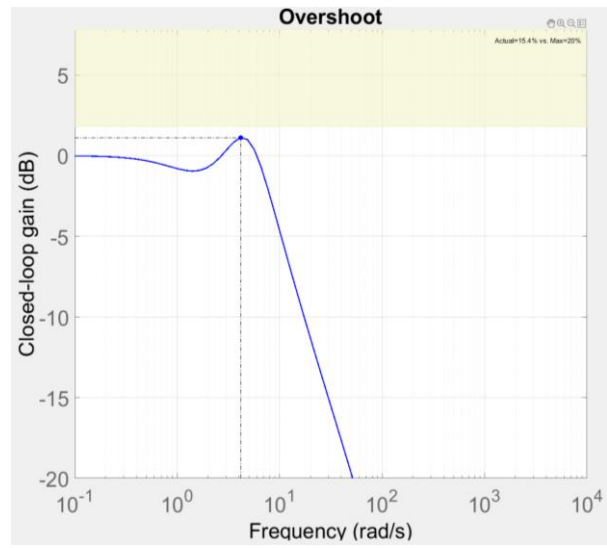


Fig. 10 - Overshoot produced the system

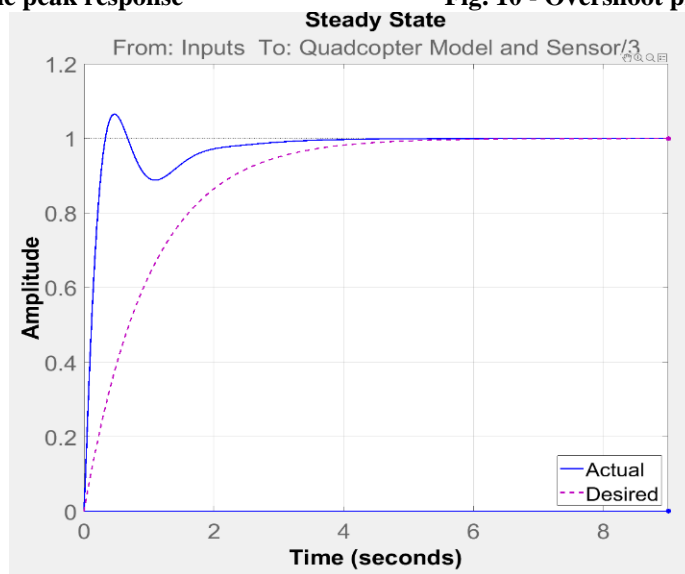


Fig. 11 - Steady state of the system

The tuned PID controller gives a better response time towards the input signal that has been sent to the rotors of the quadcopter. Table 1 below shows the parameter used in this research.

Table 1 - Parameter used in this research

Number	Parameter Name (unit)	Symbol	Value
1	Proportional	K_p	1.3038
2	Integral	K_i	1.4994
3	Derivative	K_d	0.28201
4	Filter coefficient	N	535.0236
5	Response Time (s)	-	0.31
6	Transient behavior	-	0.60
7	Rise time (s)	tr	0.238
8	Gain margin (rad/s)	-	346
9	Phase margin (rad/s)	ϕ_m	6.39
10	Close-loop stability	-	Stable
			0.68/0
			0.798/2.16
11	Amplitude/time (s)	A/t	0.92/3.16
			1.09/4.16
			0.796/5.02
			1.004/30
12	Gain	-	1

5. Conclusion

This study presents a virtual simulation of quadcopter altitude control. The performance of the constructed PID controller with K_p , K_i , and K_d gains tuning has been demonstrated and analyzed. Implementation of the tuned PID controller enables the quadcopter to achieve the desired altitude and hover in vertical motion. The simulator of the quadcopter's altitude controller may now be used to evaluate the performance of various controllers, such as the sliding mode controller, adaptive, and many others. This simulator can also be used to fine-tune the controller of other quadcopter models. In the future, additional controls such as yaw, pitch, and roll movement controllers can be added to the simulator.

Acknowledgement

The author would also like to thank the Faculty of Mechanical and Manufacturing Engineering, Universiti Tun Hussein Onn Malaysia for its support.

References

- [1] Ma'Sum, M. A., Jati, G., Arrofi, M. K., Wibowo, A., Mursanto, P., & Jatmiko, W. 2013, November
- [2] Yadav, D., Singh, G., Singh, B. M., Chauhan, H., Varshney, K. G., & Singh, G. 2018
- [3] Rizwan, R., Shehzad, M. N., & Awais, M. N. 2019
- [4] Xuan-Mung, Nguyen, and Sung-Kyung Hong. "Improved altitude control algorithm for quadcopter unmanned aerial vehicles." *Applied sciences* 9, no. 10 (2019): 2122.
- [5] Santos, M. F., V. S. Pereira, A. C. Ribeiro, M. F. Silva, M. J. do Carmo, V. F. Vidal, L. M. Honório, A. S. Cerqueira, and E. J. Oliveira. "Simulation and comparison between a linear and nonlinear technique applied to altitude control in quadcopters." In *2017 18th International Carpathian Control Conference (ICCC)*, pp. 234-239. IEEE, 2017.
- [6] Tanveer, M. Hassan, S. Faiz Ahmed, Desa Hazry, Faizan A. Warsi, and M. Kamran Joyo. "Stabilized controller design for attitude and altitude controlling of quad-rotor under disturbance and noisy conditions." *American Journal of Applied Sciences* 10, no. 8 (2013): 819.

- [7] Mukarram, Aniq, Usman Amin Fiaz, and Uzair Ijaz Khan. "Altitude control of a quadcopter." Department of Electrical Engineering, Pakistan Institute of Engineering and Applied Sciences, Islamabad, Pakistan (2015).
- [8] SolidWorks, Dassault Systèmes. "SolidWorks®."Version Solidworks1 (2005).
- [9] Matlab, S. (2012). Matlab. The MathWorks, Natick, MA.
- [10] Khanh Dang (2022). Simulate Quadrotor in Simulink with SimMechanics (<https://www.mathworks.com/matlabcentral/fileexchange/48052-simulate-quadrotor-in-simulink-with-simmechanics>), MATLAB Central File Exchange. Retrieved June 21, 2022.