Autonomous Quadcopter Control System Design Using LQG Controller to Perform Obstacle Avoidance

Purwadi Agus Darwito[1(✉)], Bima Dardaa Alfathrah[1], Hermawan Nugroho[2], and Totok Ruki Biyanto[1]

[1] Department of Engineering Physics, Sepuluh Nopember Institute of Technology, Surabaya, Indonesia
padarwito@gmail.com

[2] Department of Electrical and Electronic Engineering, University of Nottingham Malaysia Selangor, Selangor, Malaysia
Hermawan.Nugroho@nottingham.edu.my

Abstract. This research presents the implementation of the Linear Quadratic Gaussian (LQG) Controller and LiDAR sensors on the quadcopter used to avoid obstacles. The LQG controller is a combination of a Linear Quadratic Regulator (LQR) with a Kalman Filter as a State Estimator. The results of the estimate by Kalman Filter are compared against the actual value of the measurement by reviewing the Mean Square Error value. Linear position and angular position are variables that are controlled in the LQG control system to perform obstacle avoidance. The variations used in this study are the Q and R matrix values in the LQG controller design. Some of the tests that have been carried out include the open loop test, the closed loop test, the quadcopter test without an obstacle and the quadcopter test with an obstacle. The test results show that the quadcopter is able to follow the flight reference path and that the obstacle avoidance algorithm designed produces a quadcopter that is able to avoid obstacles. The test results show that the Q and R matrices on the Gain Regulator and Kalman filters can affect the quadcopter in carrying out avoidance.

Keywords: Quadcopter · Autonomous · LQG · Obstacle Avoidance

1 Introduction

Unmanned Aerial Vehicles (UAVs) having capability to trace over large areas and reaching environments that have difficult access. One type of UAV, quadcopter, also known as quadrotor, is a helicopter with four rotors. The quadcopter is controlled by regulating the angular velocity of the rotating rotor. The quadcopter is a distinctive design for small unmanned aerial vehicles (UAVs) due to its structure [1]. The quadcopter has many advantages such as agility, compact shape, low cost, ease of manufacturing compared to other UAV systems. Quadcopters have been tested and used in various technologies such as obstacle avoidance, air formation, path planning, and object tracking [2].

S. Sugiman et al. (Eds.): MIMSE-M-E 2022, AER 216, pp. 284–298, 2023.
https://doi.org/10.2991/978-94-6463-078-7_27

The surrounding environment becomes one of the factors affecting the operation of the UAV, the problems that exist in the environment can be weather conditions, flight altitude, and obstacles on the track. Safety factors on UAVs can be threatened by collisions with other UAVs, aircraft, birds, or other obstacles. The problem can be solved with a fully autonomous UAV to eliminate human error and be able to identify and avoid obstacles [3].

An appropriate control system is required to achieve the desired performance. Many controlling methods have been proposed for quadcopters. It aims to find a controlling strategy that allows the quadcopter state to approach time.

changing reference paths and be able to meet the objectives of obstacle avoidance. In previous studies it has been shown that linear control techniques with linearization of dynamics are used to control quadcopter hovering. However, wider flight coverage and better performance can be achieved using nonlinear control techniques that consider the more common forms of vehicle dynamics across all flight zones [4].

Many control system designs have been applied to autonomous quadcopters. Based on previous research, the Quadcopter is controlled by several types of control systems such as PID, LQR, LQG, and others [5] – [9]. The control system applied to the autonomous quadcopter has applications in various quadcopter flight missions. Based on comparisons in other studies, this study proposes the design of a quadcopter control system using the LQG Controller. This algorithm is designed for systems with Gaussian interference [10]. The scope of this study is to design a Linear Quadratic Gaussian control system on an autonomous quadcopter that is able to avoid obstacles and follow the route. The quadcopter will move from the starting point to the next point of travel. Along the way, the quadcopter will encounter predetermined obstacles, these obstacles will be detected by LiDAR sensors and result in measurements of the relative distance of the quadcopter to the obstacles in front of it. Simulation is carried out using MATLAB R2022b software.

2 Methodology

2.1 Quadcopter Data

The quadcopter specification data is derived from calculations on prototypes in the Instrumentation, Control and Optimization Laboratory, ITS Engineering Physics which includes arm length, inertial coefficient, thrust coefficient, quadcopter mass, and so on, as shown in Table 1.

2.2 Mathematical Model of Quadcopter

Quadcopter modeling is obtained from the differential equations of quadcopter dynamics which are rewritten in the form of state space equations. The physical system of the quadcopter has different inputs and outputs, such input- output relationships can be represented in the form of a state- space. First, the analysis used for kinematics is the use of three-dimensional cartesian diagram frames (X, Y, Z). The cartesian diagram frame is divided into two, namely the rigid or immovable earth frame, as well as the quadcopter

Table 1. Table captions should be placed above the tables.

Parameter	Value	Unit
Arm Length (l)	0.23	meters
Inertial Coefficient (I_{xx})	0.01557	Kg. m^2
Inertial Coefficient (I_{yy})	0.01557	Kg. m^2
Inertial Coefficient (I_{zz})	0.02098	Kg. m^2
Mass (m)	1.3	Kg
Coefficient thrust (b)	9.56×10^{-6}	N. s^2
Gravity (g)	9.81	m/s^2

body frame that moves rotationally and translationally. The rotation equation in [11] the quadcopter kinematics analysis for all axes is the product of each of its axis rotation matrices as shown in Eq. 1.

Displayed equations are centered and set on a separate line.

$$R_{xyz} = \begin{bmatrix} c_\psi c_\theta & -s_\psi c_\varphi + c_\psi s_\theta s_\varphi & s_\psi s_\varphi + c_\varphi s_\theta s_\psi \\ s_\psi c_\theta & c_\psi c_\varphi + s_\psi s_\theta s_\varphi & -c_\psi s_\varphi + s_\psi s_\theta c_\varphi \\ -s_\theta & c_\theta s_\varphi & c_\theta c_\varphi \end{bmatrix} \tag{1}$$

R is a rotational matrix to change the attitude and angular speed of the quadcopter on the body frame to the inertial frame, c and s are representations of the sin and cos functions in the matrix. The mathematical model of the quadcopter used in this study has been developed previously by [12]. Based on the laws of aerodynamics analyzed on the quadcopter, Eq. 2 is the result of linearization of the quadcopter dynamics model used for linear and angular acceleration in the quadcopter which is presented in the form of state-space as follows.

$$\dot{x} = \begin{cases} \ddot{\phi} = \frac{U_2}{I_{xx}} \\ \ddot{\theta} = \frac{U_3}{I_{yy}} \\ \ddot{\Psi} = \frac{U_4}{I_{xx}} \\ \ddot{x} = \frac{U_1}{m} U_x \\ \ddot{y} = \frac{U_1}{m} U_y \\ \ddot{z} = \frac{U_1}{m} - g \end{cases} \tag{2}$$

$$
\begin{aligned}
U_1 &= b\left(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2\right) \\
U_2 &= bl\left(\Omega_3^2 - \Omega_1^2\right) \\
U_3 &= bl\left(\Omega_4^2 - \Omega_2^2\right) \\
U_4 &= d\left(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2\right) \\
U_x &= \cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi \\
U_y &= \cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi \\
U_z &= g - \frac{U_1}{m} \cos\phi \cos\psi
\end{aligned}
\tag{3}
$$

It is known that the quadcopter has several movements such as translation and rotation, also has an attitude that represents the orientation of the quadcopter and the linear position on each axis. To minimize calculations, state reduction techniques are also used to facilitate mathematical calculations. [12] The state vector used in this study is written in Eq. 4 as follows.

$$
\dot{x} = \left[\dot{\phi}\ \dot{\theta}\ \dot{\psi}\ \dot{p}\ \dot{q}\ \dot{r}\ \dot{u}\ \dot{v}\ \dot{w}\ \dot{x}\ \dot{y}\ \dot{z}\right]^T \in \mathbb{R} \tag{4}
$$

With this technique the state vector on Eq. 4 will be divided into 3 parts, namely the attitude control represented by the vector $(\phi, \dot{\varphi}, \theta, \dot{\theta}, \psi, \dot{\psi})^T$, the position control represented by the vector $(x, \dot{x}, y, \dot{y})^T$, and the altitude control represented by the vector $(z, \dot{z})^T$ as in the following equation:

$$
x = \begin{bmatrix} \phi \\ \dot{\phi} \\ \theta \\ \dot{\theta} \\ \psi \\ \dot{\psi} \\ x \\ \dot{x} \\ y \\ \dot{y} \\ z \\ \dot{z} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{bmatrix} = \left(\text{ Orientation, Position}\right)^T \tag{5}
$$

The state-space equations for the attitude, position, and altitude control systems are expressed in Eqs. (6–8) as follows.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} 0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1 \\ 0\,0\,0\,0\,0\,0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{x_{xx}} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix} \tag{6}$$

$$\begin{bmatrix} \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \end{bmatrix} = \begin{bmatrix} 0\,1\,0\,0 \\ 0\,0\,0\,0 \\ 0\,0\,1\,0 \\ 0\,0\,0\,0 \end{bmatrix} \begin{bmatrix} x_7 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{U_1}{m} & 0 \\ 0 & 0 \\ 0 & \frac{U_1}{m} \end{bmatrix} \begin{bmatrix} U_x \\ U_y \end{bmatrix} \tag{7}$$

$$\begin{bmatrix} \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} 0\,1 \\ 0\,0 \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} [U_z] \tag{8}$$

2.3 LQG Control System Design

LQG Controller has stages in designing this control system, namely designing the Gain Regulator, and designing the Kalman filter on controlling the attitude and position (including altitude) of the quadcopter. There is a sensor used in this design is the Inertial Navigation Sensor (INS) sensor on the Simulink MATLAB. Figure 1 describes the block diagram designed in this study. Block diagrams are created by referring to block diagrams that have previously been researched by making modifications as a form of adjustment to the system design. The block diagram illustrates how the control system works [13, 14] and the position of the quadcopter.

The inputs in the system are Set Point Coordinates in the form of x, y, z and Obstacle Coordinates in the form of x, y, z. Set Point Coordinates is the travel point of the quadcopter, this results in the quadcopter will move from the starting point to the next

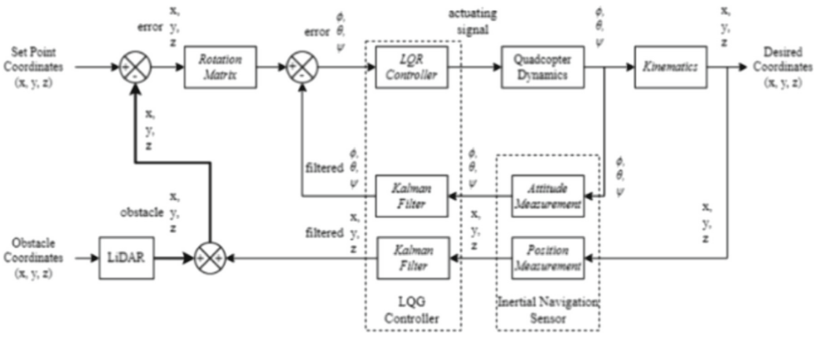

Fig. 1. Quadcopter block diagram with LQG controller.

travel point until the last point. Obstacle Coordinates are the coordinates of the position the obstacle is located which is measured by a LiDAR sensor. The INS sensor is a sensor that measures the linear position of the quadcopter in the x, y, z spaces and the angular position of the inner quadcopter φ, θ, ψ

- Gain Regulator

The Q and R matrices can affect the obtained gain value. The acquisition of the Q and R matrices was carried out using the trial-and-error method used in determining the state (Q) and control (R) weighting matrices. Matriks Q and R are positive and symmetrical semidefinites. Thus, the size of the Q and R matrices adjusts from each model of attitude, position, and altitude. Here is the design of the Q and R matrices used.

$$\begin{aligned} Q_{attitude} &= \begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&1&0\\ 0&0&0&0&0&1 \end{bmatrix}, R_{attitude} = \begin{bmatrix} 1&0&0\\ 0&1&0\\ 0&0&1 \end{bmatrix} \\ Q_{position} &= \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix}, R_{position} = \begin{bmatrix} 1&0\\ 0&1 \end{bmatrix} \\ Q_{altitude} &= \begin{bmatrix} 1&0\\ 0&1 \end{bmatrix}, R_{altitude} = [1] \end{aligned} \tag{9}$$

The variations are set as a comparison in this design. Variations in the values of the matrix Q and R are obtained from the results of the research carried out by [12], the value of the matrix is expressed in Eq. 10.

$$\begin{aligned} Q_{attitude} &= \begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&1&0\\ 0&0&0&0&0&1 \end{bmatrix} R_{attitude} = \begin{bmatrix} 0.3&0&0\\ 0&0.3&0\\ 0&0&0.3 \end{bmatrix} \\ Q_{position} &= \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix}, R_{position} = \begin{bmatrix} 0.5&0\\ 0&0.5 \end{bmatrix} \\ Q_{altitude} &= \begin{bmatrix} 1&0\\ 0&1 \end{bmatrix}, R_{altitude} = [1] \end{aligned} \tag{10}$$

Table 2. Variations of the Matrix Q and R Kalman filter.

Q	R
1	10
1	100
0.1	10
0.1	100

- Kalman Filter Design

In this study, the Kalman filter concept was used as an estimator on the state variables of linear position (x y z) and angular position ($\varphi\ \theta\ \Psi$). This is done to observe the ability of Kalman Filter to estimate state variables that have measurement noise that represents the use of physical sensors in the prototype. The Q and R matrices used were obtained from research [15] conducted by several modifications, here is a list of variations in the values of the Q and R matrices to be used for the Kalman Filter design presented in Table 2.

2.4 Obstacle Avoidance System Design

The algorithm of the obstacle avoidance system designed is adapted from the research [16] carried out by making several modifications to match the system designed. The obstacle avoidance algorithm used in this study is based on the detection of obstacles using LiDAR sensors with cloud output points. The flow of the data processing process using LiDAR sensors with cloud output points. The flow of the data processing process using LiDAR sensors is shown in the flow chart below:

3 Results and Discussion

3.1 Open Loop and Closed Loop Analysis

The open loop test aims to observe the response of the quadcopter when the system is not given a controller. Observations are made by observing the quadcopter response when given input. Open loop testing is performed on translational motion and quadcopter rotational motion. The open loop test of translational motion is represented by testing on the X-axis only, while the test of rotational motion is represented by the angle of yaw only. The test was performed by running the MATLAB program as shown in Fig. 3. In translational motion the X-axis is given $Ux = 1$ as input and rotational motion at the yaw angle is given as $U4 = 1$ input. Figure 3 shows the results of open loop testing on quadcopter translational motion and Fig. 4 shows the results of open loop testing on quadcopter rotational motion.

Based on Figs. 3 and 4, the response results in the open loop test show the same graphic characteristics, namely exponential in both translational movements and rotational movements. This is closely related to the relationship of the thrust force with the

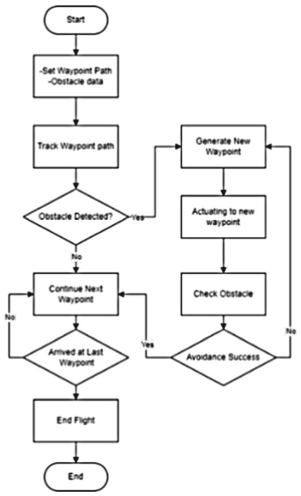

Fig. 2. Obstacle Avoidance System flow chart.

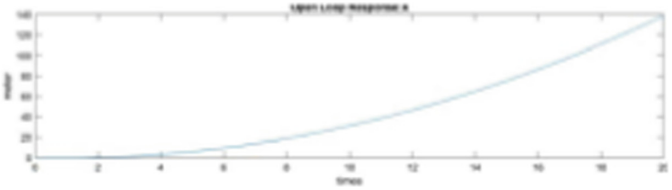

Fig. 3. Quadcopter Translational Motion Open Loop Test.

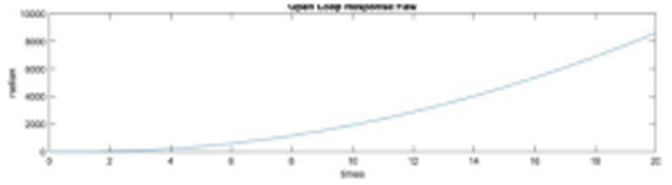

Fig. 4. Quadcopter Rotational Motion Open Loop Test.

square of the rotating speed of the quadcopter motor. The existence of the quadcopter will be continuously moving by being given an actuation signal input in both rotational and translational movements. This is because in open loop testing the system has no feedback or feedback.

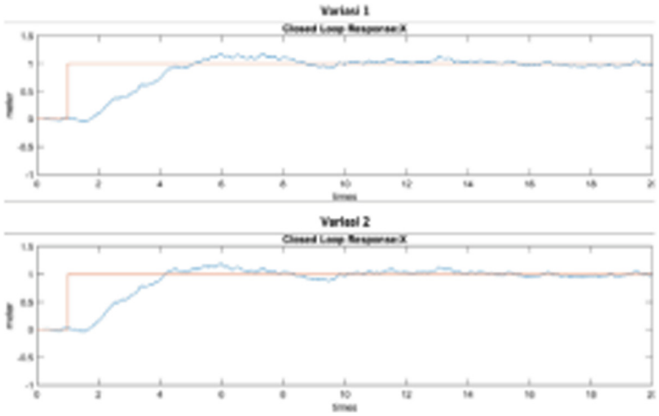

Fig. 5. Quadcopter Translational Motion Closed Loop Test

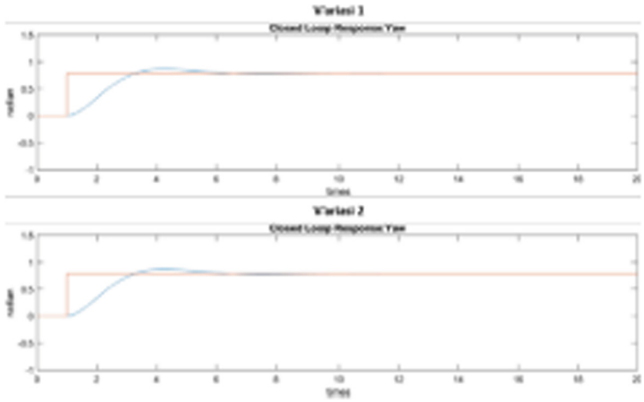

Fig. 6. Quadcopter Rotational Motion Closed Loop Test.

Figure 5 shows the results of closed loop testing on quadcopter translational motion and Fig. 6 shows the results of closed loop testing on quadcopter rotational motion.

Based on Figures 5 and 6, the response results in the closed loop test show the ability of the LQG control system to reach the set point. The red line indicates the set point, and the blue line indicates the process variable or controlled variable. To analyze the performance of the LQG controller in controlling the quadcopter, it is necessary to analyze the performance of the controller. Control performance analysis is carried out by calculating the rise time, settling time and maximum overshoot values of each state variable. The following is the result of the analysis of the performance of the controller as shown in Table 3.

Table 3. Control System Performance Analysis.

Translation Movement			
Variasi	*Rise Time (s)*	*Settling Time (s)*	*Max. Overshoot (%)*
I	2.3049	19.9735	17.1251
II	2.1315	19.9733	17.1142
Rotation Movement			
I	1.5341	5.9981	10.3981
II	1.5323	5.9798	10.2958

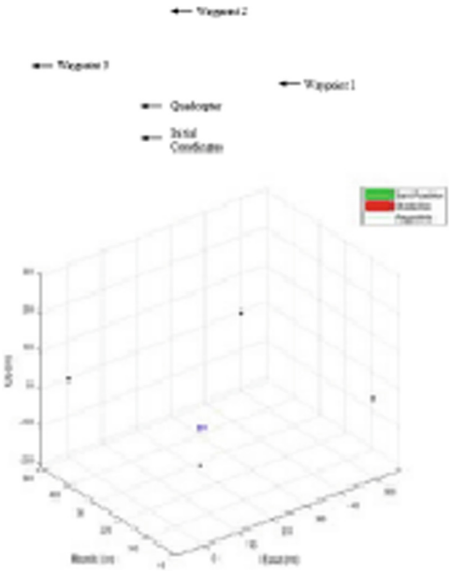

Fig. 7. Quadcopter Rotational Motion Closed Loop Test without any obstacles.

3.2 Quadcopter Test Without Obstacles

The simulation scenario is determined by running a quadcopter that follows the point to the next point. There are 4 points that the quadcopter will reach including the starting point. The quadcopter initially is at position (0.0,0), then moves towards the coordinate points (0.50, -10), (50.50, -10), and (50,0, -10) respectively. In this simulation test, the quadcopter moves without any obstruction along its trajectory and will be reviewed for the average error value resulting from the position of the quadcopter against the reference trajectory with variations in the Q and R matrices in the LQG Controller design. The following is an illustration of a quadcopter testing scenario without any obstacles as shown in Fig. 7.

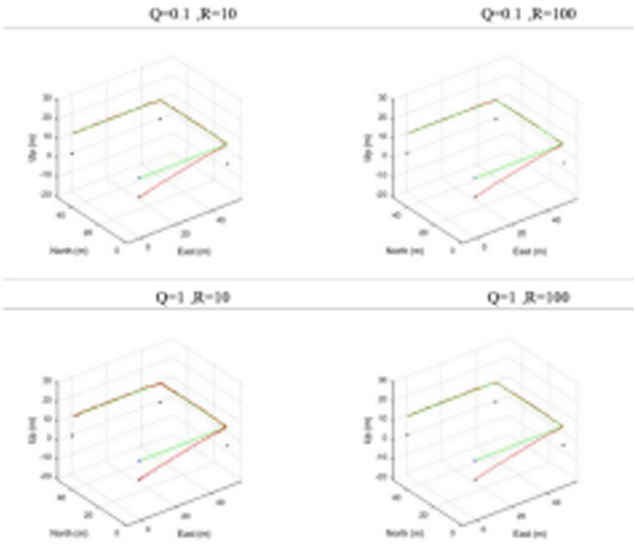

Fig. 8. Quadcopter Testing Results Without Obstacles.

The movement of the quadcopter from point to point will be used on all quadcopter tests. Then, the quadcopter test results without obstacles are taken the mean square error value of each state variable. The determination of the value of the mean square error is used with reference to the study [17]. The combination value of the matrix Q and R which has the smallest mean square error value will be used in quadcopter testing with obstacles. The following are the results of quadcopter testing without any obstacles as shown in Fig. 8.

The results showed that the quadcopter was able to reach all points of travel, but the translational movement of the quadcopter was still not good because it was influenced by noise measurement. Quadcopter testing without obstacles was carried out in the presence of variations in the Q and R covariance error matrices on the Kalman filter. The results of testing each variation were compared by calculating the Mean Square Error (MSE) value of each state variable, namely the position of the quadcopter. A comparison of such results is presented in Table 4.

From the results of quadcopter testing with obstacles, the results of the Mean Square Error calculation were also obtained on each of the actual variables to the estimated value. The Mean Square Error value represents Kalman Filter's ability to overcome noise measurement in sensor measurements. These results are shown in Table 5.

3.3 Abbreviations and Acronyms

In this activity, the quadcopter is simulated following the same travel point, that is, the quadcopter moves from point to point according to the order described earlier. However, in this test, there are differences in flight conditions, namely with additional obstacles or obstacles. The obstacles used in this simulation are 2 which are arranged in series with

Table 4. MSE Calculation Results Estimates State Variables Against Actual Values.

Mean Square Error Quadcopter Position Estimation				
Variabel	*Q* dan R Matrix on *Kalman filter Variation*			
	Q = 1, R = 10	Q = 1, R = 100	Q = 0.1, R = 100	Q = 0.1, R = 10
X	1.488 E-01	3.011E-01	9.798E-06	2.130E-01
Y	1.811 E-01	4.841E-01	1.630E-05	2.721E-01
Z	1.538 E-01	2.218E-01	5.500E-03	2.106E-01

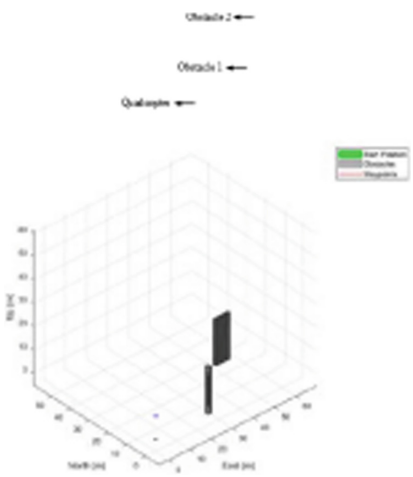

Fig. 9. Quadcopter Testing Scenario with an Obstacles.

variations in shape, namely the obstacles in the shape of the cylinder and the shape of the beam. The test scenario of this simulation is shown as follows.

The obstacle avoidance algorithm corresponding to Fig. 2 of the quadcopter will avoid such obstacle. In this test, the LQG controller is a control strategy used in the quadcopter given variations in the form of matrix values Q and R for the Gain Regulator in the design of the LQG controller as a representation of the weighting matrix against state variables and input states. The following are the results of quadcopter testing with obstacles.

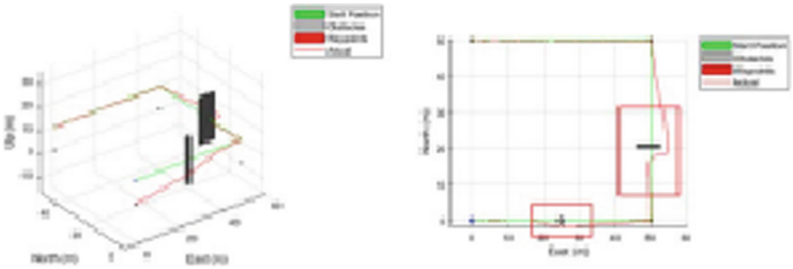

Fig. 10. Quadcopter Testing Results with Obstacle.

Table 5. MSE Calculation Results Estimates State Variables Against Actual Values with obstacles.

Mean Square Error Linear dan Angular Position Estimation	
Kalman filter: Q = 0.1, R = 100	
Variable	Value
X	4.146.E-01
Y	7.540.E-01
Z	2.421.E-01
Roll	2.579E-05
Pitch	4.129E-07
Yaw	1.100E-03

4 Conclusion

The design of the LQG control system is carried out by modeling the dynamics of the quadcopter, changing into a state-space form, calculating the Gain Regulator, and designing the Kalman Filter design. The optimal Gain Regulator value is obtained in Variation 2 of the matrix values Q and R, while the optimal Gain Kalman value is obtained in Variation 3 of the Q and R matrix values. The design of the LQG control system is able to achieve the set points given to translational movements and rotational movements. The designed obstacle avoidance control system and algorithm are used in testing without obstacles and with obstacles. The results showed that the mean square error performance of each linear position variable (x, y, z) was valued at 9,798E-06, 1,630E-05, 5,500E-03 using variations of 3 Q and R matrices on the Kalman Filter. The obstacle avoidance algorithm is used in quadcopter testing with obstacles, the results show that the quadcopter is able to avoid obstacles.

Acknowledgement. The authors gratefully acknowledge the contributions of the Instrumentation, Control, and Optimization Laboratory of Sepuluh Nopember Institute of Technology (ITS)'s team

and parties for supporting any software and computer requirements to meet the needs of this research.

References

1. Luukkonen, T.: Modeling and control of a quadcopter. Independent research project in applied mathematics, 22(7) (2011).
2. Balestrieri, E., Daponte, P., de Vito, L., Picariello, F., and Tudosa, I.: Sensors and Measurements for UAV Safety: An Overview. Sensors 21(24), p. 8253.
3. Balestrieri, E., Daponte, P., de Vito, L., and Lamonaca, F.: Sensors and Measurements for Unmanned Systems: An Overview. Sensors 21(4), p. 1518.
4. Nguyen, N. P., Mung, N. X., Thanh, H. L. N. N., Huynh, T. T., Lam, N. T., and Hong, S. K.: Adaptive Sliding Mode Control for Attitude and Altitude System of a Quadcopter UAV via Neural Network. IEEE Access 9 (2021).
5. Bouabdallah, S., Noth, A., and Siegwart, R.: PID vs LQ control techniques applied to an indoor micro quadrotor. in 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2004, vol. 3 (2004).
6. Cowling, I. D., Yakimenko, O. A., Whidborne, J. F., and Cooke, A. K.: A prototype of an autonomous controller for a quadrotor UAV (2007).
7. Saraf, P., Gupta, M., and Parimi, A. M.: A Comparative Study between a Classical and Optimal Controller for a Quadrotor (2020).
8. Minh L. D., and Ha, C.: Modeling and control of quadrotor MAV using vision-based measurement (2010).
9. Argentim, L. M., Rezende, W. C., Santos, P. E., and Aguiar, R. A.: PID, LQR and LQR-PID on a quadcopter platform (2013).
10. Zulu, A., and John, S.: A Review of Control Algorithms for Autonomous Quadrotors. Open Journal of Applied Sciences 4(14) (2014).
11. Eatemadi, M.: Mathematical Dynamics, Kinematics Modeling and PID Equation Controller of QuadCopter. International Journal of Applied Operational Research 7(1) (2017).
12. Okyere, E., Bousbaine, A., Poyi, G. T., Joseph, A. K., and Andrade, J. M.: LQR controller design for quad-rotor helicopters. The Journal of Engineering 2019(17), pp. 4003–4007, (2019).
13. Ha, L. N. N. T., and Hong, S. K.: Robust dynamic sliding mode control-based PID–super twisting algorithm and disturbance observer for second-order nonlinear systems: Application to UAVs. Electronics (Switzerland) 8(7), (2019).
14. Wang, P., Man, Z., Cao, Z., Zheng, J., and Zhao, Y.: Dynamics modelling and linear control of quadcopter. in 2016 International Conference on Advanced Mechatronic Systems (ICAMechS), pp. 498–503, Nov (2016).
15. Ma'arif, A., Iswanto, I., Nuryono, A. A., and Alfian, R. I.: Kalman Filter for Noise Reducer on Sensor Readings. Signal and Image Processing Letters, 1(2), pp. 11–22 (2019).
16. Guo, J., Liang, C., Wang, K., Sang, B., and Wu, Y.: Three-Dimensional Autonomous Obstacle Avoidance Algorithm for UAV Based on Circular Arc Trajectory. International Journal of Aerospace Engineering (2021).
17. Brito, V., Brito, A., Palma, L. B., and Gil, P.: Quadcopter control approaches and performance analysis. in ICINCO 2018 - Proceedings of the 15th International Conference on Informatics in: Control, Automation and Robotics, vol. 1 (2018).