Available online at www.sciencedirect.com

ScienceDirect

Franklin Institute 00 (2022) 1–8

Franklin Institute

Linear Quadratic Integral Differential Game applied to the Real-time Control of a Quadrotor Experimental setup

hasa BniAsad

address

h BniAsad

Abstract

The accurate attitude control of a quadrotor is necessary, especially when facing disturbance. Moreover, all the flight states of the quadrotor are not measured in practice. In this study, a linear quadratic Gaussian with integral action based on the differential game theory is implemented on the quadrotor experimental setup. A continuous state-space model of the setup is derived using the linearization of nonlinear equations of motion, and its parameters are identified with the experimental results. Next, the attitude control commands of the quadrotor are derived based on two players; one finds the best attitude control command, and the other creates the disturbance by mini-maximizing a quadratic criterion, defined as the sum of outputs plus the weighted control effort and disturbance. The performance of the proposed structure is investigated in level flight and compared to the linear quadratic regulator controller. Results demonstrate that the proposed approach has an excellent performance in dissipating the disturbances.

Keywords:
Linear Quadratic Gaussian, Differential Game, Quadrotor, State Estimation, 3DoF Experimental setup, Optimal Control, Robust Control.

1. Introduction

A quadrotor is a type of helicopter with four rotors that plays a significant role in today's society [1], including research, military, imaging, recreation, and agriculture. The performance of the quadrotor relies on the control system, including attitude, altitude, and position subsystems. In the attitude control of the quadrotor, it is vital to maintain the attitude outputs at the desired level using control commands, such as the rotational speed of the rotors [2], when disturbances occur suddenly. Therefore, much research is being conducted on the automatic control of the attitudes' quadrotor in facing the disturbance. In [3, 4], a Proportional Integral Derivative (PID) controller is used to regulate the quadrotor attitude. However, the control objectives have not been effectively achieved with this controller when the disturbance occurs. To solve this problem, the model-based approaches [5, 6] are utilized for controller design. These controllers work based on information from the quadrotor's attitude model and disturbance to produce the best control command.

Email address: `email address` (h BniAsad)

2. Problem Statement

Here, a nonlinear dynamic is presented for the setup of the quadrotor, as illustrated in figure 1. The quadrotor is free to rotate about its roll, pitch, and yaw axes. The acceleration and the angular velocities along three orthogonal axes are measured using the low-cost Inertial Measurement Unit (IMU). These noisy measurements are utilized in a nonlinear filter for the estimation of the quadrotor states, including the Euler angles and angular velocities. These estimated states are compensated in the structure of the LQIG-DG controller to stabilize the quadrotor setup. The block diagram of the controller structure is illustrated in Fig. 2.

Figure 1. 3DoF setup of the quadrotor.

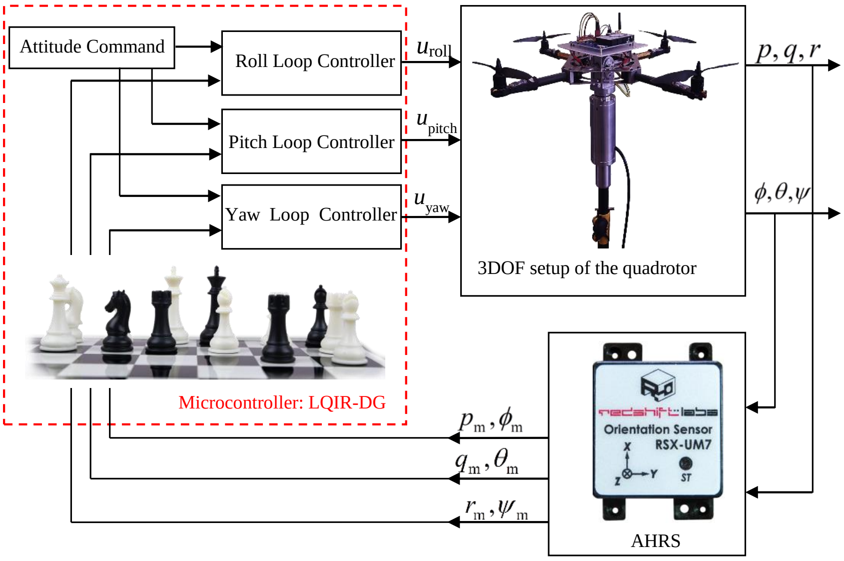

Figure 2. Block diagram of the LQIG-DG controller structure.

3. Modeling of the Quadrotor Setup

Here, the model of the three-degree-of-freedom setup of the quadrotor is presented in detail. For this purpose, first, the configuration of the quadrotor is denoted. Then, the nonlinear model of the attitude dynamics is derived from denoting the state-space form. Finally, the nonlinear model is linearized to utilize for control purposes.

3.1. Configuration of the Quadrotor

Figure 3 denotes the quadrotor schematic. Each rotor has an angular velocity, Ω_r, rotating about the z_B axis in the body coordinate system. Rotors 1 and 3 rotate counterclockwise, while rotors 2 and 4 rotate clockwise to cancel yawing moment.

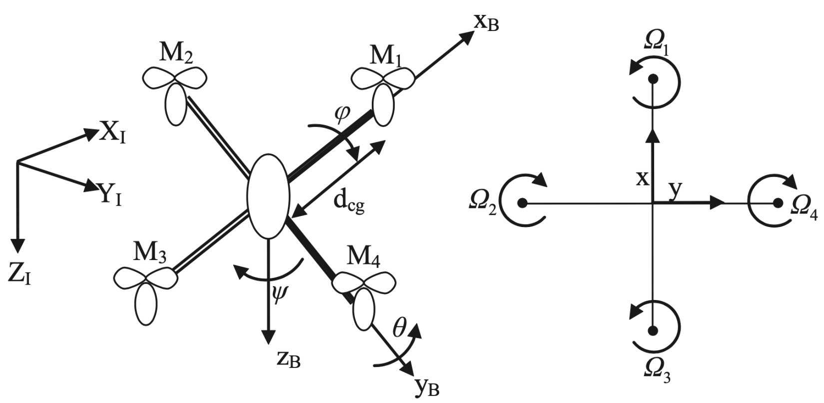

Figure 3. Configuration of the quadrotor.

3.2. Dynamic Model

The quadrotor kinetic model, derived using the Newton-Euler method, is stated as [7, 8]

$$\dot{p} = \frac{\mathrm{I_{yy}} - \mathrm{I_{zz}}}{\mathrm{I_{xx}}} qr + q \frac{\mathrm{I_{rotor}}}{\mathrm{I_{xx}}} \Omega_r + \frac{u_{\mathrm{roll}}}{\mathrm{I_{xx}}} + \frac{d_{\mathrm{roll}}}{\mathrm{I_{xx}}} \tag{1}$$

$$\dot{q} = \frac{\mathrm{I_{zz}} - \mathrm{I_{zz}}}{\mathrm{I_{yy}}} rp + p \frac{\mathrm{I_{rotor}}}{\mathrm{I_{xx}}} \Omega_r + \frac{u_{\mathrm{pitch}}}{\mathrm{I_{yy}}} + \frac{d_{\mathrm{pitch}}}{\mathrm{I_{yy}}} \tag{2}$$

$$\dot{r} = \frac{\mathrm{I_{xx}} - \mathrm{I_{yy}}}{\mathrm{I_{zz}}} pq + \frac{u_{\mathrm{yaw}}}{\mathrm{I_{zz}}} + \frac{d_{\mathrm{yaw}}}{\mathrm{I_{zz}}} \tag{3}$$

where (p, q, r) are the angular velocities. d_{roll}, d_{pitch}, and d_{yaw} are the disturbances generated in x_B, y_B, and z_B, respectively. Moreover, $\mathrm{I_{xx}}$, $\mathrm{I_{yy}}$, and $\mathrm{I_{zz}}$ are the principal moment of inertia, and $\mathrm{I_{rotor}}$ is a rotor inertia about its axis. The relation between the angular body rates and the Euler angles rates are obtained as

$$\dot{\phi} = p + (q \sin(\phi) + r \cos(\phi)) \tan(\theta) \tag{4}$$

$$\dot{\theta} = q \cos(\phi) - r \sin(\phi) \tag{5}$$

$$\dot{\psi} = (q \sin(\phi) + r \cos(\phi)) / \cos(\theta) \tag{6}$$

where (ϕ, θ, ψ) are roll, pitch, and yaw angles. Moreover, Ω_r, called the overall residual rotor angular velocity, is computed as

$$\Omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4 \tag{7}$$

3.3. Control Commands

The control inputs u_{roll}, u_{pitch}, and u_{yaw} are roll, pitch, and yaw moments, obtained from the rotors, defined as

$$u_{\text{roll}} = \mathrm{b}\,\mathrm{d}_{\text{cg}}(\Omega_2^2 - \Omega_4^2) \tag{8}$$
$$u_{\text{pitch}} = \mathrm{b}\,\mathrm{d}_{\text{cg}}(\Omega_1^2 - \Omega_3^2) \tag{9}$$
$$u_{\text{yaw}} = \mathrm{d}(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \tag{10}$$

Also, d and b are, respectively, drag and thrust coefficients. d_{cg} is the distance of rotors from the gravity center. Hence, the angular velocity commands are obtained as

$$\Omega_{c,1}^2 = \Omega_{\text{mean}}^2 + \frac{1}{2\mathrm{b}\,\mathrm{d}_{\text{cg}}}u_{\text{pitch}} + \frac{1}{4d}u_{\text{yaw}} \tag{11}$$
$$\Omega_{c,2}^2 = \Omega_{\text{mean}}^2 + \frac{1}{2\mathrm{b}\,\mathrm{d}_{\text{cg}}}u_{\text{roll}} - \frac{1}{4d}u_{\text{yaw}} \tag{12}$$
$$\Omega_{c,3}^2 = \Omega_{\text{mean}}^2 - \frac{1}{2\mathrm{b}\,\mathrm{d}_{\text{cg}}}u_{\text{pitch}} + \frac{1}{4d}u_{\text{yaw}} \tag{13}$$
$$\Omega_{c,4}^2 = \Omega_{\text{mean}}^2 - \frac{1}{2\mathrm{b}\,\mathrm{d}_{\text{cg}}}u_{\text{roll}} - \frac{1}{4d}u_{\text{yaw}} \tag{14}$$

where Ω_{mean} is the nominal of the rotor angular velocities.

3.4. State-Space Form

Here, the state-space model is presented for control purposes. By defining $x_1 = p$, $x_2 = q$, $x_3 = r$, $x_4 = \phi$, $x_5 = \theta$, and $x_6 = \psi$; the model of in state-space form are denoted as

$$\dot{x}_1 = \frac{\mathrm{I_{yy}} - \mathrm{I_{zz}}}{\mathrm{I_{xx}}}x_2x_3 + x_2\frac{\mathrm{I_{rotor}}}{\mathrm{I_{xx}}}\Omega_r + \frac{u_{\text{roll}}}{\mathrm{I_{xx}}} + \frac{d_{\text{roll}}}{\mathrm{I_{xx}}} \tag{15}$$
$$\dot{x}_2 = \frac{\mathrm{I_{zz}} - \mathrm{I_{zz}}}{\mathrm{I_{yy}}}x_1x_3 - x_1\frac{\mathrm{I_{rotor}}}{\mathrm{I_{xx}}}\Omega_r + \frac{u_{\text{pitch}}}{\mathrm{I_{yy}}} + \frac{d_{\text{pitch}}}{\mathrm{I_{yy}}} \tag{16}$$
$$\dot{x}_3 = \frac{\mathrm{I_{xx}} - \mathrm{I_{yy}}}{\mathrm{I_{zz}}}x_1x_2 + \frac{u_{\text{yaw}}}{\mathrm{I_{zz}}} + \frac{d_{\text{yaw}}}{\mathrm{I_{zz}}} \tag{17}$$
$$\dot{x}_4 = x_1 + (x_2\sin(x_4) + x_3\cos(x_4))\tan(x_5) \tag{18}$$
$$\dot{x}_5 = x_2\cos(x_4) - x_3\sin(x_4) \tag{19}$$
$$\dot{x}_6 = (x_2\sin(x_4) + x_3\cos(x_4))/\cos(x_5) \tag{20}$$

Equations (15)-(17) are rewritten in the following form:

$$\dot{x}_1 = \alpha_1x_2x_3 + \alpha_2x_2\Omega_r + \alpha_3u_{\text{roll}} + \alpha_3d_{\text{roll}} \tag{21}$$
$$\dot{x}_2 = \beta_1x_1x_3 - \beta_2x_1\Omega_r + \beta_3u_{\text{pitch}} + \beta_3d_{\text{pitch}} \tag{22}$$
$$\dot{x}_3 = \gamma_1x_1x_2 + \gamma_2u_{\text{yaw}} + \gamma_2d_{\text{yaw}} \tag{23}$$

The measurement model is written as

$$\mathbf{z} = \begin{bmatrix} p_m & q_m & r_m & \phi_m & \theta_m & \psi_m \end{bmatrix}^{\mathrm{T}} \tag{24}$$

3.5. Linear Model

The continuous-time linear model is utilized to drive the control commands on the quadrotor. The linear state-space model is denoted as

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{B_d}\mathbf{d}(t) \tag{25}$$

where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{B_d}$ are the system, input and disturbance matrices, respectively. Moreover, $\mathbf{d}$ is the disturbance. The measurements equation is stated as

$$\mathbf{z}(t) = \mathbf{x}(t) \tag{26}$$

According to equations(15)-(20), the linear dynamic model around the equilibrium points ($\mathbf{x}_e = 0$ and $\mathbf{u}_e = 0$) of the quadrotor setup is denoted as

$$\begin{aligned}\dot{\mathbf{x}} = \begin{bmatrix}\dot{\mathbf{x}}_{\text{roll}}\\ \dot{\mathbf{x}}_{\text{pitch}}\\ \dot{\mathbf{x}}_{\text{yaw}}\end{bmatrix} &= \begin{bmatrix}\mathbf{A}_{\text{roll}} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{A}_{\text{pitch}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{\text{yaw}}\end{bmatrix}\begin{bmatrix}\mathbf{x}_{\text{roll}}\\ \mathbf{x}_{\text{pitch}}\\ \mathbf{x}_{\text{yaw}}\end{bmatrix}\\ &+ \begin{bmatrix}\mathbf{B}_{\text{roll}} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{B}_{\text{pitch}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{B}_{\text{yaw}}\end{bmatrix}\begin{bmatrix}\mathbf{u}_{\text{roll}}\\ \mathbf{u}_{\text{pitch}}\\ \mathbf{u}_{\text{yaw}}\end{bmatrix}\\ &+ \begin{bmatrix}\mathbf{B}_{\text{roll}} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{B}_{\text{pitch}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{B}_{\text{yaw}}\end{bmatrix}\begin{bmatrix}\mathbf{d}_{\text{roll}}\\ \mathbf{d}_{\text{pitch}}\\ \mathbf{d}_{\text{yaw}}\end{bmatrix}\end{aligned} \tag{27}$$

where $\mathbf{x}_{\text{roll}} = \begin{bmatrix}p & \phi\end{bmatrix}^{\text{T}}$, $\mathbf{x}_{\text{pitch}} = \begin{bmatrix}q & \theta\end{bmatrix}^{\text{T}}$, and $\mathbf{x}_{\text{yaw}} = \begin{bmatrix}r & \psi\end{bmatrix}^{\text{T}}$.

Moreover, the state and input matrices are presented as

$$\mathbf{A}_{\text{roll}} = \mathbf{A}_{\text{pitch}} = \mathbf{A}_{\text{yaw}} = \begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix} \tag{28}$$

$$\mathbf{B}_{\text{roll}} = \begin{bmatrix}\frac{1}{\text{I}_{\text{xx}}}\\ 0\end{bmatrix};\ \mathbf{B}_{\text{pitch}} = \begin{bmatrix}\frac{1}{\text{I}_{\text{yy}}}\\ 0\end{bmatrix};\ \mathbf{B}_{\text{yaw}} = \begin{bmatrix}\frac{1}{\text{I}_{\text{zz}}}\\ 0\end{bmatrix} \tag{29}$$

3.6. System Identification

Here, the system identification was made using the simulation of the roll, pitch, and yaw states and sensor data output from the quadrotor setup. For this purpose, the same input is given to the simulated model and setup. In the case of system identification, the cost function was defined as the sum square error between simulations and quadrotor setup measurements. Then, the Nonlinear Least Squares (NLS) optimization technique minimizes the cost function. In NLS, the goal is to look for the model parameters vector $\rho = \begin{bmatrix}\alpha_1 & \alpha_2 & \alpha_3 & \beta_1 & \beta_2 & \beta_3 & \gamma_1 & \gamma_2\end{bmatrix}$, which would minimize the sum of squares of residual errors. In other words, the following cost function has to minimize:

$$\text{Residual sum of squares} = RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{30}$$

where y is the value of the variable to be predicted, and $\hat{y}$ is the predicted value of y, which $\hat{y}$ is a function of the 3DoF setup model parameters vector, ρ, and the system states, $\mathbf{x}$, i.e.:

$$\hat{y}_i = f(\mathbf{x}_i, \rho) \tag{31}$$

One way to minimize RSS is to differentiate RSS with respect to ρ, then set the differentiation to zero and solve for ρ, i.e.:

$$\frac{\partial}{\partial \rho_j} RSS = 0, \quad \forall j \in [1, n] \tag{32}$$

Since there is no closed-form solution for this system of equations, so iterative optimization technique has to be used in which, at each iteration k, minor adjustments have been made to the values of ρ as shown below, and re-evaluate RSS:

$$\rho_j^{(k)} = \rho_j^{(k-1)} + \delta\rho_j \tag{33}$$

Trust Region Reflective (TRR) has been devised to update ρ efficiently. To increase the accuracy of system identification, at first, the parameters of each channel were estimated separately, and then the coupled parameters of the attitude channels were modified. In the parameter modification process, after each parameter modification step mentioned above, the estimated parameters of the previous step are assumed to be fixed, and other parameters are modified. To identification of each stage, several experiments with different scenarios have been performed.

4. Formulation of the Controller Design

In the LQIR-DG controller structure, an integral action is added to the LQR-DG controller to cancel the steady-state errors for reference tracking. For this purpose, first, the augmented state space of the linear quadrotor model is defined to utilize in the controller architecture. Then, the LQR-DG controller design procedure is presented to produce the best control commands for the experimental setup of the quadrotor.

4.1. Augmented State Space Formulation

To add the integral action to the controller structure, the augmented states are defined as follows:

$$\mathbf{x_{a_i}} = \begin{bmatrix} \mathbf{x_i} & \int \mathbf{x_i} \end{bmatrix}^{\mathrm{T}} \tag{34}$$

where i = roll, pitch, and yaw. Then, the quadrotor dynamics model, denoted by Eq.(25), is denoted in the augmented state-space model as

$$\dot{\mathbf{x}}_a(t) = \mathbf{A_a x_a}(t) + \mathbf{B_a u}(t) + \mathbf{B_{d_a} d}(t) \tag{35}$$

where matrices $\mathbf{A_a}$ and $\mathbf{B_a}$ are defined as follows:

$$\mathbf{A_a} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \tag{36}$$

$$\mathbf{B_a} = \mathbf{B_{d_a}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \tag{37}$$

In the above equation $\mathbf{I}$ denotes the identity matrix.

4.2. LQIR-DG Controller Method

The LQIR-DG controller is an optimal and robust method based on the differential game theory. This controller consists of two essential players: one finds the best control command, and the other creates the worst disturbance. For this purpose, the first player tries to minimize a cost function, while the second is assumed to maximize it. Therefore, the quadratic cost function equation is denoted using min-max operators as follows:

$$\min_u \max_d J(\mathbf{x_{a_i}}, u_i, d_i) = J(\mathbf{x_{a_i}}, u_i^*, d_i^*) = \min_u \max_d \int_0^{\mathrm{t_f}} \left(\mathbf{x_{a_i}^T Q_i x_{a_i}} + u_i^{\mathrm{T}} R u_i - d_i^{\mathrm{T}} R_d d_i \right) \mathrm{d}t \tag{38}$$

where R and R_d are symmetric nonnegative definite matrices and $\mathbf{Q_i}$ is a symmetric positive definite matrix. Moreover, $\mathrm{t_f}$ is the final time. ??????????????????To solve this problem, connections between the general optimal problem and the LQIR problem are considered [9]. Consequently, the optimum control effort is computed for each control loop as follows:

$$u_i(t) = -\mathbf{K}_{\boldsymbol{i}}(t)\mathbf{x}_{\mathbf{a_i}}(t) \tag{39}$$
$$d_i(t) = \mathbf{K}_{\mathbf{d_i}}(t)\mathbf{x}_{\mathbf{a_i}}(t) \tag{40}$$

where $\mathbf{K_i}$ and $\mathbf{K_{d_i}}$ are a time varying gain, given by

$$\mathbf{K_i} = R^{-1}\mathbf{B}_{a_i}^{\mathbf{T}}\mathbf{P}_{a_i}(t) \tag{41}$$
$$\mathbf{K_{d_i}} = R_d^{-1}\mathbf{B}_{a_{d_i}}^{\mathbf{T}}\mathbf{P}_{a_{d_i}}(t) \tag{42}$$

where $\mathbf{P}_{a_i}(t)$ and $\mathbf{P}_{a_{d_i}}(t)$ satisfy

$$\dot{\mathbf{P}}_{a_i}(t) = -\mathbf{A}_{\mathbf{a}}^{\mathbf{T}}\mathbf{P}_{\mathbf{a_i}}(t) - \mathbf{P}_{\mathbf{a_i}}(t)\mathbf{A_a} - \mathbf{Q_i} + \mathbf{P}_{\mathbf{a_i}}(t)\mathbf{S}_{\mathbf{a_i}}(t)\mathbf{P}_{\mathbf{a_i}}(t) + \mathbf{P}_{\mathbf{a_i}}(t)\mathbf{S}_{\mathbf{a_{d_i}}}(t)\mathbf{P}_{\mathbf{a_{d_i}}}(t) \tag{43}$$
$$\dot{\mathbf{P}}_{a_{d_i}}(t) = -\mathbf{A}_{\mathbf{a}}^{\mathbf{T}}\mathbf{P}_{\mathbf{a_{d_i}}}(t) - \mathbf{P}_{\mathbf{a_{d_i}}}(t)\mathbf{A_a} - \mathbf{Q_i} + \mathbf{P}_{\mathbf{a_{d_i}}}(t)\mathbf{S}_{\mathbf{a_{d_i}}}(t)\mathbf{P}_{\mathbf{a_{d_i}}}(t) + \mathbf{P}_{\mathbf{a_{d_i}}}(t)\mathbf{S}_{\mathbf{a_i}}(t)\mathbf{P}_{\mathbf{a_i}}(t) \tag{44}$$

where $\mathbf{S}_{\mathbf{a_i}} = \mathbf{B}_{\mathbf{a_i}}R^{-1}\mathbf{B}_{a_i}^{\mathbf{T}}$ and $\mathbf{S}_{\mathbf{a_{d_i}}} = \mathbf{B}_{a_{d_i}}R_d^{-1}\mathbf{B}_{a_{d_i}}^{\mathbf{T}}$. In this study, the steady-state values of the above equations ($\mathbf{P}$ as $t_f \to \infty$) are utilized to generate a feedback control law.

5. Result and Discussion

Here, the results of the LQIR-DG controller method are devoted to the control loops of the roll, pitch, and yaw of the experimental setup of the quadrotor. First, the controller parameters are tuned using the results of numerical simulations. Moreover, the performance of the LQIR-DG controller is compared to an LQR control strategy. The quadrotor parameters are shown in table 1. Moreover, the parameters of LQIR-DG controller weight are denoted in table 2.

Table 1. The Parameter of the Quadrotor

Parameter	***Value***	***Unit***
I_{xx}	0.02839	$kg.m^2$
I_{yy}	0.03066	$kg.m^2$
I_{zz}	0.0439	$kg.m^2$
I_{rotor}	4.4398×10^{-5}	$kg.m^2$
b	3.13×10^{-5}	$N.\,sec^2/rad^2$
d	3.2×10^{-6}	$N.m.\,sec^2/rad^2$
Ω_{mean}	3000	rpm
d_{cg}	0.2	m

References

[1] M. F. Fathoni, S. Lee, Y. Kim, K.-I. Kim, K. H. Kim, Development of multi-quadrotor simulator based on real-time hypervisor systems, Drones 5 (3). doi:10.3390/drones5030059.
URL `https://www.mdpi.com/2504-446X/5/3/59`

[2] H. Nobahari, A. Sharifi, A hybridization of extended kalman filter and ant colony optimization for state estimation of nonlinear systems, Applied Soft Computing 74. doi:10.1016/j.asoc.2018.10.010.

[3] H. Bolandi, M. Rezaei, R. Mohsenipour, H. Nemati, S. Smailzadeh, Attitude control of a quadrotor with optimized pid controller, Intelligent Control and Automation 04 (2013) 342–349. doi:10.4236/ica.2013.43040.

Table 2. The Parameters of the LQIR-DG Controller?????????????

Control Loop	***Weight***	***Value***
Roll	$\mathbf{Q}_{\text{roll}}$	diag([7.91, 0.01, 631.85, 214.28])
Pitch	$\mathbf{Q}_{\text{pitch}}$	diag([9853.09, 0.12, 0.01, 873.93])
Yaw	$\mathbf{Q}_{\text{yaw}}$	diag([$1.81e{-}4$, $4.5e{-}4$, $3e{-}6$, $1.7e{-}5$])
-	R	1
-	R_d	1.2577

[4] A. Abdul Salam, I. Ibraheem, Nonlinear pid controller design for a 6-dof uav quadrotor system, Engineering Science and Technology, an International Journal 22. doi:10.1016/j.jestch.2019.02.005.

[5] Y. Bouzid, M. Zareb, H. Siguerdidjane, M. Guiatni, Boosting a Reference Model-Based Controller Using Active Disturbance Rejection Principle for 3D Trajectory Tracking of Quadrotors: Experimental Validation, Journal of Intelligent and Robotic Systems 100 (2) (2020) 597–614. doi:10.1007/s10846-020-01182-4.
URL `https://hal.univ-grenoble-alpes.fr/hal-02543214`

[6] Z. Wang, D. Huang, T. Huang, N. Qin, Active disturbance rejection control for a quadrotor uav, in: 2020 IEEE 9th Data Driven Control and Learning Systems Conference (DDCLS), 2020, pp. 1–5. doi:10.1109/DDCLS49620.2020.9275226.

[7] S. Bouabdallah, R. Siegwart, Full control of a quadrotor, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007, pp. 153–158. doi:10.1109/IROS.2007.4399042.

[8] S. Bouabdallah, Design and control of quadrotors with application to autonomous flyingdoi:10.5075/epfl-thesis-3727.

[9] J. Engwerda, Linear quadratic games: An overview, Workingpaper, Macroeconomics, subsequently published in Advances in Dynamic Games and their Applications (book), 2009 Pagination: 32 (2006).