

Attitude Control of a 3-DoF Quadrotor Platform using a Linear Quadratic Integral Differential Game Approach

Abstract

In this study, a linear quadratic integral differential game approach is applied to regulate and track the Euler angles for a quadrotor experimental platform using two players. One produces commands for each channel of the quadrotor and another generates the worst disturbance based on the mini-maximization of a quadratic criterion with integral action. For this purpose, first, the attitude dynamics of the platform are modeled and its parameters are identified based on the Nonlinear Least Squares Trust-Region Reflective method. The performance of the proposed controller is evaluated for regulation and tracking problems. The ability of the controller is also examined in the disturbance rejection. Moreover, the influence of uncertainty modeling is studied on the obtained results. Then, the performance of the proposed controller is compared with the classic Proportional Integral Derivative, Linear Quadratic Regulator, and Linear Quadratic Integral Regulator. The results demonstrate the effectiveness of the Game Theory on the Linear Quadratic Regulator approach when the input disturbance occurs.

Keywords:

Linear Quadratic controller, Differential Game Theory, Quadrotor, 3-DoF Experimental Platform, Attitude Control.

1. Introduction

Quadrotors, a type of Vertical Unmanned Aerial Vehicle (VUAV), have found diverse applications in investigation, strategic operations, optical sensing, and entertainment. Precise control is crucial for the safe flight of quadrotors in the presence of disturbances. The Attitude Control System (ACS) plays a vital role in regulating the quadrotor's attitude, and the Proportional Integral Derivative (PID) controller has been commonly employed for this purpose in previous studies [1, 5]. However, due to the nonlinearity of the quadrotor dynamics, the PID strategy's effectiveness diminishes in the presence of disturbances and modeling errors.

To address these challenges and enhance the quadrotor's attitude control, various model-based control strategies have been implemented on the ACS. These strategies encompass nonlinear control, intelligent control, optimal control, and robust control approaches.

Nonlinear control methods such as Synergetic Control [8], Feedback Linearization (FBL) [2], Sliding Mode Control (SMC) [15, 25, 17, 9, 22, 28] have been utilized to regulate the quadrotor's Euler angles (roll, pitch, and yaw angles) intelligently.

Intelligent control approaches, including reinforcement learning [14, 16, 26, 24], iterative learning [12], machine learning [10], and fuzzy logic [13], have also been employed to control the attitude of the quadrotor.

Optimal control strategies, such as Linear Quadratic Gaussian (LQG) [29], Linear Quadratic Regulator (LQR) [3], Linear Quadratic Integral Regulator (LQIR) [4], and Model Predictive Controller (MPC) [27, 21], have been applied to generate optimal control commands for the quadrotor.

In the domain of robust control, techniques like H_∞ control [23, 20], μ -synthesis, and Linear Quadratic Regulator Differential Game (LQR-DG) [18] have been used to stabilize the quadrotor's Euler angles, considering worst-case scenarios and mini-maximization of a quadratic criterion, which includes control effort and regulation performance.

In this paper, an LQIR-DG method is implemented real-time on 3-DoF experimental platform of the quadrotor to produce the robust control commands, i.e. rotational velocity of the quadrotor. To this

end, first, the experimental platform of the quadrotor is modeled using the Newton-Euler formulation and its linear state-space form is derived. Then, the parameters of the quadrotor are estimated by matching experimental data with results from the model simulation. In the next step, the proposed controller is implemented on the Arduino Mega2560 board using the embedded coder platform in MATLAB and its performance is investigated in regulation and tracking problems. Moreover, the rejection capability of the input disturbance and modeling error is tested. Finally, a comparison is also performed between the results of classical PID, LQR, and LQIR with the proposed method. The results demonstrate that this method has an excellent performance in the attitude control of the quadrotor platform. A demo video of the results is available online here.

The remainder of this research is organized as follows: Section 2 presents the problem statement. Section 3 outlines the dynamic platform modeling process. The proposed controller architecture is described in Section 4. Section 5 presents the numerical results, and Section 6 concludes the paper. A demo video showcasing the results is available online ¹.

2. Problem Statement

The experimental quadrotor platform rotates freely with rotational velocity ($\Omega_i, i = 1, 2, 3, 4$) about its roll, pitch, and yaw axes, according to Figure 1. The angular velocities in the body frame (p, q, r) and the Euler angles (ϕ, θ, ψ) are measured using an Attitude Heading Reference System (AHRS). The measured states are utilized in the structure of the proposed controller to stabilize the quadrotor platform. The graphical abstract of the LQIR-DG controller strategy is depicted in Figure 2.



Figure 1: 3-DoF Quadrotor platform.

¹Demo video link: https://drive.google.com/drive/folders/1DIJs3wmIpmpwI8slyHeitA6Ebe-khKTct?usp=share_link



Figure 2: Graphical abstract of the LQIR-DG controller.

3. Model of the Quadrotor Platform

Here, the quadrotor platform is modeled as nonlinear. Then, a state-space model and a linear model are developed for control purposes to be utilized in the controller strategy. Finally, a nonlinear identification method is applied to identify the parameters of the quadrotor.

3.1. Quadrotor Configuration

According to Figure 3, the 3-DoF quadrotor schematic is including four rotors rotating the z_B axis in the body frame with a rotational velocity, Ω_i ($i = 1, 2, 3, 4$). To eliminate the yawing moment, rotors (2, 4) and (1, 3) rotate clockwise and counter clockwise, respectively.

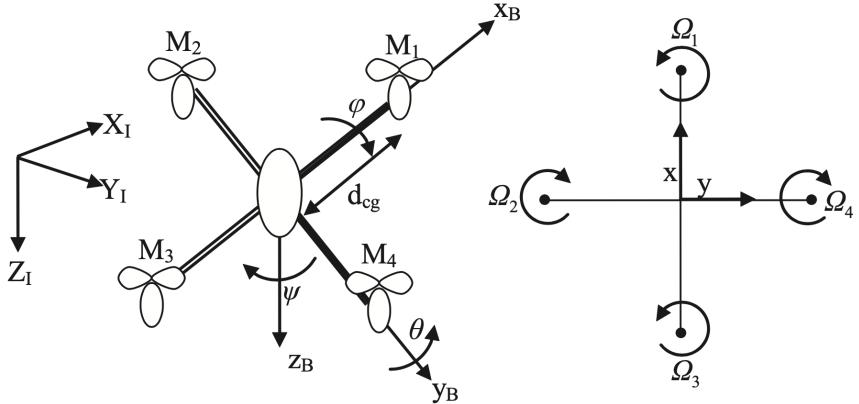


Figure 3: Quadrotor configuration.

3.2. Dynamic Modeling of the Quadrotor Platform

Here, according to Newton-Euler, the model of the quadrotor platform is presented as follows [7, 6]:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{J_z}{\Gamma} & 0 & \frac{J_{xy}}{\Gamma} \\ 0 & \frac{1}{J_y} & 0 \\ \frac{J_{xy}}{\Gamma} & 0 & \frac{J_x}{\Gamma} \end{bmatrix} \left(\begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \begin{bmatrix} J_x & 0 & -J_{xy} \\ 0 & J_y & 0 \\ -J_{xy} & 0 & J_z \end{bmatrix} + \begin{bmatrix} u_{\text{roll}} \\ u_{\text{pitch}} \\ u_{\text{yaw}} \end{bmatrix} + \begin{bmatrix} d_{\text{roll}} \\ d_{\text{pitch}} \\ d_{\text{yaw}} \end{bmatrix} \right) \quad (1)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \frac{1}{\Gamma} \begin{bmatrix} J_z & 0 & J_{xy} \\ 0 & \frac{\Gamma}{J_y} & 0 \\ J_{xy} & 0 & J_x \end{bmatrix} \left(\begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \begin{bmatrix} J_x & 0 & -J_{xy} \\ 0 & J_y & 0 \\ -J_{xy} & 0 & J_z \end{bmatrix} + \begin{bmatrix} u_{\text{roll}} \\ u_{\text{pitch}} \\ u_{\text{yaw}} \end{bmatrix} + \begin{bmatrix} d_{\text{roll}} \\ d_{\text{pitch}} \\ d_{\text{yaw}} \end{bmatrix} \right). \quad (2)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \frac{1}{\Gamma} \begin{bmatrix} J_z & 0 & J_{xy} \\ 0 & \frac{\Gamma}{J_y} & 0 \\ J_{xy} & 0 & J_x \end{bmatrix} \left(\begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \begin{bmatrix} J_x & 0 & -J_{xy} \\ 0 & J_y & 0 \\ -J_{xy} & 0 & J_z \end{bmatrix} + \begin{bmatrix} b d_{\text{cg}}(\Omega_{c,2}^2 - \Omega_{c,4}^2) \\ b d_{\text{cg}}(\Omega_{c,1}^2 - \Omega_{c,3}^2) \\ d(\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2) \end{bmatrix} + \begin{bmatrix} d_{\text{roll}} \\ d_{\text{pitch}} \\ d_{\text{yaw}} \end{bmatrix} \right) \\ \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \frac{1}{\Gamma} \begin{bmatrix} J_z & 0 & J_{xy} \\ 0 & \frac{\Gamma}{J_y} & 0 \\ J_{xy} & 0 & J_x \end{bmatrix} \left(\begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \begin{bmatrix} J_x & 0 & -J_{xy} \\ 0 & J_y & 0 \\ -J_{xy} & 0 & J_z \end{bmatrix} + \begin{bmatrix} b d_{\text{cg}}(\Omega_{c,2}^2 - \Omega_{c,4}^2) \\ b d_{\text{cg}}(\Omega_{c,1}^2 - \Omega_{c,3}^2) \\ d(\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2) \end{bmatrix} + \begin{bmatrix} d_{\text{roll}} \\ d_{\text{pitch}} \\ d_{\text{yaw}} \end{bmatrix} \right) \quad (3)$$

$$\dot{p} = \frac{I_{yy} - I_{zz}}{I_{xx}} qr + q \frac{I_{\text{rotor}}}{I_{xx}} \Omega_{c,r} + \frac{b d_{\text{cg}}(\Omega_{c,2}^2 - \Omega_{c,4}^2)}{I_{xx}} + \frac{d_{\text{roll}}}{I_{xx}} \quad (4)$$

$$\dot{q} = \frac{I_{zz} - I_{xx}}{I_{yy}} rp - p \frac{I_{\text{rotor}}}{I_{yy}} \Omega_{c,r} + \frac{b d_{\text{cg}}(\Omega_{c,1}^2 - \Omega_{c,3}^2)}{I_{yy}} + \frac{d_{\text{pitch}}}{I_{yy}} \quad (5)$$

$$\dot{r} = \frac{I_{xx} - I_{yy}}{I_{zz}} pq + \frac{d(\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2)}{I_{zz}} + \frac{d_{\text{yaw}}}{I_{zz}} \quad (6)$$

where Γ is defined as

$$\Gamma = J_x J_z - J_{xy}^2 \quad (7)$$

and $\Omega_{c,i}$ ($i = 1, 2, 3, 4$) is the rotational velocity, computed as

$$\Omega_{c,1}^2 = \Omega_{\text{mean}}^2 + \frac{1}{2b d_{\text{cg}}} u_{\text{pitch}} + \frac{1}{4d} u_{\text{yaw}} \quad (8)$$

$$\Omega_{c,2}^2 = \Omega_{\text{mean}}^2 + \frac{1}{2b d_{\text{cg}}} u_{\text{roll}} - \frac{1}{4d} u_{\text{yaw}} \quad (9)$$

$$\Omega_{c,3}^2 = \Omega_{\text{mean}}^2 - \frac{1}{2b d_{\text{cg}}} u_{\text{pitch}} + \frac{1}{4d} u_{\text{yaw}} \quad (10)$$

$$\Omega_{c,4}^2 = \Omega_{\text{mean}}^2 - \frac{1}{2b d_{\text{cg}}} u_{\text{roll}} - \frac{1}{4d} u_{\text{yaw}} \quad (11)$$

In the above equation, Ω_{mean} is the rotational velocity of the rotors. Also, d_{cg} , d , and b represent the distance between the rotors and the gravity center, drag factor, and thrust factor, respectively. d_{roll} , d_{pitch} , and d_{yaw} denote the disturbances produced in the body coordinate frame. Additionally, u_{roll} , u_{pitch} , and

u_{yaw} are control commands generated by the LQIR-DG controller. I_{xx} , I_{yy} , and I_{zz} are the moments of inertia. Euler angle rates are also determined from angular body rates as follows:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi) \tan(\theta) & \cos(\phi) \tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (12)$$

3.3. State-Space Formulation

By defining $\mathbf{x}_{\text{roll}} = [x_1 \ x_2]^T = [p \ \phi]^T$, $\mathbf{x}_{\text{pitch}} = [x_3 \ x_4]^T = [q \ \theta]^T$, and $\mathbf{x}_{\text{yaw}} = [x_5 \ x_6]^T = [r \ \psi]^T$, the formulation of the quadrotor platform is presented as follows:

$$\dot{x}_1 = \Gamma_1 x_3 x_5 + \Gamma_2 x_3 \Omega_r + \Gamma_3 b d_{\text{cg}} (\Omega_{c,2}^2 - \Omega_{c,4}^2) + \Gamma_3 d_{\text{roll}} \quad (13)$$

$$\dot{x}_2 = x_1 + (x_3 \sin(x_2) + x_3 \cos(x_2)) \tan(x_4) \quad (14)$$

$$\dot{x}_3 = \Gamma_4 x_1 x_5 - \Gamma_5 x_1 \Omega_r + \Gamma_6 b d_{\text{cg}} (\Omega_{c,1}^2 - \Omega_{c,3}^2) + \Gamma_6 d_{\text{pitch}} \quad (15)$$

$$\dot{x}_4 = x_3 \cos(x_4) - x_5 \sin(x_2) \quad (16)$$

$$\dot{x}_5 = \Gamma_7 x_1 x_3 + \Gamma_8 d (\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2) + \Gamma_8 d_{\text{yaw}} \quad (17)$$

$$\dot{x}_6 = (x_3 \sin(x_4) + x_5 \cos(x_2)) / \cos(x_4) \quad (18)$$

where $\Gamma_i (i = 1, \dots, 8)$ is defined as

$$\begin{aligned} \Gamma_1 &= \frac{I_{yy} - I_{zz}}{I_{xx}}, & \Gamma_2 &= \frac{I_{\text{rotor}}}{I_{xx}}, & \Gamma_3 &= \frac{1}{I_{xx}} \\ \Gamma_4 &= \frac{I_{zz} - I_{xx}}{I_{yy}}, & \Gamma_5 &= \frac{I_{\text{rotor}}}{I_{yy}}, & \Gamma_6 &= \frac{1}{I_{yy}} \\ \Gamma_7 &= \frac{I_{xx} - I_{yy}}{I_{zz}}, & \Gamma_8 &= \frac{1}{I_{zz}} \end{aligned} \quad (19)$$

$$\dot{x}_1 = \Gamma_1 x_1 x_3 - \Gamma_2 x_3 x_5 + \Gamma_3 b d_{\text{cg}} (\Omega_{c,2}^2 - \Omega_{c,4}^2) + \Gamma_4 d (\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2) + \Gamma_3 d_{\text{roll}} \quad (20)$$

$$\dot{x}_2 = x_1 + (x_3 \sin(x_2) + x_3 \cos(x_2)) \tan(x_4) \quad (21)$$

$$\dot{x}_3 = \Gamma_5 x_1 x_5 - \Gamma_6 (x_1^2 - x_5^2) + \frac{1}{J_y} b d_{\text{cg}} (\Omega_{c,1}^2 - \Omega_{c,3}^2) + \frac{1}{J_y} d_{\text{pitch}} \quad (22)$$

$$\dot{x}_4 = x_3 \cos(x_4) - x_5 \sin(x_2) \quad (23)$$

$$\dot{x}_5 = \Gamma_7 x_1 x_3 - \Gamma_1 x_3 x_5 + \Gamma_4 b d_{\text{cg}} (\Omega_{c,2}^2 - \Omega_{c,4}^2) + \Gamma_8 d (\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2) + \Gamma_8 d_{\text{yaw}} \quad (24)$$

$$\dot{x}_6 = (x_3 \sin(x_4) + x_5 \cos(x_2)) / \cos(x_4) \quad (25)$$

where $\Gamma_i (i = 1, \dots, 8)$ is defined as

$$\begin{aligned} \Gamma_1 &= \frac{J_{xz}(J_x - J_y + J_z)}{\Gamma}, & \Gamma_2 &= \frac{J_z(J_z - J_y) + J_{xz}^2}{\Gamma}, & \Gamma_3 &= \frac{J_z}{\Gamma}, \\ \Gamma_4 &= \frac{J_{xz}}{\Gamma}, & \Gamma_5 &= \frac{J_z - J_x}{J_y}, & \Gamma_6 &= \frac{J_{xz}}{J_y}, \\ \Gamma_7 &= \frac{J_x(J_x - J_y) + J_{xz}^2}{\Gamma}, & \Gamma_8 &= \frac{J_x}{\Gamma}. \end{aligned} \quad (26)$$

The measurement vector, obtained from the AHRS, is presented as follows:

$$\mathbf{z} = [p \ q \ r \ \phi \ \theta \ \psi]^T + \boldsymbol{\nu} \quad (27)$$

where $\boldsymbol{\nu}$ is a Gaussian white noise. Moreover, the superscripts T indicate the transpose notation.

3.4. Linear Model

By defining $\dot{\mathbf{x}} = [\dot{\mathbf{x}}_{\text{roll}} \quad \dot{\mathbf{x}}_{\text{pitch}} \quad \dot{\mathbf{x}}_{\text{yaw}}]^T$, the linear model of the quadrotor platform represented about the equilibrium points ($\mathbf{x}_e^* = 0$ and $\mathbf{u}_e^* = 0$) as

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} (\mathbf{u} + \mathbf{d}) \quad (28)$$

where $d = \text{diag}([d_{\text{roll}}, d_{\text{pitch}}, d_{\text{yaw}}])$ denotes the input disturbance. \mathbf{A} is the dynamic system matrix, denoted as

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{\text{roll}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{\text{pitch}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{\text{yaw}} \end{bmatrix} \quad (29)$$

$\mathbf{A}_{\text{roll}} = \mathbf{A}_{\text{pitch}} = \mathbf{A}_{\text{yaw}} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$. Also, \mathbf{B} is the input matrix defined as

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_{\text{roll}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{\text{pitch}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}_{\text{yaw}} \end{bmatrix} \quad (30)$$

where $\mathbf{B}_{\text{roll}} = \begin{bmatrix} 1 & 0 \\ \Gamma_3 & 0 \end{bmatrix}^T$, $\mathbf{B}_{\text{pitch}} = \begin{bmatrix} 1 & 0 \\ I_{yy} & 0 \end{bmatrix}^T$, and $\mathbf{B}_{\text{yaw}} = \begin{bmatrix} 1 & 0 \\ \Gamma_8 & 0 \end{bmatrix}^T$.

3.5. Identification of the Platform Parameters

In this section, the Nonlinear Least Squares (NLS) algorithm is utilized for estimating the model parameters (Γ) of the 3-DoF experimental platform using experimental data. This technique is based on the Trust-Region Reflective (TRR) method, which finds the best values for Γ by minimizing a cost function, defined as

$$\min_{\Gamma} (\| e(\Gamma) \|^2) = \min_{\Gamma} \left(\sum_{j=1}^n (\mathbf{z}_j - \tilde{\mathbf{z}}_j)(\mathbf{z}_j - \tilde{\mathbf{z}}_j)^T \right) \quad (31)$$

where \mathbf{z} and $\tilde{\mathbf{z}}$ are the experimental and simulated output signals when the same input signals are applied. Moreover, n is the number of scenarios. To find a vector Γ , the optimization process performs until convergence is achieved. The structure of the identification approach is illustrated in figure 4.



Figure 4: Structure of TRRLS identification approach.

4. LQIR-DG Controller Structure

First, the augmented states of the quadrotor platform, including the states and their integrals are selected to use in the structure of the LQIR-DG controller for eliminating the steady-state errors. Then, the design methodology of the controller structure is introduced to produce the best commands for the 3-DoF quadrotor platform.

4.1. Augmented States

To augment an integral action into the control strategy architecture, the augmented states are defined as $\mathbf{x}_a = \begin{bmatrix} \mathbf{x} & \int \mathbf{x} \end{bmatrix}^T$. Then, the quadrotor platform model, utilized in the controller structure, is presented as

$$\dot{\mathbf{x}}_a = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{x}_a + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} (\mathbf{u} + \mathbf{d}) \quad (32)$$

The notation \mathbf{I} denotes the identity matrix.

4.2. LQIR-DG Control Scheme with Integral Action

In the proposed controller scheme, two fundamental players are selected in accordance with the game theory approach. The primary player determines the control commands, while another player generates the worst possible disturbance. To achieve the primary objective, the first player minimizes the following cost function but the other player maximizes it:

$$\min_u \max_d J(\mathbf{x}_{\mathbf{a}_i}, d_i, u_i) = \min_d \max_u \int_0^{t_f} \left(\mathbf{x}_{\mathbf{a}_i}^T \mathbf{Q}_i \mathbf{x}_{\mathbf{a}_i} + u_i^T R u_i - d_i^T R_d d_i \right) dt \quad (33)$$

where t_f is the stop time and i -index denotes the roll, pitch, and yaw channels of the quadrotor. \mathbf{Q}_i , R_d , and R are weight coefficients of the cost function. By solving the above problem, the optimal control command is computed as follows [11]:

$$u_i = -\mathbf{K}_i \mathbf{x}_{\mathbf{a}_i} \quad (34)$$

Moreover, the worst disturbance is obtained as

$$d_i = \mathbf{K}_{d_i} \mathbf{x}_{\mathbf{a}_i} \quad (35)$$

Here, \mathbf{K}_{d_i} and \mathbf{K}_i are gain values defined as follows:

$$\mathbf{K}_{d_i} = R_d^{-1} \mathbf{B}_{a_{d_i}}^T \mathbf{P}_{a_{d_i}} \quad (36)$$

$$\mathbf{K}_i = R^{-1} \mathbf{B}_{a_i}^T \mathbf{P}_{a_i} \quad (37)$$

\mathbf{P}_{a_i} and $\mathbf{P}_{a_{d_i}}$ satisfy

$$-\mathbf{A}_a^T \mathbf{P}_{a_{d_i}} - \mathbf{Q}_i - \mathbf{P}_{a_{d_i}} \mathbf{A}_a + \mathbf{P}_{a_{d_i}} \mathbf{S}_{a_i} \mathbf{P}_{a_i} + \mathbf{P}_{a_{d_i}} \mathbf{S}_{a_{d_i}} \mathbf{P}_{a_{d_i}} = \mathbf{0} \quad (38)$$

$$-\mathbf{A}_a^T \mathbf{P}_{a_i} - \mathbf{Q}_i - \mathbf{P}_{a_i} \mathbf{A}_a + \mathbf{P}_{a_i} \mathbf{S}_{a_{d_i}} \mathbf{P}_{a_{d_i}} + \mathbf{P}_{a_i} \mathbf{S}_{a_i} \mathbf{P}_{a_i} = \mathbf{0} \quad (39)$$

where $\mathbf{S}_{a_i} = \mathbf{B}_{a_i} R^{-1} \mathbf{B}_{a_i}^T$ and $\mathbf{S}_{a_{d_i}} = \mathbf{B}_{a_{d_i}} R_d^{-1} \mathbf{B}_{a_{d_i}}^T$.

4.3. TCACS Optimization for Tuning the Weighting Matrices

To optimize the weighting matrix of the LQIR-DG (Linear Quadratic Integral Regulator with Disturbance Rejection) controller, the TCACS (Tabu Continuous Ant Colony System) [19] optimization method was utilized. The objective was to tune the controller parameters for improved performance in a 3-degree-of-freedom simulation.

In the optimization process, the cost function was formulated based on the LQIR-DG controller, where the state feedback matrix \mathbf{Q} and the disturbance rejection matrix R_d were the key parameters to be determined. To simplify the problem, it was assumed that R (the penalty matrix for control inputs) was fixed at a value of 1.

By employing the TCACS optimization method, the algorithm explored the search space to find the optimal values of \mathbf{Q} and R_d for each channel. The objective was to achieve a balance between control effort and disturbance rejection while ensuring stable and robust control performance.

The optimization process aimed to fine-tune the controller parameters for the specific dynamics of the system under consideration. The resulting weighting matrices \mathbf{Q} and R_d would enable the LQIR-DG controller to efficiently regulate the system while effectively rejecting disturbances in the simulation. In this section, the TCACS optimization method is utilized to tune the weighting matrices of the LQIR-DG controller.

5. Results

The results of the parameter identification and the LQIR-DG Controller for the quadrotor platform are presented. First, the quadrotor parameters are estimated based on the NLS method. Then, the performance

of the LQIR-DG structure is evaluated. Tables 1 and 2 present the quadrotor and LQIR-DG parameters, respectively.

Table 1: Quadrotor parameters

Parameter	Unit	Value	Parameter	Unit	Value
m_{total}	kg	1.074	I_{xx}	kg.m ²	0.02839
d	N.m.sec ² /rad ²	3.2×10^{-6}	I_{yy}	kg.m ²	0.03066
b	N.sec ² /rad ²	3.13×10^{-5}	I_{zz}	kg.m ²	0.0439
d_{cg}	m	0.2	I_{rotor}	kg.m ²	4.4398×10^{-5}
Ω_{mean}	rpm	2000	I_{xz}	kg.m ²	6.87×10^{-7}

5.1. Tuning of LQIR-DG Weighting Matrix

The TCACS (Tabu Continuous Ant Colony System) optimization method was employed to fine-tune the weighting matrices \mathbf{Q} and R_d of the LQIR-DG controller. The objective was to enhance the controller's performance in a 3-degree-of-freedom simulation. The psudo code of TCACS is shown in Algorithm 1.

The cost function for the optimization was formulated based on the Integral Squared Time Error (ITSE) criterion, given by:

$$\text{ITSE} = \int_{t_1}^{t_2} (e(t)^2 t) dt$$

where \mathbf{x} represents the state vector and \mathbf{u} denotes the control input vector.

Due the relation between the weighting matrices is important, the R matrix is assumed as 1, while the weighting matrix \mathbf{Q} is represented as $\text{diag}([\mathbf{Q}_{\text{Roll}}, \mathbf{Q}_{\text{Pitch}}, \mathbf{Q}_{\text{Yaw}}])$, and R_d is individually optimized for each channel. The optimization process was conducted for the weighting matrices \mathbf{Q} and R_d on a per-channel basis. The primary objective was to minimize the ITSE cost function concerning control effort and disturbance rejection, while simultaneously ensuring stable and robust control performance.

Table 2: LQIR-DG controller parameters

Channel	Weighting Matrix	Values
Roll	\mathbf{Q}_{roll}	$\text{diag}([0.02, 65.96, 83.04, 0.00])$
Pitch	$\mathbf{Q}_{\text{pitch}}$	$\text{diag}([435.01, 262.60, 262.60, 0.00])$
Yaw	\mathbf{Q}_{yaw}	$\text{diag}([4 \times 10^{-4}, 0.00, 0.133, 0])$
-	R	1
-	R_d	1.2764

5.2. Identification of the 3-DoF quadrotor platform model

As described in section 3.3, the parameters of the quadrotor platform, denoted by $\Gamma_i (i = 1, \dots, 8)$, are identified using the NLS-TRR algorithm. To increase the accuracy of parameter identification, three scenarios are considered according to Table 3. In the first scenario, depicted in Figure 5, the quadrotor rotates about only one axis (roll, pitch, or yaw axes) to identify the parameters Γ_3 , Γ_6 , and Γ_8 . In the second scenario, according to Figure 6, the parameters Γ_2 and Γ_5 are estimated by rotating the experimental platform around its roll and pitch axes simultaneously. Finally, Figure 7 displays the results of the third scenario including the estimation of the parameters Γ_1 , Γ_4 , and Γ_7 for the UAV model, when the platform freely rotates around three axes. After the termination condition is met, the optimal values of the quadrotor parameters

Algorithm 1 Tabu Continuous Ant Colony System (TCACS)

```
1: procedure TCACS
2:   Initialize parameters
3:   Initialize tabu list  $TL$  and promising list  $PL$ 
4:   while termination condition not met do
5:     if first iteration then
6:       Uniformly sample initial ant positions
7:     else
8:       Move ants to new locations
9:     end if
10:    Update  $TL$ 
11:    Update  $PL$ 
12:    Update tabu ball radius  $r_T$ 
13:    Update coordinate transformation matrix
14:    Update pheromone distribution
15:   end while
16: end procedure
17: procedure MOVEANTS
18:   for each ant  $i$  do
19:     repeat
20:       for each dimension  $j$  do
21:         Sample new  $x_j^i$  from normal distribution
22:       end for
23:       until  $x^i$  not in tabu ball
24:   end for
25: end procedure
26: procedure UPDATEPHEROMONE
27:   Update best solution  $x^*$ 
28:   for each dimension  $j$  do
29:     Update standard deviation  $\sigma_j$ 
30:   end for
31: end procedure
```

are computed and denoted in Table 4. These results illustrate that the outputs of the simulation results for the quadrotor model are consistent with reality.

Table 3: Scenarios for identification of quadrotor parameters.

Scenario	Description	Initial Condition (deg)			Rotational Velocity Commands (rpm)			
		ϕ_0	θ_0	ψ_0	Ω_1	Ω_2	Ω_3	Ω_4
I	roll free	38	-	-	2000	2000	2000	3400
	pitch free	-	-15	-	3700	2000	2000	2000
	yaw free	-	-	-75	2000	3300	2000	3300
II	roll & pitch free	8	-5	-	1700	3800	2400	1700
III	roll, pitch, & yaw free	8	-3	-146	1700	3800	2400	1700

Table 4: True values of the quadrotor parameters.

Parameter	Value	Parameter	Value
Γ_1	-0.9622	Γ_5	5.8336×10^{-6}
Γ_2	3.0516×10^{-7}	Γ_6	7.5395×10^{-5}
Γ_3	5.4716×10^{-5}	Γ_7	0.1308
Γ_4	1.0457	Γ_8	4.3753×10^{-5}

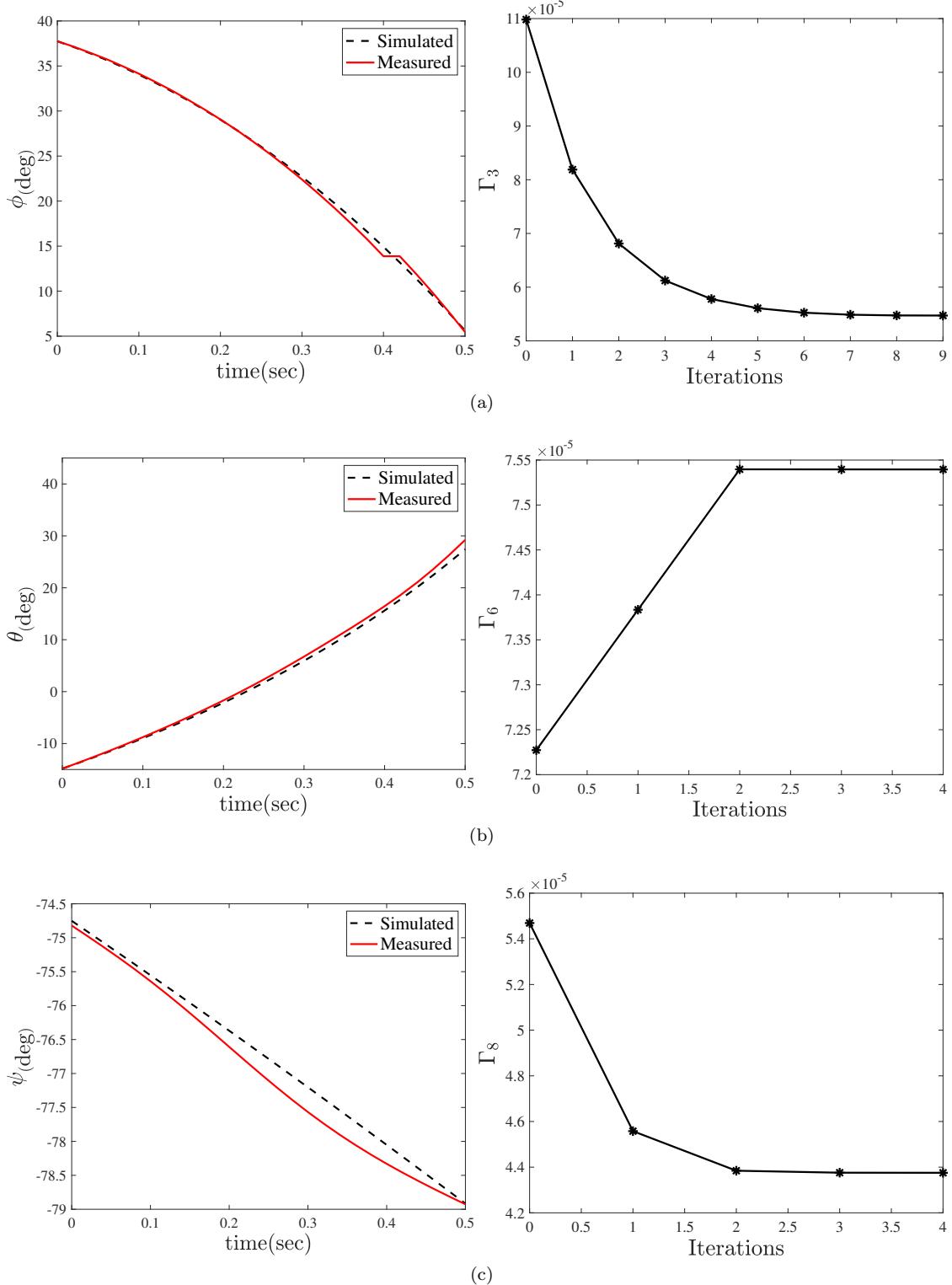


Figure 5: Identification process results when the quadrotor rotates about only one axis: (a) identification of Γ_3 in free roll motion. (b) identification of Γ_6 in free pitch motion. (c) identification of Γ_8 in free yaw motion.

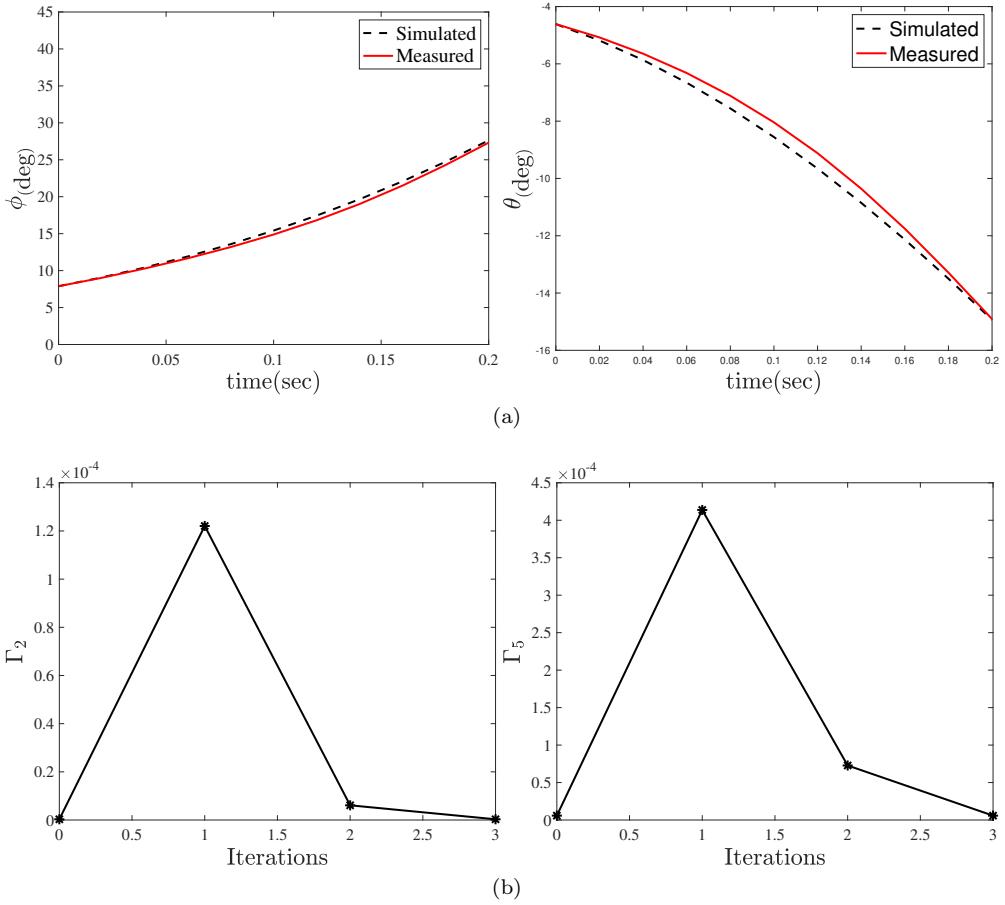


Figure 6: Identification process results when the quadrotor rotates about its roll and pitch axes: (a) comparison of simulation and experimental results. (b) identification of Γ_2 and Γ_5 .

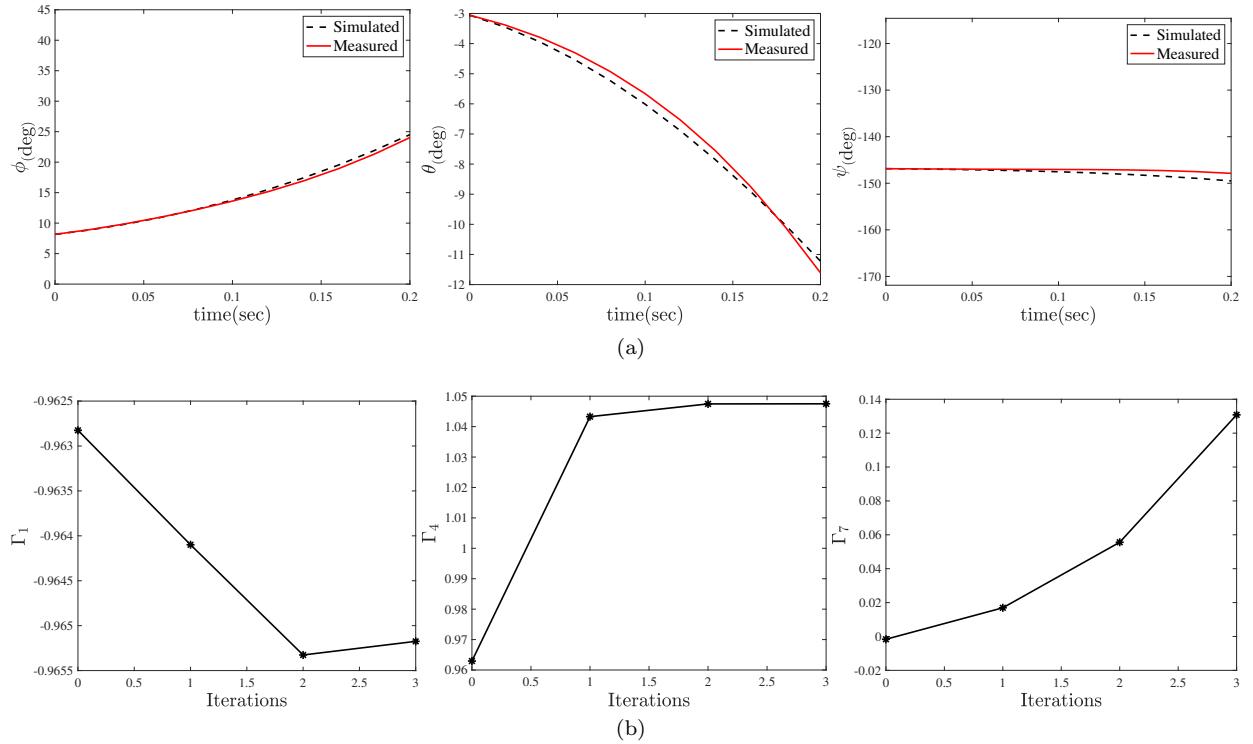


Figure 7: Identification process results when the quadrotor rotates about its roll, pitch, and yaw axes: (a) comparison of simulation and experimental results. (b) identification of Γ_1 , Γ_4 and Γ_7 parameters.

5.3. Evaluation of LQIR-DG Performance

In this section, the LQIR-DG controller algorithm is evaluated in three scenarios i) regulation and tracking problems, ii) disturbance rejection, and iii) impact of model uncertainty. Finally, a comparison of the proposed controller is performed with a PID controller and variants of the LQR controller. The PID controller parameters are presented in Table 5.

Table 5: PID controller parameters

Channel	K_p	K_i	K_d
roll	18	6	9
pitch	22	15	16

5.3.1. Investigating of the Regulation and Tracking Problems

The results of the proposed approach are presented for tracking the desired roll and pitch angles in Figures 8 and 9. Figure 8 (a) compares the desired and output signals, i.e., the Euler angles during the regulation problem. Moreover, Figure 8 (b) compares the desired square wave signals with a frequency of 0.02 Hz and an amplitude of 20 degrees with the output signals, when the quadrotor platform freely rotates around roll and pitch simultaneously. Figures 9 (a) and (b) show the rotational velocity commands of the quadrotor in the regulation and tracking problems, respectively. These results demonstrate that the roll and pitch angles are accurately controlled by the proposed approach.

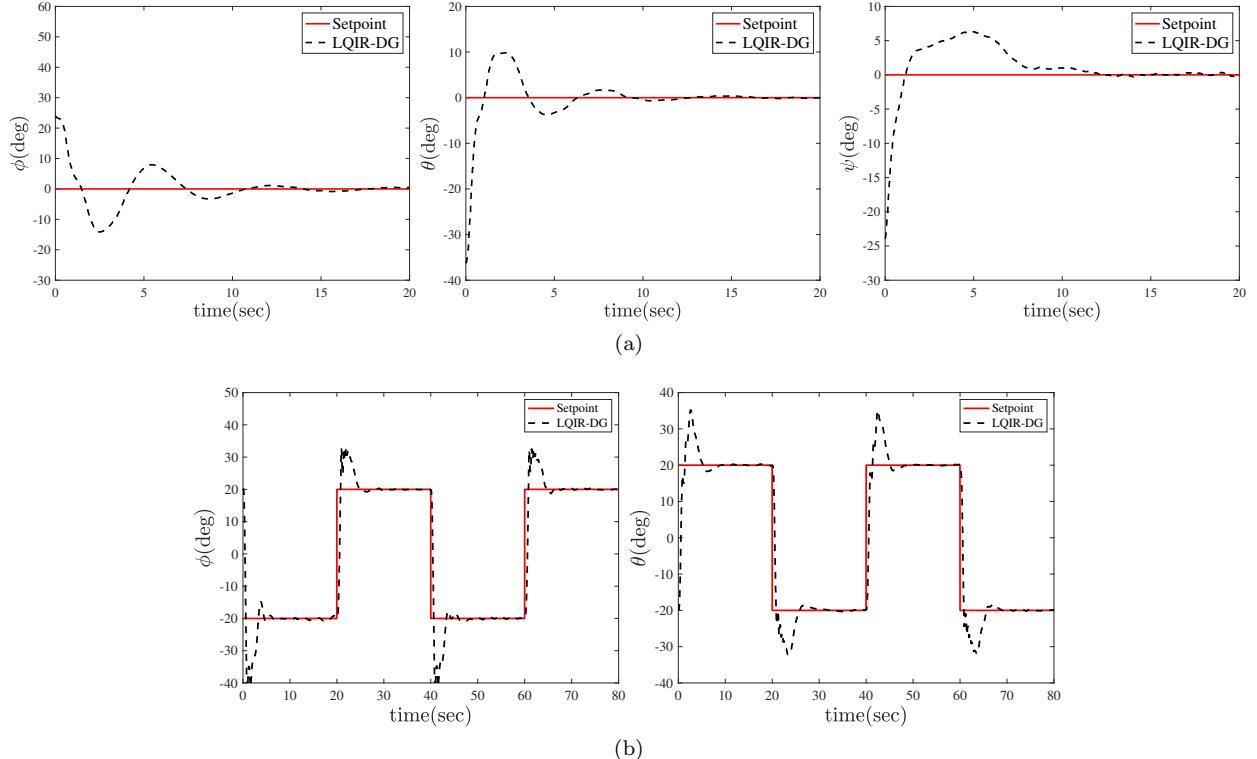


Figure 8: Comparison of actual roll and pitch angles with the desired values in (a) regulation and (b) tracking problems.

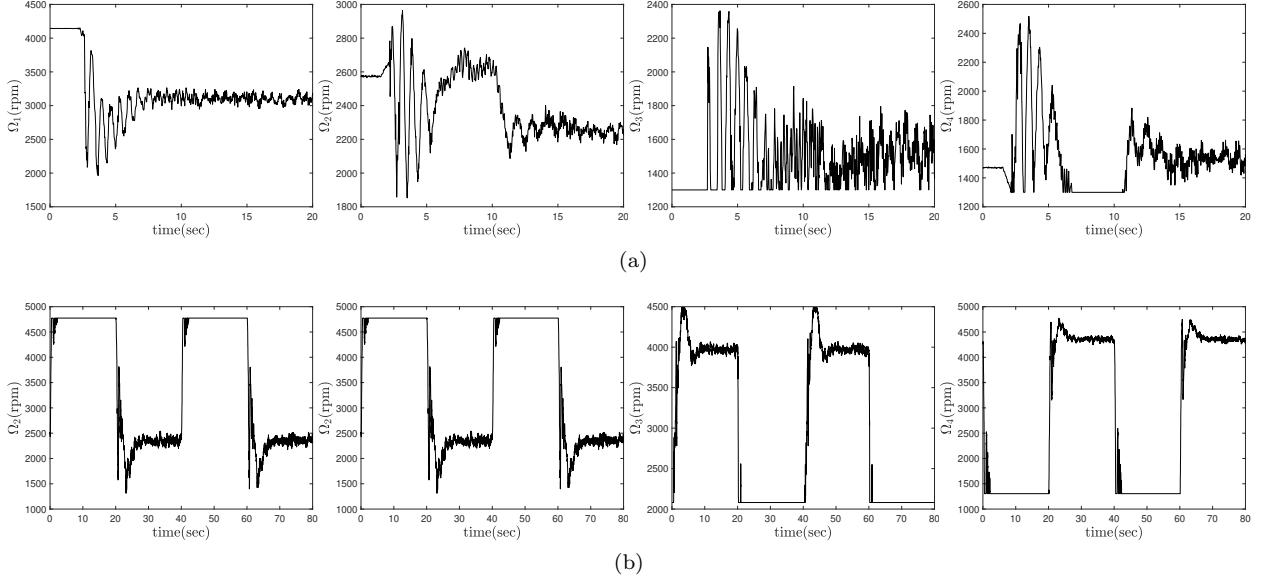


Figure 9: Rotational velocity commands in (a) regulation and (b) tracking problems.

5.3.2. Investigating the Disturbance Rejection

This section delves into the influence of input disturbance on the performance of the proposed controller. The input disturbance, denoted as d_{Ω_i} , represents a change in the command of the rotational velocity. The input disturbance is modeled as follows:

$$d_{\Omega_1} = d_{\Omega_2} = -d_{\Omega_3} = -d_{\Omega_4} = \begin{cases} 500 \text{ rpm} & 20 < t < 60 \\ 0 & \text{otherwise} \end{cases} \quad (40)$$

This disturbance profile simulates a sudden change in rotational velocity commands between time instants 20 seconds and 60 seconds. The aim is to assess the effectiveness of the proposed controller in stabilizing the quadrotor platform in the presence of such input disturbances.

Additionally, a comparative analysis is conducted to evaluate the performance of the proposed Linear Quadratic Integral Differential Game (LQIR-DG) controller against two other well-known disturbance rejection methods: Active Disturbance Rejection Control (ADRC) and Disturbance Observer-Based Control (DOBC).

Active Disturbance Rejection Control (ADRC): (ADRC) is a robust control method widely used to handle uncertainties and disturbances in dynamic systems. It relies on the concept of an extended state observer to estimate and compensate for disturbances, allowing for improved control performance. ADRC has gained popularity in various fields due to its ability to deal with disturbances in real-time and enhance system robustness.

In ADRC, an extended state observer estimates the total disturbances affecting the system, including external disturbances and model uncertainties. By actively rejecting these disturbances, ADRC ensures accurate tracking and control of the system. The control law in ADRC is designed to incorporate the estimated disturbances and adjust the control inputs accordingly to minimize the effects of disturbances on the system.

One of the key advantages of ADRC is its simplicity and ease of implementation, making it suitable for a wide range of control applications. The ability to handle uncertainties and disturbances in real-time allows ADRC to achieve better performance and stability compared to traditional control methods in challenging and dynamic environments.

Disturbance Observer-Based Control (DOBC): (DOBC) is robust control strategy used to estimate

and compensate for external disturbances affecting the system. DOBC employs a disturbance observer to accurately estimate the disturbances and uses this information to modify the control inputs accordingly.

DOBC is designed to address disturbances that cannot be directly measured but can significantly affect the system's behavior. By accurately estimating and compensating for these disturbances, DOBC improves the system's tracking and control performance, especially in the presence of uncertainties and external perturbations.

The implementation of DOBC involves designing an observer that accurately estimates the disturbances based on the system's dynamics and available measurements. The observer continuously updates its estimates, allowing the controller to adapt and respond to changing disturbance conditions.

Both ADRC and DOBC are effective tools for disturbance rejection and robust control in dynamic systems. Their ability to estimate and compensate for disturbances contributes to improved system stability and tracking performance, making them valuable options for various control applications, including quadrotor platforms.

Figure 10 illustrates the roll and pitch angles during the regulation problem when the input disturbance occurs. The results demonstrate the superior disturbance rejection capabilities of the proposed LQIR-DG controller compared to ADRC and DOBC. The LQIR-DG approach exhibits robust stabilization and superior tracking performance in the presence of input disturbances, further validating the effectiveness of our proposed method.

Overall, the comparative analysis highlights the remarkable disturbance rejection capabilities of the LQIR-DG controller, positioning it as a promising and impactful choice for controlling quadrotor platforms in real-world scenarios.

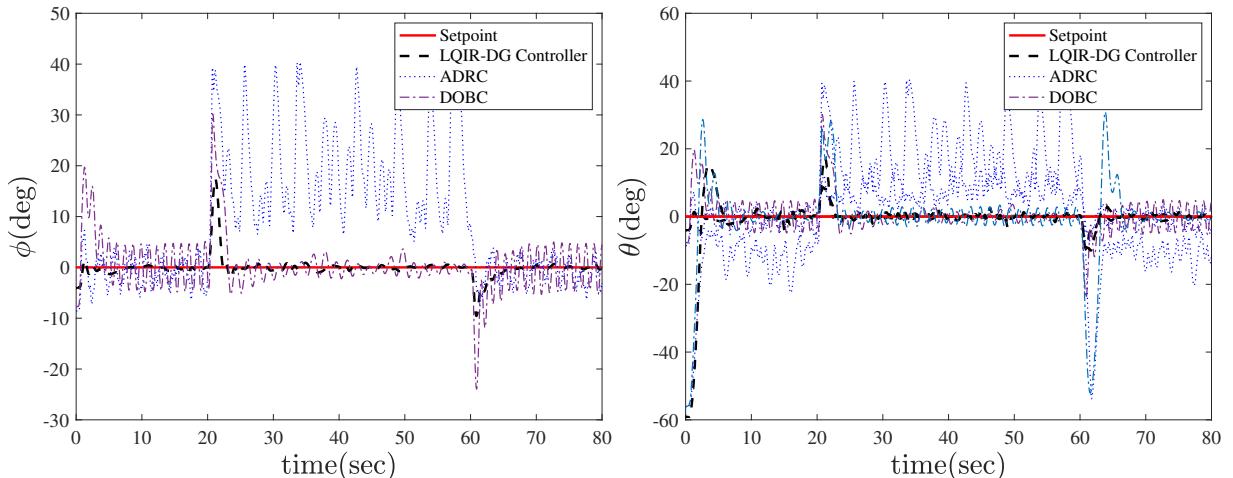


Figure 10: Comparison of actual roll and pitch angles with the desired, when the input disturbance occurs.

5.3.3. Investigating the Impact of Modeling Uncertainty

The effect of the modeling uncertainty is investigated on the performance of the proposed controller. To achieve this, 50 and 100 grams weights are added to the roll and pitch axes, respectively, as shown in Figure 11. Figure 12 (a) compares the desired and the actual roll angle and Figure 12 (b) shows the desired and the actual pitch angle, when the uncertainty of moments of inertia is present. Moreover, Figure 12 (c) shows the rotational velocity commands of the experimental platform, when the model uncertainty is applied. The implementation results show that the platform outputs converge to the desired values in the presence of the modeling uncertainty.

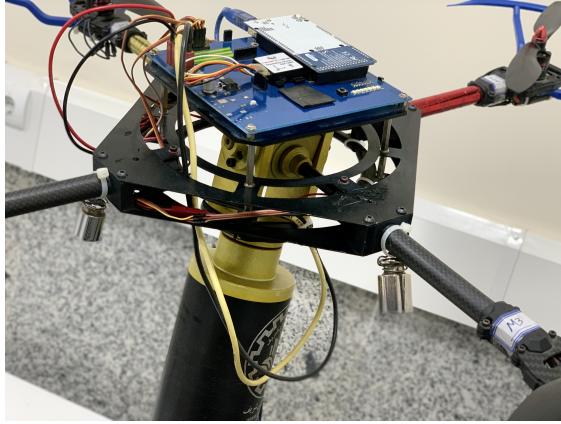


Figure 11: Quadrotor 3-DoF platform with added weights.

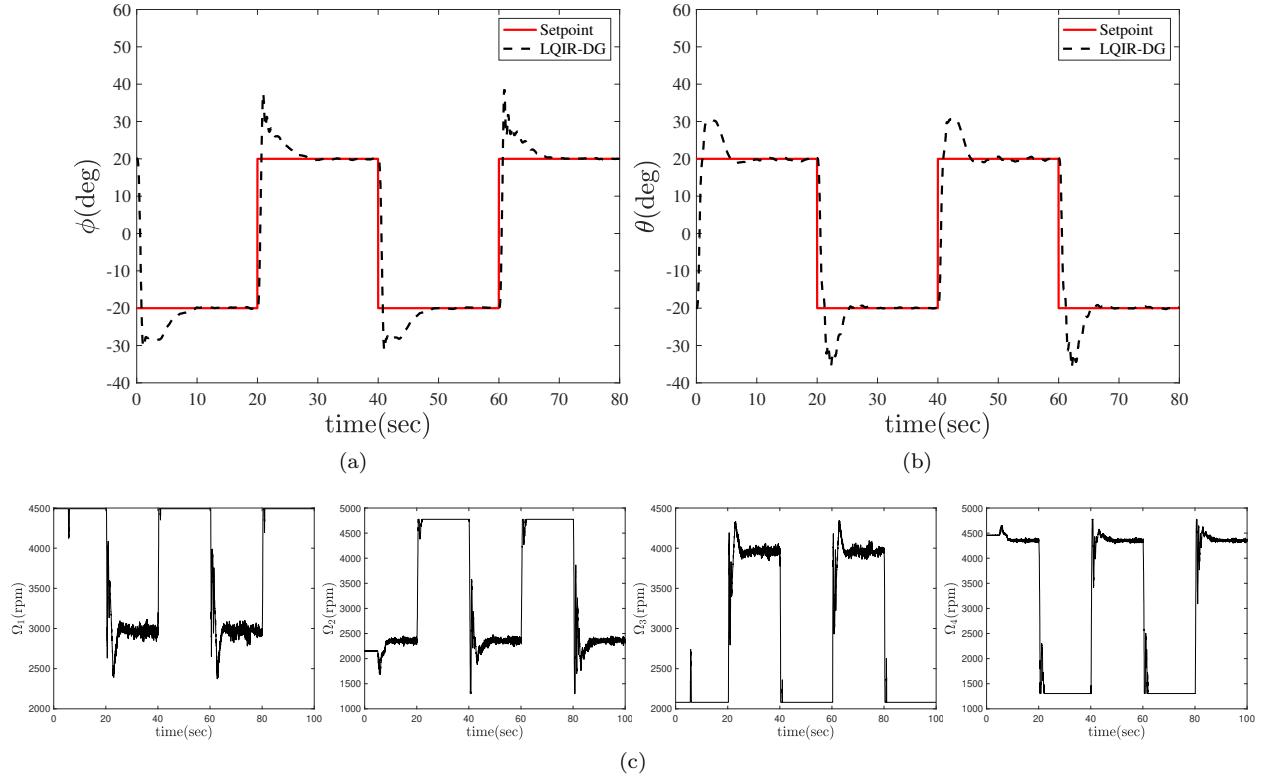


Figure 12: Comparison of actual roll and pitch angles with desired values, when the modeling uncertainty is present.

5.3.4. Comparison with the Control Strategies

Figure 13 compares the LQIR-DG controller performance with the PID controller and variant of the LQR strategies such as the LQR and LQIR. Moreover, the box plot of all controllers is plotted in Figure 14 for the cost function, introduced in equation (33). The median of Root Mean Square Error (RMSE) is shown in the crossline in the box plot. These results indicate that the proposed controller is able to provide rapid convergence and excellent transient response relative to other controllers for attitude control of the experimental platform.

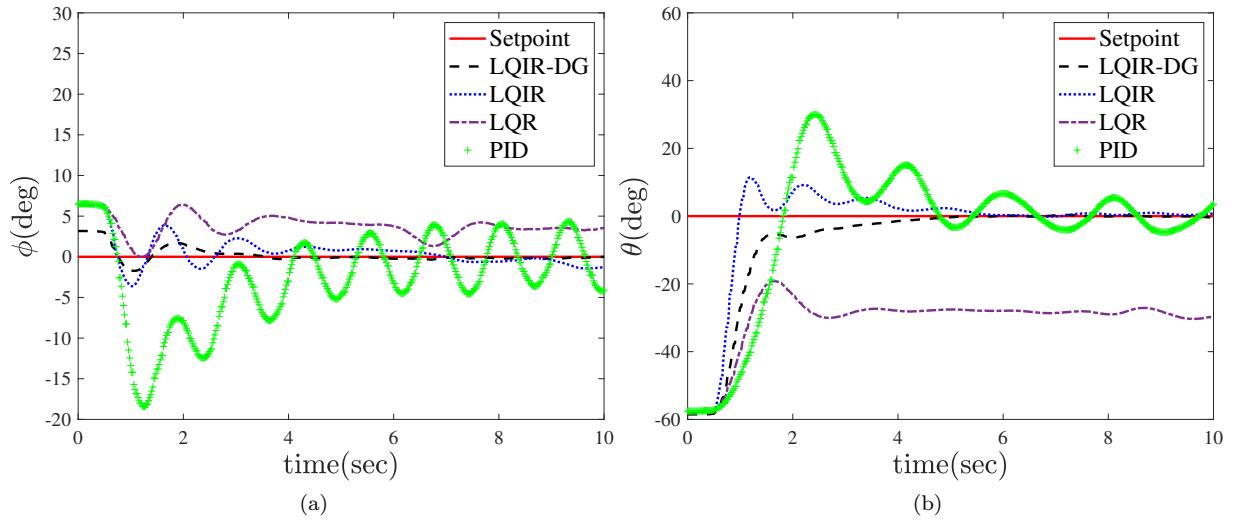


Figure 13: Comparison of LQIR-DG structure with the variant of LQR and PID in regulation problem: (a) roll angle (b) pitch angle.

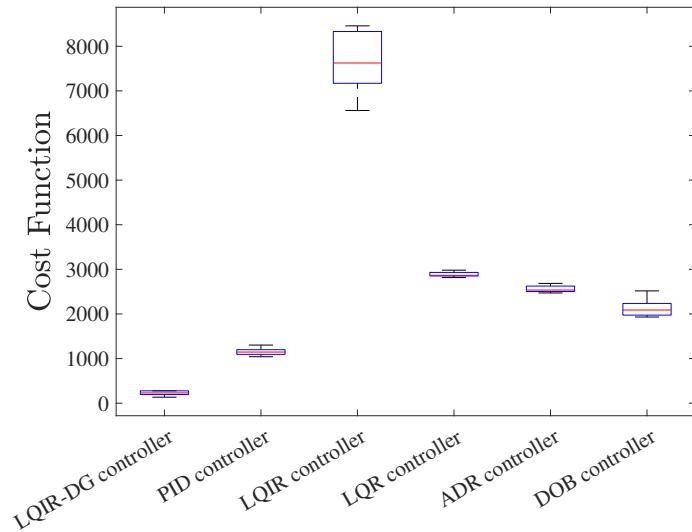


Figure 14: Box plot of LQIR-DG, LQR, LQIR, and PID controllers.

6. Conclusion

In this paper, the linear quadratic integral differential game approach, was used in real-time for attitude control of the platform quadrotor. For the implementation of the controller structure, an accurate dynamic model was considered for the experimental platform. Then, the model parameters were identified using the NSL method. For evaluation of the proposed method, the regulation and tracking proposed were successfully performed. Moreover, the ability of the proposed method was investigated in the rejection of the input disturbance and modeling error in the experimental platform. Finally, a comparison was also performed between the results of classical PID, LQR, and LQIR with the proposed method. The implementation results illustrated the excellent performance of the LQIR controller based on the game theory approach in attitude control for the quadrotor platform.

Here is a more detailed proof for the two player robust Nash equilibrium with coupled Riccati equations:

Proof:. [Two player coupled Riccati robust Nash equilibrium]

Consider the two player game between the system controller and disturbance.

The system dynamics are:

$$\dot{x} = Ax + Bu + Ew \quad (41)$$

The system controller minimizes the infinite horizon quadratic cost:

$$J = \int_0^\infty (x^\top Qx + u^\top Ru) dt \quad (42)$$

Subject to the system dynamics, where $Q \succeq 0, R \succ 0$.

The disturbance maximizes:

$$J_d = - \int_0^\infty w^\top Vw dt \quad (43)$$

Where $V \succ 0$ represents aversion to model uncertainty.

A robust Nash equilibrium satisfies:

$$J(u^*, w) \leq J(u^*, w^*) \leq J(u, w^*), \quad \forall u \quad (44)$$

$$J_d(u^*, w^*) \geq J_d(u^*, w) \geq J_d(u, w), \quad \forall w \quad (45)$$

That is, neither player can improve their cost by unilaterally deviating from their strategy.

Suppose the strategies are given by:

$$u^* = -R^{-1}B^\top P_1 x \quad (46)$$

$$w^* = V^{-1}E^\top P_2 x \quad (47)$$

Where P_1, P_2 satisfy the coupled Riccati equations:

$$0 = A^\top P_1 + P_1 A - P_1 B R^{-1} B^\top P_1 + P_1 E V^{-1} E^\top P_2 + Q \quad (48)$$

$$0 = A^\top P_2 + P_2 A + P_2 E V^{-1} E^\top P_2 - P_2 B R^{-1} B^\top P_1 \quad (49)$$

Substituting these strategies into the system dynamics gives the closed loop matrix:

$$A_{CL} = A - BR^{-1}B^\top P_1 + EV^{-1}E^\top P_2 \quad (50)$$

Take the derivative of VJ along the closed loop system dynamics:

$$\frac{dJ}{dt} = \dot{x}^\top VP_1x + x^\top P_1V\dot{x} = x^\top(Q + A_{CL}^\top P_1 + P_1A_{CL})x \quad (51)$$

For J to be non-increasing, $Q + A_{CL}^\top P_1 + P_1A_{CL} \preceq 0 \implies$ the Riccati equation for P_1 . Similarly, take the derivative of J_d :

$$\frac{dJ_d}{dt} = -x^\top P_2V\dot{x} - \dot{x}^\top VP_2x = x^\top(-Q - A_{CL}^\top P_2 - P_2A_{CL})x \quad (52)$$

For J_d to be non-decreasing, $-Q - A_{CL}^\top P_2 - P_2A_{CL} \succeq 0 \implies$ the Riccati equation for P_2 .

If P_1, P_2 satisfy the coupled Riccati equations and render A_{CL} stable, then u^*, w^* form a robust Nash equilibrium. \square

Here is a proof for the two player robust Nash equilibrium using Hamiltonian matrices:

Proof:. [Robust Nash Equilibrium with Hamiltonian Matrix]

Consider the two player uncertain dynamic game between the system controller and disturbance.

The system dynamics and cost functions are:

$$\begin{aligned} \dot{x} &= Ax + Bu + Ew \\ J &= \int_0^\infty x^\top Qx + u^\top Ru \\ J_d &= - \int_0^\infty w^\top Vw \end{aligned}$$

Where $Q \succeq 0, R \succ 0, V \succ 0$.

Suppose the strategies are given by:

$$\begin{aligned} u &= -R^{-1}B^\top P_1x \\ w &= V^{-1}E^\top P_2x \end{aligned}$$

Where P_1, P_2 satisfy coupled algebraic Riccati equations arising from the first-order optimality conditions. To derive these conditions, form the Hamiltonian:

$$\begin{aligned} H &= x^\top Qx + u^\top Ru + \lambda^\top(Ax + Bu + Ew) \\ &= x^\top(Q + A^\top\lambda)x + u^\top(R + B^\top\lambda)u + w^\top E^\top\lambda \end{aligned}$$

The optimality conditions are:

$$\begin{aligned} \frac{\partial H}{\partial u} &= 0 \Rightarrow R + B^\top\lambda = 0 \Rightarrow \lambda = -R^{-1}B^\top P_1x \\ \frac{\partial H}{\partial w} &= 0 \Rightarrow E^\top\lambda = -Vw = -V^{-1}E^\top P_2x \end{aligned}$$

Substitute these back into the Hamiltonian to obtain the coupled algebraic Riccati equations.

If P_1, P_2 satisfy these equations and render the closed-loop system stable, the strategies form a robust Nash equilibrium. \square

This shows the basics of using the Hamiltonian approach to derive the conditions for robust strategic optimization in the two player uncertain dynamic game.

Here is a detailed proof of the coupled algebraic Riccati equations for the linear quadratic inverse dynamic game: Let's consider a linear time-invariant system:

$$\dot{\mathbf{x}}_a = \mathbf{A}a\mathbf{x}_a + \mathbf{B}a_i\mathbf{u}_i + \mathbf{B}a_d\mathbf{d}_i \quad (53)$$

$$\mathbf{y}_i = \mathbf{C}a_i\mathbf{x}_a \quad (54)$$

Where $\mathbf{x}_a \in \mathbb{R}^n$ is the state, $\mathbf{u}_i \in \mathbb{R}^{m_i}$ is the control input of player i , $\mathbf{d}_i \in \mathbb{R}^{p_i}$ is the disturbance input of player i , $\mathbf{y}_i \in \mathbb{R}^{q_i}$ is the output for player i . The cost functions for the two players are:

$$J_i(\mathbf{u}_i, \mathbf{d}_i) = \int_0^\infty (\mathbf{x}_a^\top \mathbf{Q}_i \mathbf{x}_a + \mathbf{u}_i^\top \mathbf{R}_i \mathbf{u}_i) dt \quad (55)$$

$$J_{d_i}(\mathbf{u}_i, \mathbf{d}_i) = \int_0^\infty (\mathbf{x}_a^\top \mathbf{Q}_{d_i} \mathbf{x}_a - \mathbf{d}_i^\top \mathbf{R}_{d_i} \mathbf{d}_i) dt \quad (56)$$

Where $\mathbf{Q}_i, \mathbf{Q}_{d_i} \in \mathbb{R}^{n \times n}$ are positive semi-definite matrices and $\mathbf{R}_i, \mathbf{R}_{d_i} \in \mathbb{R}^{m_i \times m_i}, \mathbb{R}^{p_i \times p_i}$ are positive definite matrices. The optimal control law for player i is given by:

$$\mathbf{u}_i^* = -\mathbf{K}_i \mathbf{x}_a \quad (57)$$

Where $\mathbf{K}_i \in \mathbb{R}^{m_i \times n}$ is the optimal feedback gain matrix. The worst case disturbance for player i is given by:

$$\mathbf{d}_i^* = \mathbf{K}_{d_i} \mathbf{x}_a \quad (58)$$

Where $\mathbf{K}_{d_i} \in \mathbb{R}^{p_i \times n}$ is the optimal disturbance feedback matrix. The coupled algebraic Riccati equations to determine \mathbf{K}_i and \mathbf{K}_{d_i} are:

$$\mathbf{A}\mathbf{a}^\top \mathbf{P}_{a_{d_i}} + \mathbf{P}_{a_{d_i}} \mathbf{A}\mathbf{a} - \mathbf{P}_{a_{d_i}} \mathbf{B}\mathbf{a}_i \mathbf{R}_i^{-1} \mathbf{B}\mathbf{a}_i^\top \mathbf{P}_{a_{d_i}} + \mathbf{Q}_i + \mathbf{P}_{a_{d_i}} \mathbf{B}\mathbf{a}_i \mathbf{R}_{d_i}^{-1} \mathbf{B}\mathbf{a}_{d_i}^\top \mathbf{P}_{a_{d_i}} = 0 \quad (59)$$

$$\mathbf{A}\mathbf{a}^\top \mathbf{P}_{a_i} + \mathbf{P}_{a_i} \mathbf{A}\mathbf{a} - \mathbf{P}_{a_i} \mathbf{B}\mathbf{a}_{d_i} \mathbf{R}_i^{-1} \mathbf{B}\mathbf{a}_{d_i}^\top \mathbf{P}_{a_{d_i}} + \mathbf{Q}_i + \mathbf{P}_{a_i} \mathbf{B}\mathbf{a}_i \mathbf{R}_i^{-1} \mathbf{B}\mathbf{a}_i^\top \mathbf{P}_{a_i} = 0 \quad (60)$$

The optimal gain matrices are then:

$$\mathbf{K}_{d_i} = \mathbf{R}_{d_i}^{-1} \mathbf{B}\mathbf{a}_{d_i}^\top \mathbf{P}_{a_{d_i}} \quad (61)$$

$$\mathbf{K}_i = \mathbf{R}_i^{-1} \mathbf{B}\mathbf{a}_i^\top \mathbf{P}_{a_i} \quad (62)$$

Therefore, the coupled algebraic Riccati equations arise from the necessary conditions for optimality of the linear quadratic inverse dynamic game problem. Solving these equations yields the optimal feedback control and disturbance gains.

Here is a detailed derivation of the coupled algebraic Riccati equations for the linear quadratic inverse dynamic game: Consider the linear time-invariant system:

$$\dot{\mathbf{x}}_a = \mathbf{A}a\mathbf{x}_a + \mathbf{B}a_i\mathbf{u}_i + \mathbf{B}a_d\mathbf{d}_i \quad \mathbf{y}_i = \mathbf{C}a_i\mathbf{x}_a \quad (63)$$

The cost functions for the two players are:

$$J_i(\mathbf{u}_i, \mathbf{d}_i) = \int_0^\infty (\mathbf{x}_a^\top \mathbf{Q}_i \mathbf{x}_a + \mathbf{u}_i^\top \mathbf{R}_i \mathbf{u}_i) dt \quad J_{d_i}(\mathbf{u}_i, \mathbf{d}_i) = \int_0^\infty (\mathbf{x}_a^\top \mathbf{Q}_{d_i} \mathbf{x}_a - \mathbf{d}_i^\top \mathbf{R}_{d_i} \mathbf{d}_i) dt \quad (64)$$

The Hamiltonian for player i is:

$$H_i = \mathbf{x}_a^\top \mathbf{Q}_i \mathbf{x}_a + \mathbf{u}_i^\top \mathbf{R}_i \mathbf{u}_i + \lambda^\top (\mathbf{A}a\mathbf{x}_a + \mathbf{B}a_i\mathbf{u}_i + \mathbf{B}a_d\mathbf{d}_i) \quad (65)$$

Where $\lambda \in \mathbb{R}^n$ is the costate vector. The optimality conditions are:

$$\frac{\partial H_i}{\partial \mathbf{u}_i} = 0 \Rightarrow \mathbf{u}_i = -\mathbf{R}i^{-1}\mathbf{B}\mathbf{a}_i^\top \boldsymbol{\lambda} \quad \frac{d\boldsymbol{\lambda}}{dt} = -\frac{\partial H_i}{\partial \mathbf{x}_a} \Rightarrow \dot{\boldsymbol{\lambda}} = -\mathbf{Q}_i \mathbf{x}_a - \mathbf{A}_a^\top \boldsymbol{\lambda} \quad (66)$$

Assuming steady-state optimal solution:

$$\mathbf{u}_i = -\mathbf{K}_i \mathbf{x}_a \boldsymbol{\lambda} = \mathbf{P}a_i \mathbf{x}_a \quad (67)$$

Where $\mathbf{K}_i \in \mathbb{R}^{m_i \times n}$ and $\mathbf{P}a_i \in \mathbb{R}^{n \times n}$ are constant gain matrices. Substituting these into the costate equation:

$$0 = -\mathbf{Q}_i \mathbf{x}_a - \mathbf{A}a^\top \mathbf{P}a_i \mathbf{x}_a \Rightarrow 0 = -\mathbf{Q}_i - \mathbf{A}a^\top \mathbf{P}a_i - \mathbf{P}a_i \mathbf{A}_a \quad (68)$$

Pre-multiplying by $\mathbf{P}a_i^{-1}$:

$$0 = -\mathbf{P}a_i^{-1} \mathbf{Q}_i - \mathbf{A}a^\top - \mathbf{P}a_i \mathbf{A}a \mathbf{P}a_i^{-1} 0 = -\mathbf{P}a_i^{-1} \mathbf{Q}_i - \mathbf{A}a^\top - \mathbf{P}a_i \mathbf{A}a + \mathbf{P}a_i \mathbf{B}a_i \mathbf{R}i^{-1} \mathbf{B}a_i^\top \quad (69)$$

Where we have substituted $\mathbf{K}_i = \mathbf{R}i^{-1} \mathbf{B}a_i^\top \mathbf{P}a_i$. This gives the first Riccati equation:

$$\mathbf{A}a^\top \mathbf{P}a_i + \mathbf{P}a_i \mathbf{A}a - \mathbf{P}a_i \mathbf{B}a_i \mathbf{R}i^{-1} \mathbf{B}a_i^\top \mathbf{P}a_i + \mathbf{Q}_i = 0 \quad (70)$$

Repeating the steps above for the maximizing disturbance player gives:

$$\mathbf{A}a^\top \mathbf{P}a_{d_i} + \mathbf{P}a_{d_i} \mathbf{A}a - \mathbf{P}a_{d_i} \mathbf{B}a_i \mathbf{R}d_i^{-1} \mathbf{B}a_{d_i}^\top \mathbf{P}a_{d_i} + \mathbf{Q}_{d_i} = 0 \quad (71)$$

Thus, we obtain two coupled algebraic Riccati equations that must be solved to find the optimal control and disturbance feedback gains. The cross-coupling occurs through the state dynamics.

References

- [1] Abdul Salam, A., Ibraheem, I., 2019. 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.
- [2] Aboudonia, A., El-Badawy, A., Rashad, R., 2016. Disturbance observer-based feedback linearization control of an unmanned quadrotor helicopter. *Proceedings of the Institution of Mechanical Engineers Part I Journal of Systems and Control Engineering* 230. doi:10.1177/0959651816656951.
- [3] Ahmad, F., Kumar, P., Bhandari, A., Patil, P.P., 2020. Simulation of the quadcopter dynamics with lqr based control. *Materials Today: Proceedings* 24, 326–332. URL: <https://www.sciencedirect.com/science/article/pii/S2214785320329047>, doi:<https://doi.org/10.1016/j.matpr.2020.04.282>. international Conference on Advances in Materials and Manufacturing Applications, IConAMMA 2018, 16th -18th August, 2018, India.
- [4] Anjali, B., A., V., J L, N., 2016. Simulation and analysis of integral lqr controller for inner control loop design of a fixed wing micro aerial vehicle (mav). *Procedia Technology* 25, 76–83. doi:10.1016/j.protcy.2016.08.083.
- [5] Bolandi, H., Rezaei, M., Mohsenipour, R., Nemati, H., Smailzadeh, S., 2013. Attitude control of a quadrotor with optimized pid controller. *Intelligent Control and Automation* 04, 342–349. doi:10.4236/ica.2013.43040.
- [6] Bouabdallah, S., 2007. Design and control of quadrotors with application to autonomous flying. URL: <https://api.semanticscholar.org/CorpusID:108233951>.
- [7] Bouabdallah, S., Siegwart, R., 2007. Full control of a quadrotor, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 153–158. doi:10.1109/IROS.2007.4399042.
- [8] Chara, K., Yassine, A., Srairi, F., Mokhtari, K., 2022. A robust synergetic controller for quadrotor obstacle avoidance using b閦ier curve versus b-spline trajectory generation. *Intelligent Service Robotics* 15. doi:10.1007/s11370-021-00408-0.
- [9] Chen, L., Liu, Z., Gao, H., Wang, G., 2022a. Robust adaptive recursive sliding mode attitude control for a quadrotor with unknown disturbances. *ISA Transactions* 122, 114–125. URL: <https://www.sciencedirect.com/science/article/pii/S0019057821002433>, doi:<https://doi.org/10.1016/j.isatra.2021.04.046>.
- [10] Chen, S., Li, Y., Lou, Y., Lin, K., Wu, X., 2022b. Learning real-time dynamic responsive gap-traversing policy for quadrotors with safety-aware exploration. *IEEE Transactions on Intelligent Vehicles* , 1–14doi:10.1109/TIV.2022.3229723.
- [11] Engwerda, J., 2006. Linear Quadratic Games: An Overview. WorkingPaper. Macroeconomics. Subsequently published in *Advances in Dynamic Games and their Applications* (book), 2009 Pagination: 32.
- [12] Foudeh, H.A., Luk, P., Whidborne, J., 2020. Application of norm optimal iterative learning control to quadrotor unmanned aerial vehicle for monitoring overhead power system. *Energies* 13. URL: <https://www.mdpi.com/1996-1073/13/12/3223>, doi:10.3390/en13123223.

- [13] Glida, H.E., Chelihi, A., Abdou, L., Sentouh, C., Perozzi, G., 2022. Trajectory tracking control of a coaxial rotor drone: Timedelay estimation-based optimal modelfree fuzzy logic approach. *ISA Transactions* URL: <https://www.sciencedirect.com/science/article/pii/S0019057822006462>, doi:<https://doi.org/10.1016/j.isatra.2022.12.015>.
- [14] Hwangbo, J., Sa, I., Siegwart, R., Hutter, M., 2017. Control of a quadrotor with reinforcement learning. *IEEE Robotics and Automation Letters* 2, 2096–2103. URL: <https://doi.org/10.1109/LRA.2017.2720851>, doi:[10.1109/LRA.2017.2720851](https://doi.org/10.1109/LRA.2017.2720851).
- [15] Labbadi, M., Cherkaoui, M., 2020. Robust adaptive nonsingular fast terminal sliding-mode tracking control for an uncertain quadrotor uav subjected to disturbances. *ISA Transactions* 99, 290–304. URL: <https://www.sciencedirect.com/science/article/pii/S0019057819304665>, doi:<https://doi.org/10.1016/j.isatra.2019.10.012>.
- [16] Lin, X., Liu, J., Yu, Y., Sun, C., 2020. Event-triggered reinforcement learning control for the quadrotor uav with actuator saturation. *Neurocomputing* 415, 135–145. URL: <https://www.sciencedirect.com/science/article/pii/S0925231220311504>, doi:<https://doi.org/10.1016/j.neucom.2020.07.042>.
- [17] Mofid, O., Mobayen, S., Zhang, C., Esakki, B., 2022. Desired tracking of delayed quadrotor uav under model uncertainty and wind disturbance using adaptive super twisting terminal sliding mode control. *ISA Transactions* 123. URL: <https://www.sciencedirect.com/science/article/pii/S0019057821003086>, doi:<https://doi.org/10.1016/j.isatra.2021.06.002>.
- [18] Nobahari, H., Baniasad, A., Sharifi, A., 2022. Linear quadratic integral differential game applied to the real-time control of a quadrotor experimental setup, in: 2022 10th RSI International Conference on Robotics and Mechatronics (ICRoM), pp. 578–583. doi:[10.1109/ICRoM57054.2022.10025263](https://doi.org/10.1109/ICRoM57054.2022.10025263).
- [19] Pourtakdoust, S.H., Nobahari, H., 2004. An extension of ant colony system to continuous optimization problems, in: Dorigo, M., Birattari, M., Blum, C., Gambardella, L.M., Mondada, F., Stützle, T. (Eds.), *Ant Colony Optimization and Swarm Intelligence*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 294–301.
- [20] Rekabi, F., Shirazi, F.A., Sadigh, M.J., 2020. Distributed nonlinear h_∞ control algorithm for multi-agent quadrotor formation flying. *ISA Transactions* 96, 81–94. URL: <https://www.sciencedirect.com/science/article/pii/S001905781930165X>, doi:<https://doi.org/10.1016/j.isatra.2019.04.036>.
- [21] Santos, D.A., Lagoa, C.M., 2022. Wayset-based guidance of multirotor aerial vehicles using robust tube-based model predictive control. *ISA Transactions* 128, 123–135. URL: <https://www.sciencedirect.com/science/article/pii/S0019057821006182>, doi:<https://doi.org/10.1016/j.isatra.2021.12.002>.
- [22] Srinivasarao, G., Samantaray, A.K., Ghoshal, S.K., 2022. Cascaded adaptive integral backstepping sliding mode and super-twisting controller for twin rotor system using bond graph model. *ISA Transactions* 130, 516–532. URL: <https://www.sciencedirect.com/science/article/pii/S0019057822001458>, doi:<https://doi.org/10.1016/j.isatra.2022.03.023>.
- [23] Wang, H., Li, Z., Xiong, H., Nian, X., 2019. Robust h_∞ attitude tracking control of a quadrotor uav on $so(3)$ via variation-based linearization and interval matrix approach. *ISA Transactions* 87, 10–16. URL: <https://www.sciencedirect.com/science/article/pii/S0019057818304518>, doi:<https://doi.org/10.1016/j.isatra.2018.11.015>.
- [24] Wang, Y., Liu, W., Liu, J., Sun, C., 2023. Cooperative usv-uav marine search and rescue with visual navigation and reinforcement learning-based control. *ISA Transactions* 137, 222–235. URL: <https://www.sciencedirect.com/science/article/pii/S0019057823000071>, doi:<https://doi.org/10.1016/j.isatra.2023.01.007>.
- [25] Wu, X., Xiao, B., Qu, Y., 2022. Modeling and sliding mode-based attitude tracking control of a quadrotor uav with time-varying mass. *ISA Transactions* 124, 436–443. URL: <https://www.sciencedirect.com/science/article/pii/S0019057819303544>, doi:<https://doi.org/10.1016/j.isatra.2019.08.017>.
- [26] Xie, T., Xian, B., Gu, X., 2023. Fixed-time convergence attitude control for a tilt trirotor unmanned aerial vehicle based on reinforcement learning. *ISA Transactions* 132, 477–489. URL: <https://www.sciencedirect.com/science/article/pii/S0019057822003111>, doi:<https://doi.org/10.1016/j.isatra.2022.06.006>.
- [27] Yan, D., Zhang, W., Chen, H., Shi, J., 2023. Robust control strategy for multi-uavs system using mpc combined with kalman-consensus filter and disturbance observer. *ISA Transactions* 135, 35–51. URL: <https://www.sciencedirect.com/science/article/pii/S0019057822004797>, doi:<https://doi.org/10.1016/j.isatra.2022.09.021>.
- [28] Zhou, X., Li, X., 2021. Trajectory tracking control for electro-optical tracking system based on fractional-order sliding mode controller with super-twisting extended state observer. *ISA Transactions* 117, 85–95. URL: <https://www.sciencedirect.com/science/article/pii/S0019057821000732>, doi:<https://doi.org/10.1016/j.isatra.2021.01.062>.
- [29] Zulu, A., John, S., 2014. A review of control algorithms for autonomous quadrotors. *Open Journal of Applied Sciences* 04, 547–556. doi:[10.4236/ojapps.2014.414053](https://doi.org/10.4236/ojapps.2014.414053).

Appendix A. Linearization

The linearized dynamics are: In the previous section, it was shown that equations (21) to (25) represent nonlinear dynamics. To obtain a linearized model, these equations can be rewritten as follows:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{cases} \dot{x}_1 &= \Gamma_1 x_1 x_3 - \Gamma_2 x_3 x_5 + \Gamma_3 u_{\text{roll}} + \Gamma_4 u_{\text{yaw}} + \Gamma_3 d_{\text{roll}} \\ \dot{x}_2 &= x_1 + (x_3 \sin(x_2) + x_3 \cos(x_2)) \tan(x_4) \\ \dot{x}_3 &= \Gamma_5 x_1 x_5 - \Gamma_6 (x_1^2 - x_5^2) + \frac{1}{J_y} u_{\text{pitch}} + \frac{1}{J_y} d_{\text{pitch}} \\ \dot{x}_4 &= x_3 \cos(x_4) - x_5 \sin(x_2) \\ \dot{x}_5 &= \Gamma_7 x_1 x_3 - \Gamma_1 x_3 x_5 + \Gamma_4 u_{\text{roll}} + \Gamma_8 u_{\text{yaw}} + \Gamma_8 d_{\text{yaw}} \\ \dot{x}_6 &= (x_3 \sin(x_4) + x_5 \cos(x_2)) / \cos(x_4) \end{cases} \quad (\text{A.1})$$

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad (\text{A.2})$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector representing the system's internal states, $\mathbf{u} \in \mathbb{R}^m$ is the input vector, and $\mathbf{f}(\mathbf{x}, \mathbf{u})$ is the nonlinear function governing the state dynamics. To linearize the system around the equilibrium points ($\mathbf{x}_e^* = 0$ and $\mathbf{u}_e^* = 0$), small perturbations $\Delta \mathbf{x}$ and $\Delta \mathbf{u}$ are used:

$$\mathbf{x} = \mathbf{x}_0 + \Delta \mathbf{x}, \quad (\text{A.3})$$

$$\mathbf{u} = \mathbf{u}_0 + \Delta \mathbf{u}. \quad (\text{A.4})$$

Then, the linearized state-space representation can be expressed as:

$$\Delta \dot{\mathbf{x}} = \mathbf{A} \Delta \mathbf{x} + \mathbf{B} \Delta \mathbf{u} \quad (\text{A.5})$$

where $\mathbf{A} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\mathbf{x}_e^*, \mathbf{u}_e^*}$ is the state matrix, $\mathbf{B} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{\mathbf{x}_e^*, \mathbf{u}_e^*}$ is the input matrix.

$$\left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\mathbf{x}_e^*, \mathbf{u}_e^*} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{\mathbf{x}_e^*, \mathbf{u}_e^*} = \begin{bmatrix} \Gamma_3 & 0 & \Gamma_4 \\ 0 & 0 & 0 \\ 0 & \frac{1}{J_y} & 0 \\ 0 & 0 & 0 \\ \Gamma_4 & 0 & \Gamma_8 \\ 0 & 0 & 0 \end{bmatrix} \quad (\text{A.6})$$