

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, 6]. 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 [9], Feedback Linearization (FBL) [2], Sliding Mode Control (SMC) [18, 29, 20, 10, 26, 32] have been utilized to regulate the quadrotor's Euler angles (roll, pitch, and yaw angles) intelligently.

Intelligent control approaches, including reinforcement learning [16, 19, 30, 28], iterative learning [14], machine learning [11], and fuzzy logic [15], have also been employed to control the attitude of the quadrotor.

Optimal control strategies, such as Linear Quadratic Gaussian (LQG) [33], Linear Quadratic Regulator (LQR) [4], Linear Quadratic Integral Regulator (LQIR) [5], and Model Predictive Controller (MPC) [31, 24], have been applied to generate optimal control commands for the quadrotor.

In the domain of robust control, techniques like H_∞ control [27, 23], μ -synthesis, and Linear Quadratic Regulator Differential Game (LQR-DG) [21] 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.

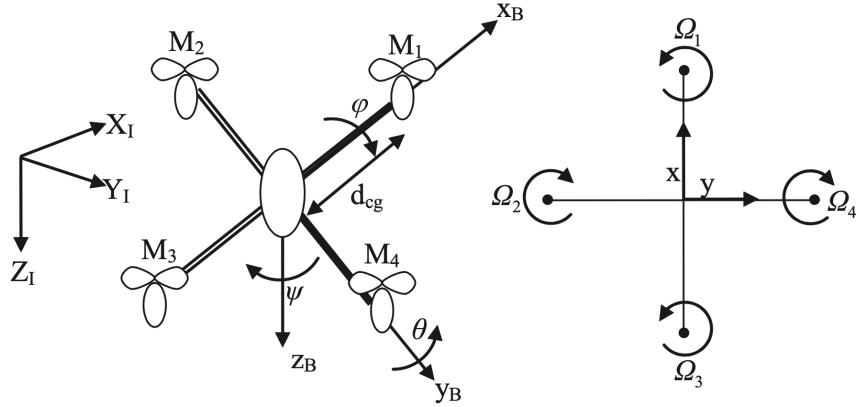


Figure 3: Quadrotor configuration.

3.2. Dynamic Modeling of the Quadrotor Platform

Here, according to Newton-Euler, the model of the quadrotor platform is presented as follows [8, 7]:

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + \Gamma_3 bd_{cg}(\Omega_{c,2}^2 - \Omega_{c,4}^2) + \Gamma_4 d(\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2) + \Gamma_5 \Omega_{c,r} + \Gamma_3 d_{roll} + \Gamma_4 d_{yaw} \quad (1)$$

$$\dot{q} = \Gamma_6 pr - \Gamma_7(p^2 - r^2) + \Gamma_8 bd_{cg}(\Omega_{c,1}^2 - \Omega_{c,3}^2) + \Gamma_9 \Omega_{c,r} + \Gamma_8 d_{pitch} \quad (2)$$

$$\dot{r} = \Gamma_{10} pq - \Gamma_1 qr + \Gamma_{11}(\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2) + \Gamma_4 bd_{cg}(\Omega_{c,2}^2 - \Omega_{c,4}^2) + \Gamma_{11} d_{roll} + \Gamma_4 d_{yaw} \quad (3)$$

In the above equations, Γ_i ($i = 1, \dots, 11$) is defined as

$$\begin{aligned} \Gamma_1 &= \frac{I_{xz}(I_{xx} - I_{yy} + I_{zz})}{\Gamma}, & \Gamma_2 &= \frac{I_{zz}(I_{zz} - I_{yy}) + I_{xz}^2}{\Gamma}, & \Gamma_3 &= \frac{I_{zz}}{\Gamma}, & \Gamma_4 &= \frac{I_{xz}}{\Gamma} \\ \Gamma_5 &= \frac{I_{rotor}}{I_{xx}}, & \Gamma_6 &= \frac{I_{zz} - I_{xx}}{I_{yy}}, & \Gamma_7 &= \frac{I_{xz}}{I_{yy}}, & \Gamma_8 &= \frac{1}{I_{yy}} \\ \Gamma_9 &= \frac{I_{rotor}}{I_{yy}}, & \Gamma_{10} &= \frac{(I_{xx} - I_{yy}) + I_{xz}^2}{\Gamma}, & \Gamma_{11} &= \frac{I_{xx}}{\Gamma} \end{aligned} \quad (4)$$

Moreover $\Gamma = J_x J_z - J_{xy}^2$. where $\Omega_{c,i}$ ($i = 1, 2, 3, 4$) is the rotational velocity, computed as

$$\Omega_{c,1}^2 = \Omega_{mean}^2 + \frac{1}{2bd_{cg}}u_{pitch} + \frac{1}{4d}u_{yaw} \quad (5)$$

$$\Omega_{c,2}^2 = \Omega_{mean}^2 + \frac{1}{2bd_{cg}}u_{roll} - \frac{1}{4d}u_{yaw} \quad (6)$$

$$\Omega_{c,3}^2 = \Omega_{mean}^2 - \frac{1}{2bd_{cg}}u_{pitch} + \frac{1}{4d}u_{yaw} \quad (7)$$

$$\Omega_{c,4}^2 = \Omega_{mean}^2 - \frac{1}{2bd_{cg}}u_{roll} - \frac{1}{4d}u_{yaw} \quad (8)$$

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:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (9)$$

3.3. State-Space Formulation

By defining $\mathbf{x}_{roll} = [x_1 \ x_2]^T = [p \ \phi]^T$, $\mathbf{x}_{pitch} = [x_3 \ x_4]^T = [q \ \theta]^T$, and $\mathbf{x}_{yaw} = [x_5 \ x_6]^T = [r \ \psi]^T$, as well as by considering the control inputs as

$$\mathbf{u} = [u_{roll} \ u_{pitch} \ u_{yaw}] = [bd_{cg}(\Omega_{c,2}^2 - \Omega_{c,4}^2) \ bd_{cg}(\Omega_{c,1}^2 - \Omega_{c,3}^2) \ bd_{cg}(\Omega_{c,1}^2 - \Omega_{c,2}^2 + \Omega_{c,3}^2 - \Omega_{c,4}^2)]$$

The nonlinear model of the quadrotor platform in the state-space form $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ is presented as follows:

$$f_1 = \dot{x}_1 = \Gamma_1 x_1 x_3 - \Gamma_2 x_3 x_5 + \Gamma_3(u_{roll} + d_{roll}) + \Gamma_4 d(u_{yaw} + d_{yaw}) + \Gamma_5 \Omega_{c,r} \quad (10)$$

$$f_2 = \dot{x}_2 = x_1 + (x_3 \sin(x_2) + x_3 \cos(x_2)) \tan(x_4) \quad (11)$$

$$f_3 = \dot{x}_3 = \Gamma_6 x_1 x_5 - \Gamma_7(x_1^2 - x_5^2) + \Gamma_8(u_{pitch} + d_{pitch}) - \Gamma_9 \Omega_{c,r} \quad (12)$$

$$f_4 = \dot{x}_4 = x_3 \cos(x_4) - x_5 \sin(x_2) \quad (13)$$

$$f_5 = \dot{x}_5 = \Gamma_{10} x_1 x_3 - \Gamma_1 x_3 x_5 + \Gamma_{11}(u_{yaw} + d_{yaw}) + \Gamma_4(u_{roll} + d_{roll}) \quad (14)$$

$$f_6 = \dot{x}_6 = \frac{x_3 \sin(x_4) + x_5 \cos(x_2)}{\cos(x_4)} \quad (15)$$

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

$$\mathbf{z} = [p \ q \ r \ \phi \ \theta \ \psi]^T + \boldsymbol{\nu} \quad (16)$$

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

3.4. Linear Model

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

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} (\mathbf{u} + \mathbf{d}) \quad (17)$$

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

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{\text{roll}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{\text{pitch}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{\text{yaw}} \end{bmatrix} \quad (18)$$

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

$$\mathbf{B} = \begin{bmatrix} \Gamma_3 & 0 & \Gamma_4 \\ 0 & 0 & 0 \\ 0 & \Gamma_8 & 0 \\ 0 & 0 & 0 \\ \Gamma_4 & 0 & \Gamma_{11} \\ 0 & 0 & 0 \end{bmatrix} \quad (19)$$

3.5. Identification of the Platform Parameters

In this section, the Nonlinear Least Squares (NLS) algorithm is utilized for estimating the model parameters ($\boldsymbol{\Gamma}$) 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 $\boldsymbol{\Gamma}$ by minimizing a cost function, defined as

$$\min_{\boldsymbol{\Gamma}} (\| e(\boldsymbol{\Gamma}) \|^2) = \min_{\boldsymbol{\Gamma}} \left(\sum_{j=1}^n (\mathbf{z}_j - \tilde{\mathbf{z}}_j)(\mathbf{z}_j - \tilde{\mathbf{z}}_j)^T \right) \quad (20)$$

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 $\boldsymbol{\Gamma}$, the optimization process performs until convergence is achieved. The structure of the identification approach is illustrated in figure 4.

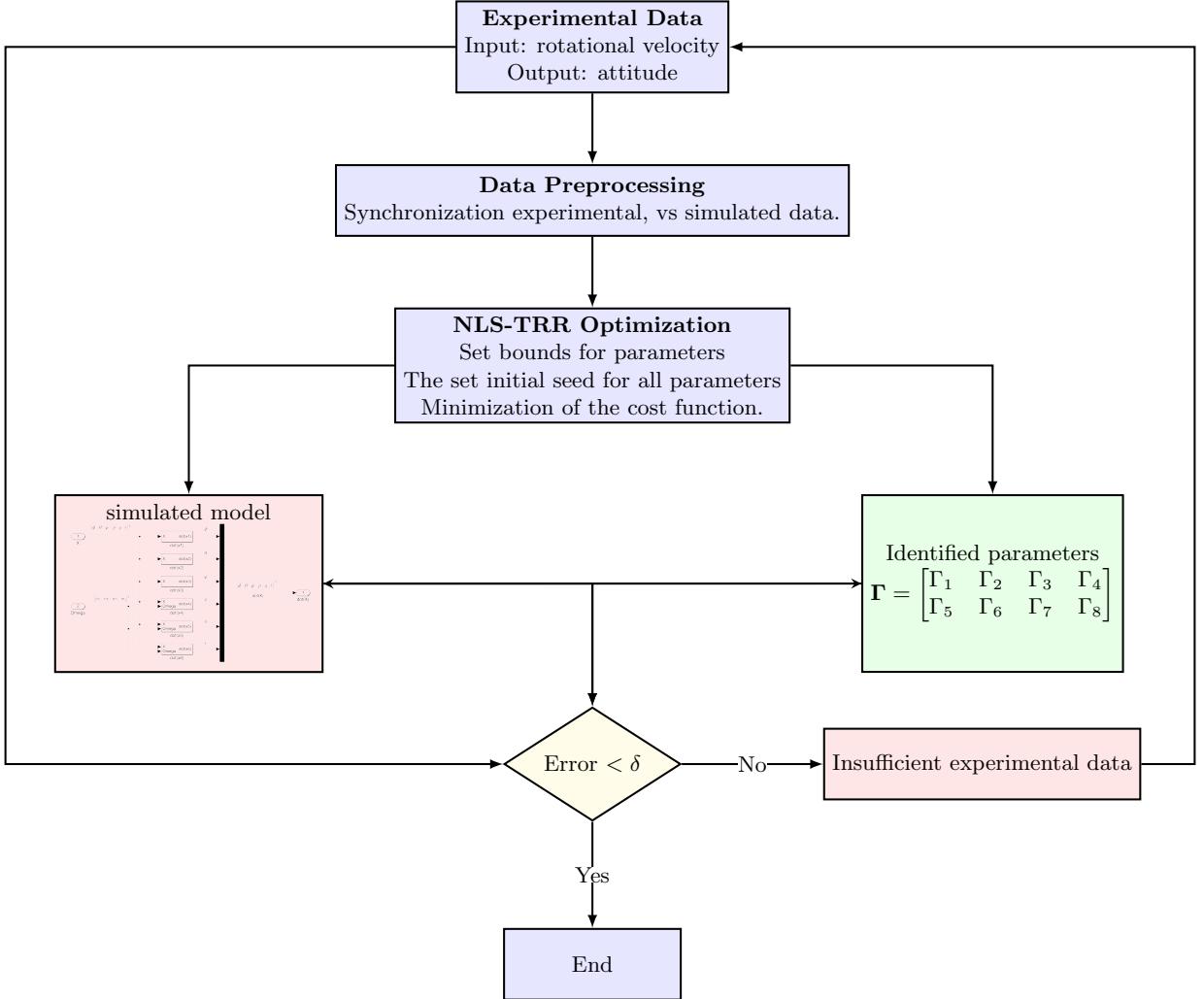


Figure 4: Structure of TRRLS identification approach.

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

$$\dot{\mathbf{x}}_a = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{x}_a + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} (\mathbf{u} + \mathbf{d}) \quad (21)$$

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:

$$\min_u \max_d J(\mathbf{x}_{a_i}, d_i, u_i) = \min_d \max_u \int_0^{t_f} \left(\mathbf{x}_{a_i}^T \mathbf{Q}_i \mathbf{x}_{a_i} + u_i^T R u_i - d_i^T R_d d_i \right) dt \quad (22)$$

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 [13]:

$$u_i = -\mathbf{K}_i \mathbf{x}_{a_i} \quad (23)$$

Moreover, the worst disturbance is obtained as

$$d_i = \mathbf{K}_{d_i} \mathbf{x}_{a_i} \quad (24)$$

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

$$\mathbf{K}_{d_i} = R_d^{-1} \mathbf{B}_{a_{d_i}}^T \mathbf{P}_{a_{d_i}} \quad (25)$$

$$\mathbf{K}_i = R^{-1} \mathbf{B}_{a_i}^T \mathbf{P}_{a_i} \quad (26)$$

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

$$-\mathbf{A}_a^T \mathbf{P}_{a_i} - \mathbf{Q}_i - \mathbf{P}_{a_i} \mathbf{A}_a + \mathbf{P}_{a_i} \mathbf{S}_{a_i} \mathbf{P}_{a_i} + \mathbf{P}_{a_i} \mathbf{S}_{a_{d_i}} \mathbf{P}_{d_{a_i}} = \mathbf{0} \quad (27)$$

$$-\mathbf{A}_a^T \mathbf{P}_{a_{d_i}} - \mathbf{Q}_i - \mathbf{P}_{a_{d_i}} \mathbf{A}_a + \mathbf{P}_{a_{d_i}} \mathbf{S}_{a_i} \mathbf{P}_{a_i} + \mathbf{P}_{a_{d_i}} \mathbf{S}_{a_{d_i}} \mathbf{P}_{a_{d_i}} = \mathbf{0} \quad (28)$$

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) [22] 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. Table 6 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}

Table 2: Quadrotor parameters

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

Table 3: Quadrotor parameters

Parameter	Unit	Value	Description
m_{total}	kg	1.074	Total Mass
d	N.m.sec ² /rad ²	3.2×10^{-6}	Drag Factor
b	N.sec ² /rad ²	3.13×10^{-5}	Thrust Factor
d_{cg}	m	0.2	CG Distance
Ω_{mean}	rpm	2000	Mean Rotor Speed
I_{xx}	kg.m ²	0.02839	Inertia about X-axis
I_{yy}	kg.m ²	0.03066	Inertia about Y-axis
I_{zz}	kg.m ²	0.0439	Inertia about Z-axis
I_{rotor}	kg.m ²	4.4398×10^{-5}	Rotor Inertia
I_{xz}	kg.m ²	6.87×10^{-7}	Inertia about XZ-axis

Table 4: Quadrotor parameters

Shape	Parameter	Unit	Value	Description
Mass	m_{total}	kg	1.074	Total mass of the quadrotor
Inertia	d	N.m.sec ² /rad ²	3.2×10^{-6}	Moment of inertia about xx-axis
Inertia	I_{yy}	kg.m ²	0.03066	Moment of inertia about yy-axis
Drag	b	N.sec ² /rad ²	3.13×10^{-5}	Drag coefficient
CG Distance	d_{cg}	m	0.2	Center of gravity distance
Mean Rotor Speed	Ω_{mean}	rpm	2000	Mean rotor speed

Table 5: Quadrotor parameters

Parameter	Unit	Value	Description	Parameter	Unit	Value	Description
m_{total}	kg	1.074	Mass	I_{xx}	kg.m ²	0.02839	Inertia X-axis
d	N.m.sec ² /rad ²	3.2×10^{-6}	Drag Factor	I_{yy}	kg.m ²	0.03066	Inertia Y-axis
b	N.sec ² /rad ²	3.13×10^{-5}	Thrust Factor	I_{zz}	kg.m ²	0.0439	Inertia Z-axis
d_{cg}	m	0.2	CG Distance	I_{rotor}	kg.m ²	4.4398×10^{-5}	Rotor Inertia
Ω_{mean}	rpm	2000	Mean Rotor Speed	I_{xz}	kg.m ²	6.87×10^{-7}	Inertia about XZ-axis

Table 6: Quadrotor parameters

Parameter	Unit	Value	Description
m_{total}	kg	1.074	Mass
d	N.m.sec ² /rad ²	3.2×10^{-6}	Drag Factor
b	N.sec ² /rad ²	3.13×10^{-5}	Thrust Factor
d_{cg}	m	0.2	CG Distance
Ω_{mean}	rpm	2000	Mean Rotor Speed
I_{xx}	kg.m ²	0.02839	Inertia X-axis
I_{yy}	kg.m ²	0.03066	Inertia Y-axis
I_{zz}	kg.m ²	0.0439	Inertia Z-axis
I_{rotor}	kg.m ²	4.4398×10^{-5}	Rotor Inertia
I_{xz}	kg.m ²	6.87×10^{-7}	Inertia about XZ-axis

5.1. Challenges in Designing and Implementation

Developing and implementing advanced control strategies for unmanned aerial vehicles (UAVs) presents several challenges that need careful consideration to ensure successful deployment in real-world scenarios. In this subsection, we briefly highlight the key challenges encountered during the design and implementation of the proposed controller.

1. **Nonlinear Dynamics:** quadrotors exhibit highly nonlinear and coupled dynamics, which demand sophisticated control algorithms to achieve precise and stable control.

2. **Model Uncertainty:** Accurate modeling of quadrotor dynamics is challenging due to uncertainties in aerodynamics, payload, and external disturbances. Robust control techniques are required to address model uncertainties effectively.
3. **Sensor Noise and Calibration:** Real-world sensors are prone to noise and calibration errors, affecting state estimation accuracy. Sensor fusion and calibration techniques are essential for reliable control.
4. **Real-time Computation:** quadrotor control systems must execute in real-time, necessitating computationally efficient algorithms to ensure low-latency response.
5. **Robustness to External Disturbances:** quadrotors operate in dynamic and unpredictable environments, making them susceptible to wind gusts and other disturbances. Robust control strategies are essential to maintain stability.
6. **Safety and Collision Avoidance:** Ensuring safety during quadrotor operation is critical, requiring collision avoidance algorithms and fail-safe mechanisms.
7. **Experimental Validation:** Real-world testing and validation of the controller pose logistical and safety challenges, requiring careful experimental setups.

Addressing these challenges is crucial to ensure the successful deployment and practicality of the proposed controller in real-world applications.

5.2. Tuning of LQIR-DG Weighting Matrices

The parameters of the LQIR-DG controller approach, including weight coefficients (\mathbf{Q}_i for $i = \text{roll}$, pitch , yaw , R_d , and R) are tuned using a heuristic optimization algorithm based on the Tabu Continuous Ant Colony System (TCACS) [17] approach. In this method, ants utilize the concepts of promising list and tabu balls to move toward the goal ant gradually. The pseudo-code of TCACS is shown in Figure 5. For this purpose, ants find the promising areas to contain the global minimum and perform searching within tabu balls of the bad regions. TCACS parameters are shown in Table 7. Here, it is assumed that the value of R is identical for all attitude channels and considered with the value of 1. Moreover, the initial values of the ants position (\mathbf{Q}_{roll} , $\mathbf{Q}_{\text{pitch}}$, \mathbf{Q}_{yaw} and R_d) for $l = 1, \dots, N$ is selected using a random distribution. The cost is denoted in iteration i using the quality of the tracking error between the set-point, \mathbf{x}_{sp} , and the quadrotor states, $\mathbf{x} = [\mathbf{x}_{\text{roll}} \quad \mathbf{x}_{\text{pitch}} \quad \mathbf{x}_{\text{yaw}}]^T$ as

$$\text{ITSE} = \int_0^{t_f} t (\mathbf{x}_{sp} - \mathbf{x})^T (\mathbf{x}_{sp} - \mathbf{x}) dt \quad (29)$$

where t and t_f are the response time of the system and the final time, respectively. Finally, when the stopping condition of the TCACS algorithm is reached, the best values of the LQIR-DG parameters are computed, shown in Table 8

Algorithm 1 Tabu Continuous Ant Colony System (TCACS)

```

1: procedure TCACS
2:   Initialize parameters, lists, and values
3:   while not terminated do
4:     if first iteration then
5:       Sample initial ant positions
6:     else
7:       Move ants
8:     end if
9:     Update structures and distributions
10:   end while
11: end procedure

```

Figure 5: Pseudo-code of the TCACS optimization algorithm [17].

Table 7: Parameters of the TCACS optimization algorithm.

Parameter	Value	Description
N	15	Number of Ants
I_{\max}	10000	Maximum of Iteration
Tolerance	10^{-4}	Maximum accepted error

Table 8: 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_d	1.2764

5.3. Identification of the 3-Dof quadrotor platform model

As described in section 3.3, the parameters of the quadrotor platform, denoted by $\Gamma_i (i = 1, \dots, 11)$, are identified using the NLS-TRR algorithm. To increase the accuracy of parameter identification, three scenarios are considered according to Table 9. In the first scenario, depicted in Figure 6, the quadrotor rotates about only one axis (roll, pitch, or yaw axes) to identify the parameters Γ_3 , Γ_5 , Γ_8 , Γ_9 , and Γ_{11} . In the second scenario, according to Figure 7, the parameters Γ_1 and Γ_7 are estimated by rotating the experimental platform around its roll and pitch axes simultaneously. Finally, Figure 8 displays the results of the third scenario including the estimation of the parameters Γ_2 , Γ_4 , Γ_6 , and Γ_{10} 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 are computed and denoted in Table 11. These results illustrate that the outputs of the simulation results for the quadrotor model are consistent with reality.

Table 9: 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 10: True values of the quadrotor parameters.

Parameter	Value	Parameter	Value
Γ_1	4.9895×10^{-6}	Γ_6	2.5294
Γ_2	0.0029	Γ_7	0.0002
Γ_3	42.1805	Γ_8	18.46
Γ_4	0.0002	Γ_9	0.0022
Γ_5	-0.0023	Γ_{10}	-1.4456×10^{-5}
Γ_{11}	24.4570		

Table 11: True values of the quadrotor parameters.

Parameter 1	Value 1	Parameter 2	Value 2	Parameter 3	Value 3
Γ_1	4.9895×10^{-6}	Γ_2	0.0029	Γ_3	42.1805
Γ_4	0.0002	Γ_5	-0.0023	Γ_6	2.5294
Γ_7	0.0002	Γ_8	18.46	Γ_9	0.0022
Γ_{10}	-1.4456×10^{-5}	Γ_{11}	24.4570		

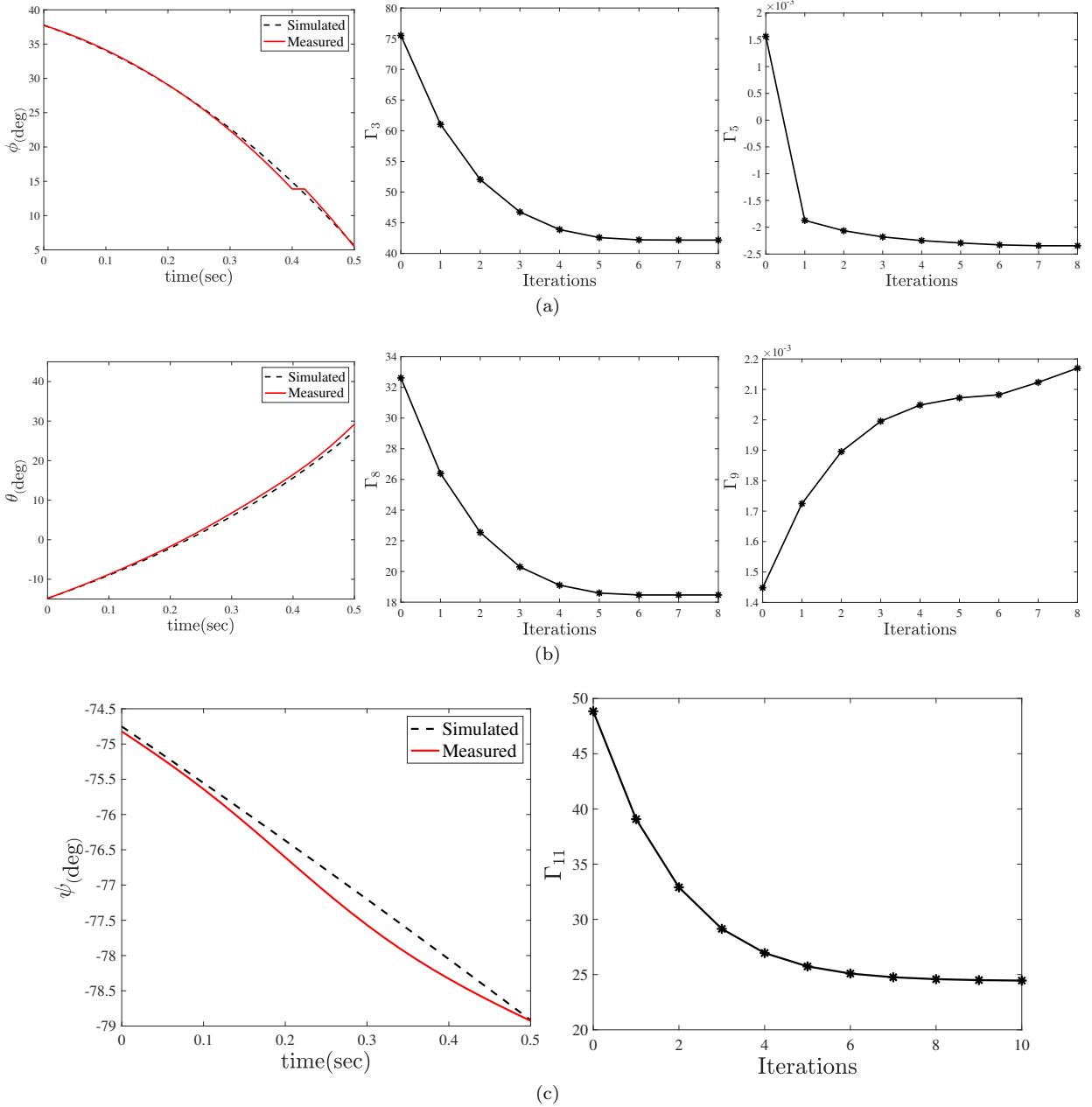
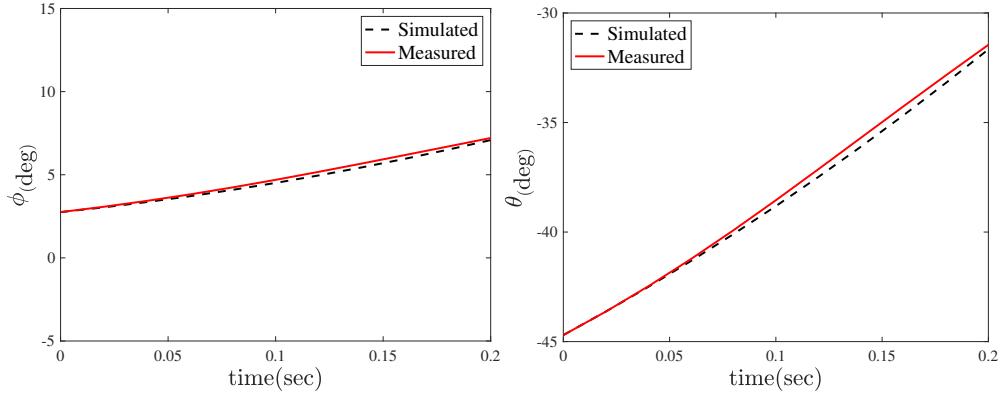
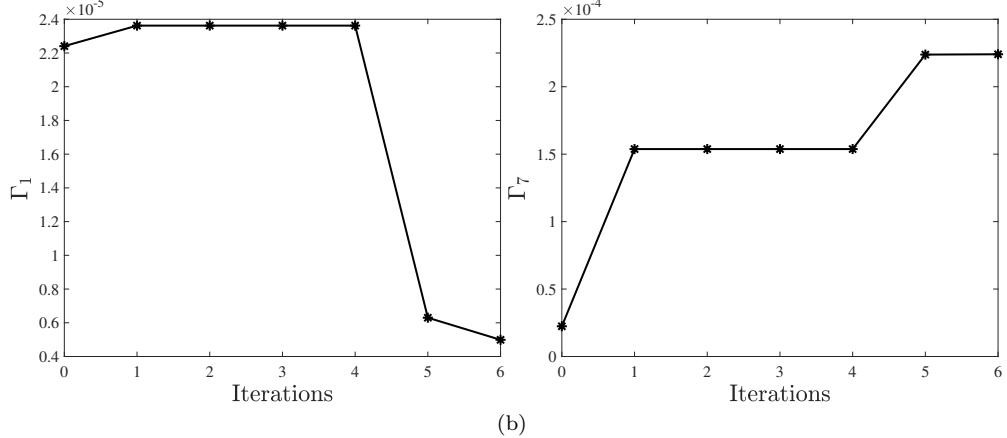


Figure 6: Identification process results when the quadrotor rotates about only one axis: (a) identification of Γ_3 and Γ_5 in free roll motion. (b) identification of Γ_8 and Γ_9 in free pitch motion. (c) identification of Γ_{11} in free yaw motion.

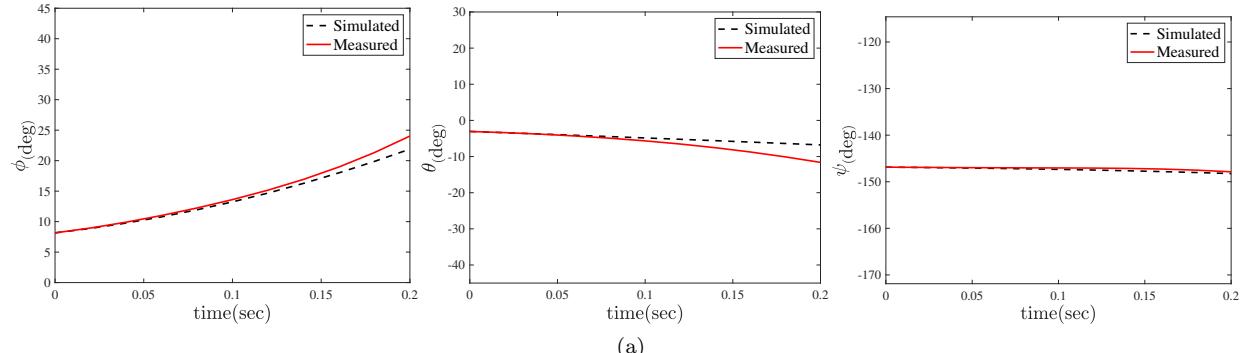


(a)

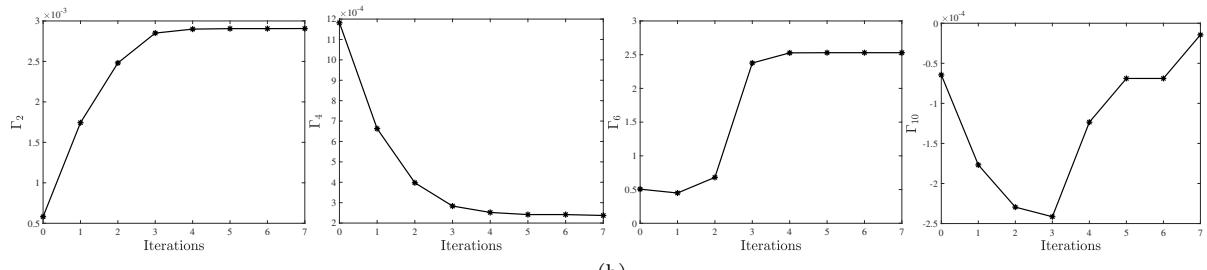


(b)

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



(a)



(b)

Figure 8: 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_2 , Γ_4 , Γ_6 , and Γ_{10} parameters.

5.4. 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 12.

Table 12: PID controller parameters

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

5.4.1. 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 9 and 10. Figure 9 (a) compares the desired and output signals, i.e., the Euler angles during the regulation problem. Moreover, Figure 9 (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 10 (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.

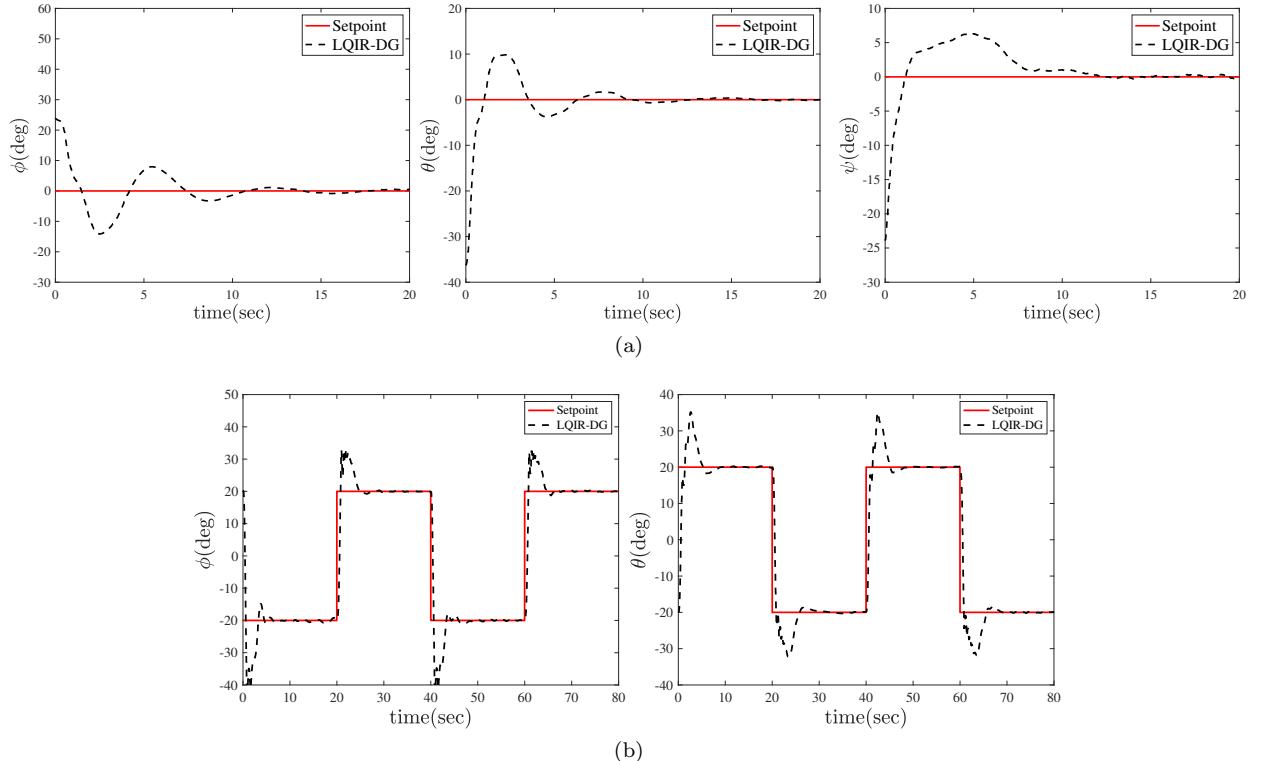


Figure 9: Comparison of actual roll and pitch angles with the desired values in (a) regulation and (b) tracking problems.

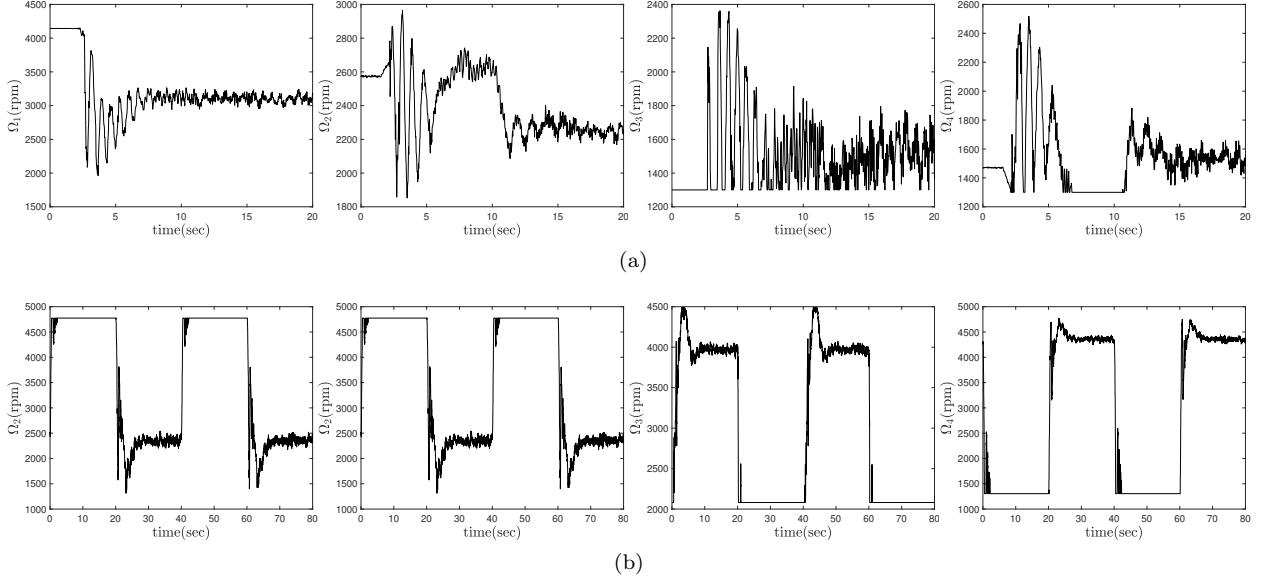


Figure 10: Rotational velocity commands in (a) regulation and (b) tracking problems.

5.4.2. Investigating the Disturbance Rejection

Here, the effect of the input disturbance is investigated on the performance of the proposed controller. The input disturbance, d_{Ω_i} , is considered as a change in command of the rotational velocity, modeled as

$$d_{\Omega_1} = d_{\Omega_2} = -d_{\Omega_3} = -d_{\Omega_4} = \begin{cases} 500 \text{ rpm} & 20 < t < 60 \\ 0 & \text{otherwise} \end{cases} \quad (30)$$

Figure 11 illustrates the roll and pitch angles in the regulation problem when the input disturbance occurs. These results indicate that the proposed controller can stabilize the quadrotor platform in the presence of input disturbance.

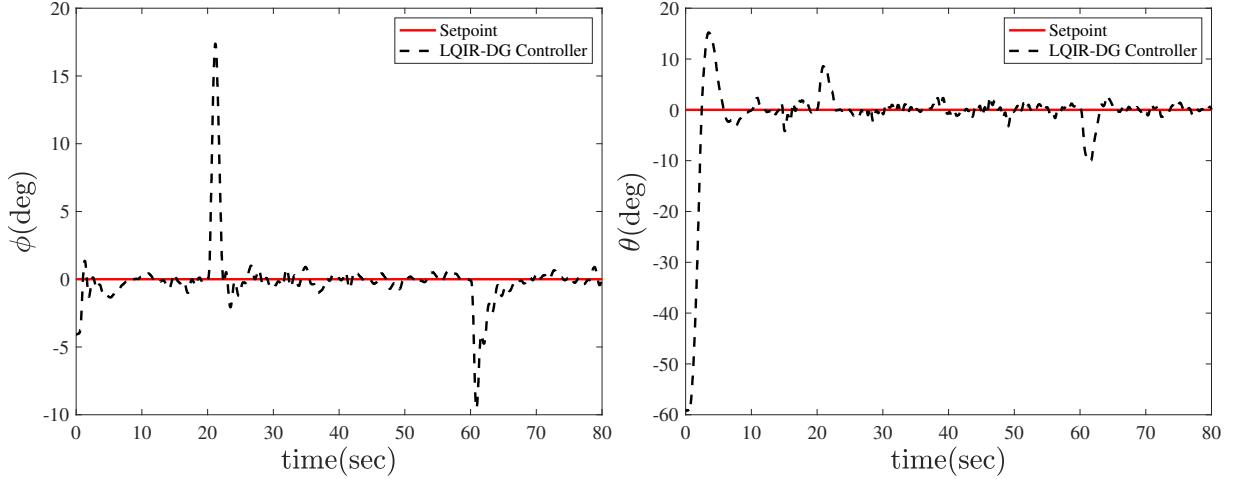


Figure 11: Comparison of actual roll and pitch angles with the desired, when the input disturbance occurs.

5.4.3. 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 12. Figure 13 (a) compares the desired and the actual roll angle and Figure 13 (b) shows the desired and the actual pitch angle, when the uncertainty of moments of inertia is present. Moreover, Figure 13 (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.

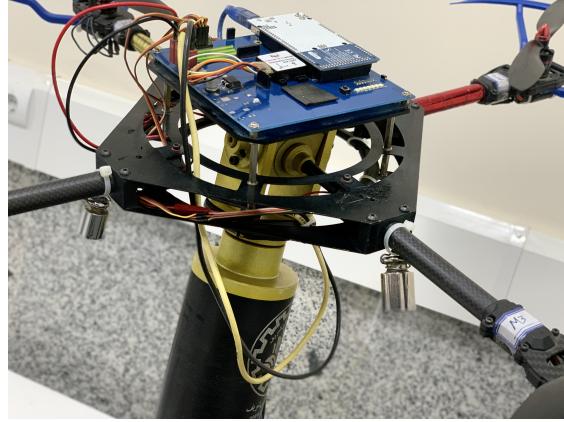


Figure 12: Quadrotor 3-DoF platform with added weights.

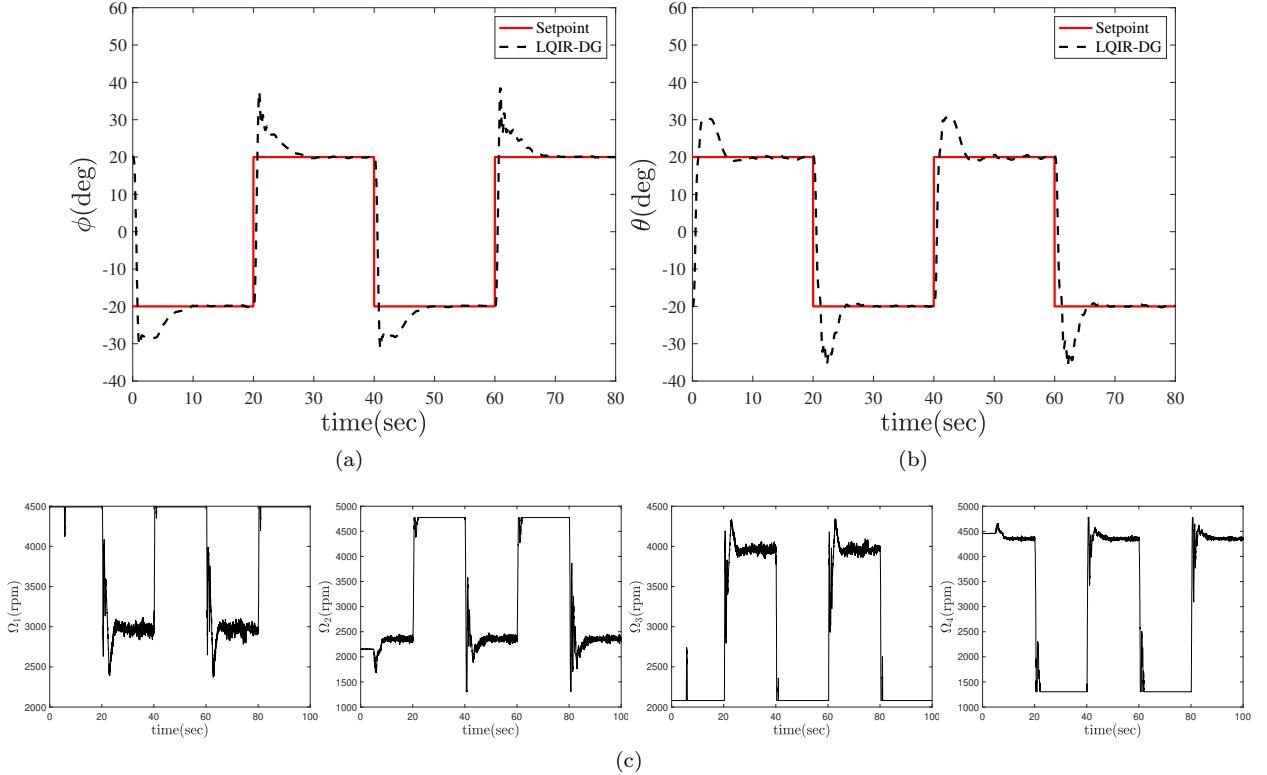


Figure 13: Comparison of actual roll and pitch angles with desired values, when the modeling uncertainty is present.

5.4.4. Comparison with the Control Strategies

Figure 16 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 15 for the cost function, introduced in equation (22). The median of Root Mean Square Error (RMSE) is shown in the crossline in the box plot. Moreover, the LQIR-DG controller performance with famous disturbance rejection methods, such as Active Disturbance Rejection Control (ADRC) [12] and Disturbance Observer-Based Control (DOBC) [3] are compared in Figure 17, when the input disturbances occur according to equation (30). For the cost function, denoted in equation (22), the box plot of these robust controllers is illustrated in Figure ?. 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.

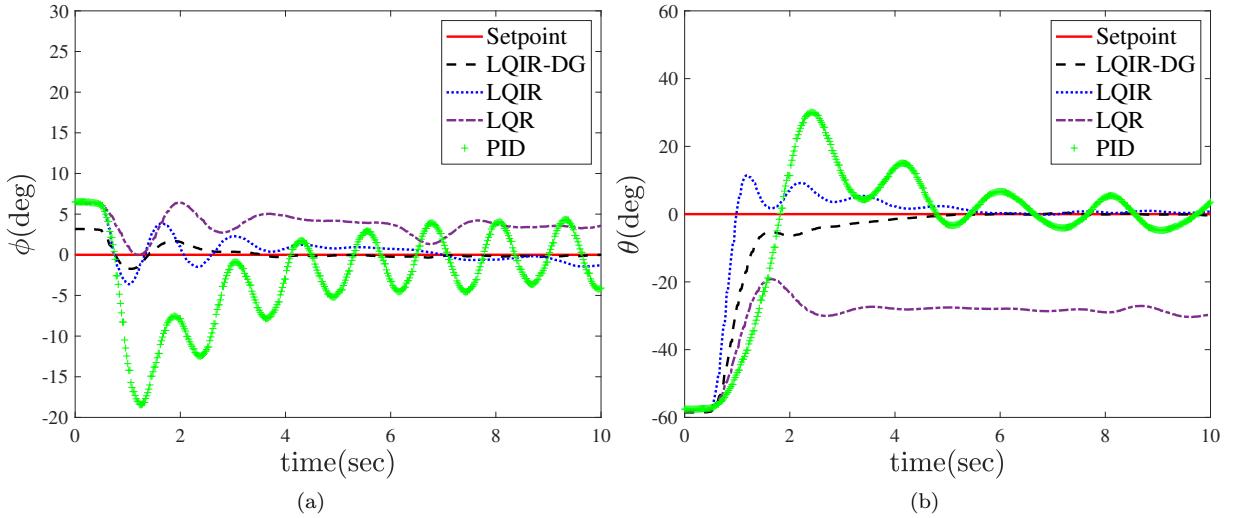


Figure 14: Comparison of LQIR-DG structure with the variant of LQR and PID in regulation problem: (a) roll angle (b) pitch angle.

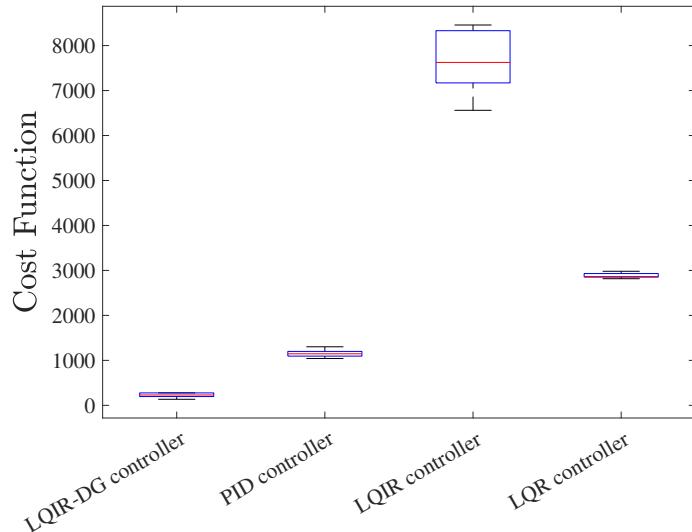
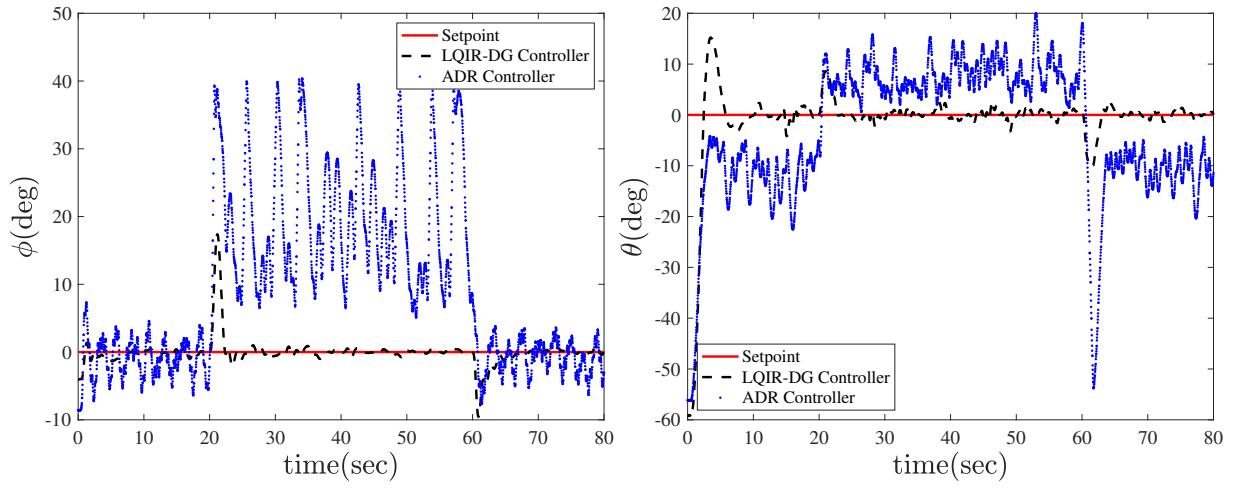
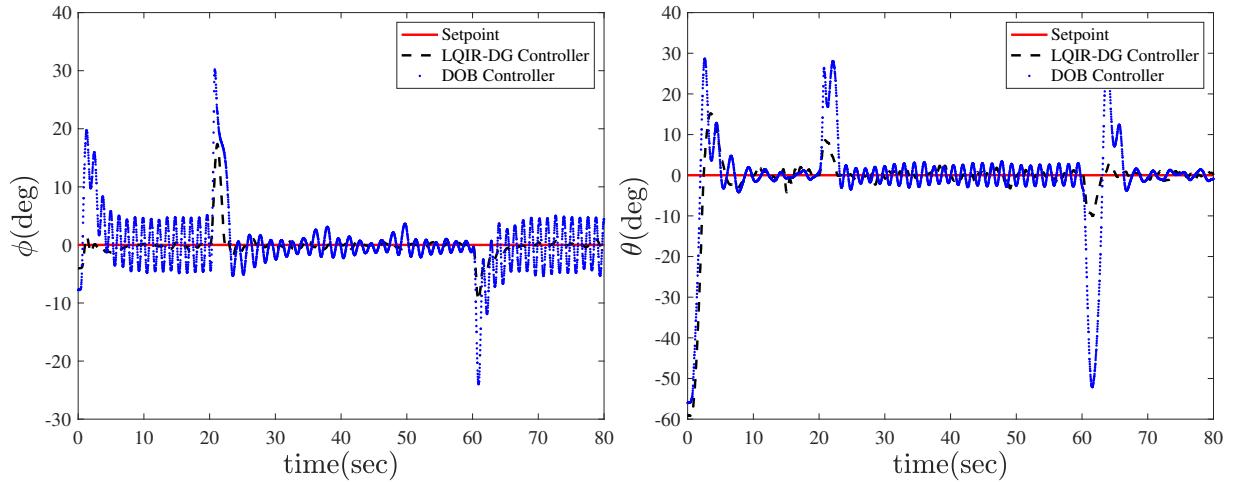


Figure 15: Box plot of LQIR-DG, LQR, LQIR, and PID controllers.



(a)



(b)

Figure 16: Comparison of LQIR-DG structure with the famous disturbance rejection methods: (a) Active Disturbance Rejection Control (ADRC) (b) Disturbance Observer-Based Control (DOBC).

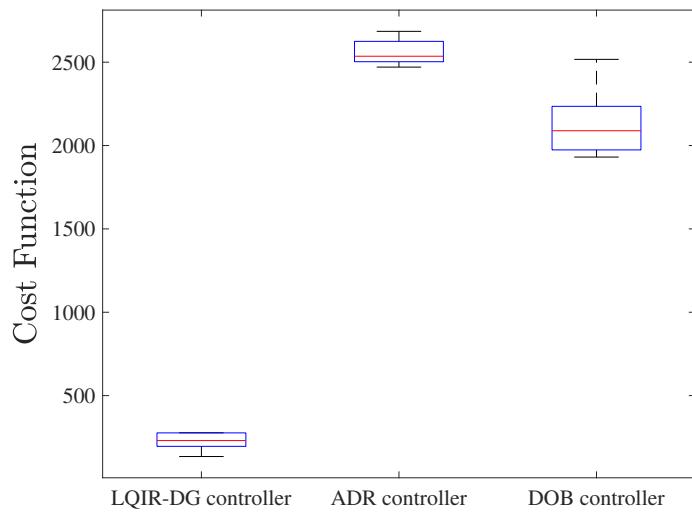


Figure 17: Box plot of LQIR-DG, ADRC, and DOBC methods.

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.

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] Aghayan, Z.S., Alfi, A., Lopes, A.M., 2023. Disturbance observer-based delayed robust feedback control design for a class of uncertain variable fractional-order systems: Order-dependent and delay-dependent stability. *ISA Transactions* 138, 20–36. URL: <https://www.sciencedirect.com/science/article/pii/S0019057823001052>, doi:<https://doi.org/10.1016/j.isatra.2023.03.008>.
- [4] 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.
- [5] 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.
- [6] 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.
- [7] Bouabdallah, S., 2007. Design and control of quadrotors with application to autonomous flying. URL: <https://api.semanticscholar.org/CorpusID:108233951>.
- [8] 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.
- [9] Chara, K., Yassine, A., Srairi, F., Mokhtari, K., 2022. A robust synergetic controller for quadrotor obstacle avoidance using bézier curve versus b-spline trajectory generation. *Intelligent Service Robotics* 15. doi:10.1007/s11370-021-00408-0.
- [10] 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>.
- [11] 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.
- [12] Cheng, Y., Fan, Y., Zhang, P., Yuan, Y., Li, J., 2023. Design and parameter tuning of active disturbance rejection control for uncertain multivariable systems via quantitative feedback theory. *ISA Transactions* URL: <https://www.sciencedirect.com/science/article/pii/S0019057823002926>, doi:<https://doi.org/10.1016/j.isatra.2023.06.025>.
- [13] Engwerda, J., 2006. Linear Quadratic Games: An Overview. WorkingPaper. Macroeconomics. Subsequently published in *Advances in Dynamic Games and their Applications* (book), 2009 Pagination: 32.
- [14] 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.
- [15] 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>.
- [16] 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.
- [17] Karimi, A., Nobahari, H., Siarry, P., 2010. Continuous ant colony system and tabu search algorithms hybridized for global minimization of continuous multi-minima functions. *Computational Optimization and Applications* 45, 639–661. doi:10.1007/s10589-008-9176-7.
- [18] 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>.

- [19] 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>.
- [20] 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>.
- [21] 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:<https://doi.org/10.1109/ICRoM57054.2022.10025263>.
- [22] 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.
- [23] 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>.
- [24] 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>.
- [25] Simon, D., 2006. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley-Interscience, 605 Third Avenue, New York, NY, United States.
- [26] 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>.
- [27] 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>.
- [28] 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>.
- [29] 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>.
- [30] 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>.
- [31] 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>.
- [32] 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>.
- [33] Zulu, A., John, S., 2014. A review of control algorithms for autonomous quadrotors. *Open Journal of Applied Sciences* 04, 547–556. doi:<https://doi.org/10.4236/ojapps.2014.414053>.

Appendix A. Linearization Proof of the Quadrotor Nonlinear Model

Here, the nonlinear model of the quadrotor, described in equations (10)-(15), are linearized using first-order Taylor series expansion about the equilibrium points (\mathbf{x}^* and \mathbf{u}^*). For this purpose, the linear form of the nonlinear system, denoted as $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, is computed as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (\text{A.1})$$

where \mathbf{A} and \mathbf{B} are, respectively, the states and input matrices, computed as [25]

$$\mathbf{A} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\mathbf{x}^*, \mathbf{u}^*} \quad (\text{A.2})$$

$$\mathbf{B} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{\mathbf{x}^*, \mathbf{u}^*} \quad (\text{A.3})$$

To linearize the nonlinear model of the quadrotor around the equilibrium points ($\mathbf{x}^* = 0$ and $\mathbf{u}^* = 0$), the Jacobian matrix of nonlinear model, denoted in equations (10)-(15), is expressed as:

$$f_1 = \dot{x}_1 = \Gamma_1 x_1 x_3 - \Gamma_2 x_3 x_5 + \Gamma_3 (u_{\text{roll}} + d_{\text{roll}}) + \Gamma_4 d (u_{\text{yaw}} + u_{\text{yaw}}) + \Gamma_5 \Omega_{c,r} \quad (\text{A.4})$$

$$f_2 = \dot{x}_2 = x_1 + (x_3 \sin(x_2) + x_3 \cos(x_2)) \tan(x_4) \quad (\text{A.5})$$

$$f_3 = \dot{x}_3 = \Gamma_6 x_1 x_5 - \Gamma_7 (x_1^2 - x_5^2) + \Gamma_8 (u_{\text{pitch}} + d_{\text{pitch}}) - \Gamma_9 \Omega_{c,r} \quad (\text{A.6})$$

$$f_4 = \dot{x}_4 = x_3 \cos(x_4) - x_5 \sin(x_2) \quad (\text{A.7})$$

$$f_5 = \dot{x}_5 = \Gamma_{10} x_1 x_3 - \Gamma_1 x_3 x_5 + \Gamma_{11} (u_{\text{yaw}} + d_{\text{yaw}}) + \Gamma_4 (u_{\text{roll}} + d_{\text{roll}}) \quad (\text{A.8})$$

$$f_6 = \dot{x}_6 = \frac{x_3 \sin(x_4) + x_5 \cos(x_2)}{\cos(x_4)} \quad (\text{A.9})$$

$$\mathbf{A} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\mathbf{x}^*=\mathbf{u}^*=0} = \begin{bmatrix} f_1 & \dots & f_1 \\ \frac{x_1}{x_1} & \dots & \frac{x_6}{x_6} \\ \vdots & \ddots & \vdots \\ f_6 & \dots & f_6 \\ \frac{x_1}{x_1} & \dots & \frac{x_6}{x_6} \end{bmatrix} \quad (\text{A.10})$$

$$= \begin{bmatrix} \Gamma_1 x_3 & -\Gamma_2 x_5 & \Gamma_1 x_1 - \Gamma_2 x_3 & 0 & -\Gamma_2 x_3 & 0 \\ 1 & x_3 \cos(x_2) \tan(x_4) & a_{23} & a_{24} & 0 & 0 \\ \Gamma_6 x_5 - 2\Gamma_7 x_1 & 0 & 0 & 0 & \Gamma_6 x_1 - 2\Gamma_7 x_5 & 0 \\ 0 & -x_5 \cos(x_4) & \cos(x_4) & -x_3 \sin(x_4) & -\sin(x_4) & 0 \\ \Gamma_{10} x_3 & 0 & \Gamma_{10} x_1 - \Gamma_1 x_5 & 0 & -\Gamma_1 x_3 & 0 \\ 0 & -x_5 \cos(x_2) & \frac{\sin(x_2)}{\cos(x_4)} & a_{64} & \frac{\cos(x_2)}{\cos(x_4)} & 0 \end{bmatrix} \quad (\text{A.11})$$

$$a_{24} = (x_3 \sin(x_2) + x_3 \cos(x_2)) \sec^2(x_4)$$

$$a_{64} = x_3 + \frac{\sin(x_4)(x_5 \cos(x_2) + x_3 \sin(x_4))}{\cos(x_4)^2}$$

$$a_{23} = (\sin(x_2) + \cos(x_2)) \tan(x_4)$$

$$\mathbf{B} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{\mathbf{x}^*=\mathbf{u}^*=0} = \begin{bmatrix} \frac{\partial f_1}{\partial u_{\text{roll}}} & \frac{\partial f_1}{\partial u_{\text{pitch}}} & \frac{\partial f_1}{\partial u_{\text{yaw}}} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_6}{\partial u_{\text{roll}}} & \frac{\partial f_6}{\partial u_{\text{pitch}}} & \frac{\partial f_6}{\partial u_{\text{yaw}}} \end{bmatrix} = \begin{bmatrix} \Gamma_3 & 0 & \Gamma_4 \\ 0 & 0 & 0 \\ 0 & \Gamma_8 & 0 \\ 0 & 0 & 0 \\ \Gamma_4 & 0 & \Gamma_{11} \\ 0 & 0 & 0 \end{bmatrix} \quad (\text{A.12})$$

Finally, the linearized matrices defined at the equilibrium points are given as:

$$\mathbf{A} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\mathbf{x}_e^*, \mathbf{u}_e^*} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{B} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{\mathbf{x}_e^*, \mathbf{u}_e^*} = \begin{bmatrix} \Gamma_3 & 0 & \Gamma_4 \\ 0 & 0 & 0 \\ 0 & \Gamma_8 & 0 \\ 0 & 0 & 0 \\ \Gamma_4 & 0 & \Gamma_{11} \\ 0 & 0 & 0 \end{bmatrix} \quad (\text{A.13})$$

Appendix B. Linear Quadratic Differential Game Proof

Consider the differential game described by

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} + \mathbf{B}_d \mathbf{d} \quad (\text{B.1})$$

with for player one, the quadratic cost function:

$$J_1(u, d) = \frac{1}{2} \int (\mathbf{x}^T \mathbf{Q}_1 \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u} - \mathbf{d}^T \mathbf{R}_d \mathbf{d}) dt \quad (\text{B.2})$$

and for player two, the opposite objective function:

$$J_2(u, d) = -J_1(u, d) \quad (\text{B.3})$$

Hamiltonian for player one:

$$\mathcal{H} = g + \lambda_1^T a = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} + \frac{1}{2} \mathbf{u}^T \mathbf{R} \mathbf{u} - \frac{1}{2} \mathbf{d}^T \mathbf{R}_d \mathbf{d} + \lambda_1^T (\mathbf{Ax} + \mathbf{Bu} + \mathbf{B}_d \mathbf{d}) \quad (\text{B.4})$$

The solution for this optimization problem is unique and is given by

$$\mathbf{u}^* = \mathbf{F}_1 \mathbf{x} \quad \text{and} \quad \mathbf{d}^* = \mathbf{F}_2 \mathbf{x} \quad (\text{B.5})$$

$$\mathcal{H} = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} + \frac{1}{2} \mathbf{u}^T \mathbf{R} \mathbf{u} - \frac{1}{2} \mathbf{x}^T \mathbf{F}_2^T \mathbf{R}_d \mathbf{F}_2 \mathbf{x} + \lambda_1^T (\mathbf{Ax} + \mathbf{Bu} + \mathbf{B}_d \mathbf{F}_2 \mathbf{x}) \quad (\text{B.6})$$

$$\dot{\mathbf{x}} = \frac{\partial \mathcal{H}}{\partial \lambda_1} = \mathbf{Ax} + \mathbf{Bu} + \mathbf{B}_d \mathbf{F}_2 \mathbf{x} \quad (\text{B.7})$$

$$\dot{\lambda}_1 = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}} = -(\mathbf{Qx} - \mathbf{F}_2^T \mathbf{R}_d \mathbf{F}_2 \mathbf{x}) - \lambda_1^T (\mathbf{A} + \mathbf{B}_d \mathbf{F}_2) \quad (\text{B.8})$$

$$\frac{\partial \mathcal{H}}{\partial \mathbf{u}} = 0 \Rightarrow \mathbf{Ru} + \mathbf{B}^T \lambda_1 = 0 \Rightarrow \mathbf{u} = -\mathbf{R}^{-1} \mathbf{B}^T \lambda_1 \quad (\text{B.9})$$

Assume that $\lambda_1 = \mathbf{K}_1 \mathbf{x}$, then

$$\dot{\lambda} = \dot{\mathbf{K}}_1 \mathbf{x} + \mathbf{K}_1 \dot{\mathbf{x}} \quad (\text{B.10})$$

in infinite horizon assumed $\dot{\lambda} = 0$, so

$$\begin{aligned} \dot{\lambda} = \mathbf{K}_1 (\mathbf{Ax} + \mathbf{Bu} + \mathbf{B}_d \mathbf{F}_2 \mathbf{x}) &\xrightarrow{\mathbf{u} = -\mathbf{R}^{-1} \mathbf{B}^T \lambda_1, \mathbf{d} = \mathbf{F}_2 \mathbf{x}, \lambda = \mathbf{K}_1 \mathbf{x}} \\ &\xrightarrow{\dot{\lambda}_1 = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}}} = -(\mathbf{Qx} - \mathbf{F}_2^T \mathbf{R}_d \mathbf{F}_2 \mathbf{x}) - \lambda_1^T (\mathbf{A} + \mathbf{B}_d \mathbf{F}_2) \end{aligned} \quad (\text{B.11})$$

$$\mathbf{K}_1 (\mathbf{Ax} - \mathbf{BR}^{-1} \mathbf{B}^T \mathbf{K}_1 \mathbf{x} + \mathbf{B}_d \mathbf{F}_2 \mathbf{x}) = \mathbf{Qx} - \mathbf{F}_2^T \mathbf{R}_d \mathbf{F}_2 \mathbf{x} - (\mathbf{A} + \mathbf{B}_d \mathbf{F}_2)^T \mathbf{K}_1 \mathbf{x} \quad (\text{B.12})$$

equation above is simplified to

$$-(\mathbf{A} + \mathbf{B}_d \mathbf{F}_2)^T \mathbf{K}_1 - \mathbf{K}_1 (\mathbf{A} + \mathbf{B}_d \mathbf{F}_2) + \mathbf{K}_1 \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{K}_1 - \mathbf{Q} + \mathbf{F}_2^T \mathbf{R}_d \mathbf{F}_2 = 0 \quad (\text{B.13})$$

For the second player also use above method.

$$\mathcal{H} = g + \lambda_2^T a = -\frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} - \frac{1}{2} \mathbf{x}^T \mathbf{F}_1^T \mathbf{R} \mathbf{F}_1 \mathbf{x} + \frac{1}{2} \mathbf{d}^T \mathbf{R}_d \mathbf{d} + \lambda_2^T (\mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{F}_1 \mathbf{x} + \mathbf{B}_d \mathbf{d}) \quad (\text{B.14})$$

$$\dot{\mathbf{x}} = \frac{\partial \mathcal{H}}{\partial \lambda_2} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{F}_1 \mathbf{x} + \mathbf{B}_d \mathbf{d} \quad (\text{B.15})$$

$$\dot{\lambda}_2 = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}} = (\mathbf{Q} \mathbf{x} + \mathbf{F}_1^T \mathbf{R} \mathbf{F}_1 \mathbf{x}) - \lambda_2^T (\mathbf{A} + \mathbf{B} \mathbf{F}_1) \quad (\text{B.16})$$

$$\frac{\partial \mathcal{H}}{\partial \mathbf{d}} = 0 \Rightarrow \mathbf{R}_d \mathbf{d} + \mathbf{B}_d^T \lambda_2 = 0 \Rightarrow \mathbf{d} = -\mathbf{R}_d^{-1} \mathbf{B}_d^T \lambda_2 \quad (\text{B.17})$$

Assume that $\lambda_2 = \mathbf{K}_2 \mathbf{x}$, then

$$\dot{\lambda}_2 = \dot{\mathbf{K}}_2 \mathbf{x} + \mathbf{K}_2 \dot{\mathbf{x}} \quad (\text{B.18})$$

in infinite horizon assumed $\dot{\lambda} = 0$, so

$$\dot{\lambda}_2 = \mathbf{K}_2 (\mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{F}_1 \mathbf{x} + \mathbf{B}_d \mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2 \mathbf{x}) \quad (\text{B.19})$$

also:

$$\dot{\lambda}_2 = (\mathbf{Q} \mathbf{x} + \mathbf{F}_1^T \mathbf{R} \mathbf{F}_1 \mathbf{x}) - \lambda_2^T (\mathbf{A} + \mathbf{B} \mathbf{F}_1) \quad (\text{B.20})$$

then:

$$\mathbf{K}_2 (\mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{F}_1 \mathbf{x} + \mathbf{B}_d \mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2 \mathbf{x}) = (\mathbf{Q} \mathbf{x} + \mathbf{F}_1^T \mathbf{R} \mathbf{F}_1 \mathbf{x}) - \mathbf{K}_2 (\mathbf{A} + \mathbf{B} \mathbf{F}_1) \quad (\text{B.21})$$

$$-(\mathbf{A} + \mathbf{B} \mathbf{F}_1)^T \mathbf{K}_2 - \mathbf{K}_2 (\mathbf{A} + \mathbf{B} \mathbf{F}_1) + \mathbf{K}_2 \mathbf{B}_d \mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2 + \mathbf{Q} + \mathbf{F}_1^T \mathbf{R} \mathbf{F}_1 = 0 \quad (\text{B.22})$$

In above calculated \mathbf{u} , \mathbf{d} :

$$\mathbf{u} = -\mathbf{R}^{-1} \mathbf{B}^T \mathbf{K}_1 \mathbf{x} \text{ and } \mathbf{d} = -\mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2 \mathbf{x} \quad (\text{B.23})$$

then $\mathbf{u} = \mathbf{F}_1 \mathbf{x}$ and $\mathbf{d} = \mathbf{F}_2 \mathbf{x}$, so

$$\mathbf{F}_1 = -\mathbf{R}^{-1} \mathbf{B}^T \mathbf{K}_1 \text{ and } \mathbf{F}_2 = -\mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2 \quad (\text{B.24})$$

with implementation to above equation:

$$-(\mathbf{A} - \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{K}_1)^T \mathbf{K}_2 - \mathbf{K}_2 (\mathbf{A} - \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{K}_1) + \mathbf{K}_2 \mathbf{B}_d \mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2 + \mathbf{Q} + \mathbf{K}_1^T \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{K}_1 = 0 \quad (\text{B.25})$$

$$-(\mathbf{A} - \mathbf{B}_d \mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2)^T \mathbf{K}_1 - \mathbf{K}_1 (\mathbf{A} - \mathbf{B}_d \mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2) + \mathbf{K}_1 \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{K}_1 - \mathbf{Q} + \mathbf{K}_2^T \mathbf{B}_d \mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K}_2 = 0 \quad (\text{B.26})$$

by adding two equation there is:

$$-(\mathbf{K}_1 + \mathbf{K}_2) [\mathbf{A} - \mathbf{S} \mathbf{K}_1 - \mathbf{S}_d \mathbf{K}_2] - [\mathbf{A} - \mathbf{S} \mathbf{K}_1 - \mathbf{S}_d \mathbf{K}_2]^T (\mathbf{K}_1 + \mathbf{K}_2) = 0 \rightarrow \mathbf{K}_1 = -\mathbf{K}_2 \quad (\text{B.27})$$

The two riccati differential equation have solution:

$$-\mathbf{A}^T \mathbf{K} - \mathbf{K} \mathbf{A} + \mathbf{K} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{K} - \mathbf{Q} - \mathbf{K}^T \mathbf{B}_d \mathbf{R}_d^{-1} \mathbf{B}_d^T \mathbf{K} = 0 \quad (\text{B.28})$$