Journal Européen des Systèmes Automatisés
Vol. 58, No. 11, November, 2025, pp. 2389-2404
Journal homepage: http://iieta.org/journals/jesa

Performance Comparison of PID, LQR, and H_∞ Controllers for Quadrotor Attitude and Altitude Control

Kared Saber[1*], Mechhoud Elarkam[2], Nafir Nourreddine[1], Ahmida Zahir[1]

[1] Research Electronic Laboratory of Skikda, Université20 Août 1955 Skikda, Skikda 21000, Algeria
[2] Automatic Laboratory of Skikda, Université20 Août 1955 Skikda, Skikda 21000, Algeria

Corresponding Author Email: s.kared@univ-skikda.dz

Copyright: ©2025 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/jesa.581116

Received: 20 October 2025
Revised: 23 November 2025
Accepted: 28 November 2025
Available online: 30 November 2025

Keywords:
quadrotor, nominal, PID, LQR, stability, mathematic, simulation, dynamic

ABSTRACT

The quadrotor is a Multi Inputs-Multi Outputs (MIMO) system characterized by four control inputs and six degrees of freedom (DOF), as well as its underactuated structure and inherent stability. To improve stabilization, and precise control of the quadrotor system, a comparative study is conducted between three control strategies: a conventional proportional integral derivative (PID) controller, a Linear Quadratic Regulator (LQR), and a H_∞ controller implemented on a quadrotor unmanned aerial vehicle (QUAV). Therefore, a comprehensive mathematical model of the quadrotor dynamics is formulated as a basis for designing and evaluating the various controllers. On this basis, the H_∞ control approach is adopted, as it is widely used for multivariable systems to ensure robust performance under varying operating conditions. Thus, the three control laws are comparatively assessed through simulation in terms of stability, trajectory tracking, disturbance rejection, and transient response. The results show that while the conventional PID and LQR controllers offer satisfactory performance under nominal conditions, the proposed H_∞ controller achieves superior robustness and faster dynamic response.

1. INTRODUCTION

In the last decades unmanned aerial vehicles (UAVs) especially quadrotors have emerged as a prominent research platform in control engineering due to their mechanical simplicity, high manoeuvrability, and broad range of applications. The using of quadrotors is increased in different fields such as surveillance, environmental monitoring, mapping, logistics, and scientific exploration. Consequently, their relatively simple mechanical structure, consisting of four rotors that generate both thrust and torque, the main challenge in quadrotor is the control problem. This complexity arises from its nonlinear dynamics, strong coupling between motion axes, underactuated configuration, and sensitivity to external disturbances such as wind gusts or payload variations [1-3].

In order to ensure stability, precision, and robustness of quadrotor flight, advanced and efficient control strategies are needed. From several approaches proposed in the literature, the Three of them have attracted special attention: the proportional integral derivative (PID) controller [4-6], the Linear Quadratic Regulator (LQR) [7, 8], and the H_∞ controller [9-12]. The PID controller, appreciated for its simplicity and ease of implementation, performs effectively in systems with linear or mildly nonlinear dynamics; however, often exhibits degraded performance with parameter variations or external disturbances. However, LQR controller-derived from optimal control theory achieves superior dynamic response and stability margins by minimizing a quadratic cost function that balances control effort against state deviations. Whereas H_∞ controller, supported by robust control theory, further extends these capabilities by explicitly accounting for model uncertainties and disturbances, providing ensured performance even under highly nonlinear or uncertain operating conditions.

This paper presents a comparative study of these three control strategies PID, LQR, and H_∞ applied to the attitude and position control of a quadrotor UAV. A detailed mathematical model of the system is developed, and the controllers are evaluated through simulation analysis. Their performances are assessed based on many criteria such as transient response, overshoot, steady-state accuracy, and disturbance rejection. The comparative results aim to highlight the strengths and limitations of each control strategy, offering valuable insights for selecting an appropriate control scheme for quadrotor applications requiring both precision and robustness.

2. MODELLING OF THE QUAV

The Quadrotor is a type of unmanned aerial vehicle (UAV) characterized by four rotors arranged symmetrically at the ends of a cross-shaped frame, with the control electronics and power system typically located at the center. This structure offers key advantages such as vertical take-off and landing (VTOL) capability and the ability to hover in place with high stability [13]. The operation of a quadrotor relies on precisely adjusting the power supplied to its four motors, enabling it to ascend or descend, tilt left or right (roll), tilt forward or backward (pitch), and rotate around its vertical axis (yaw), as illustrated in Figure 1.

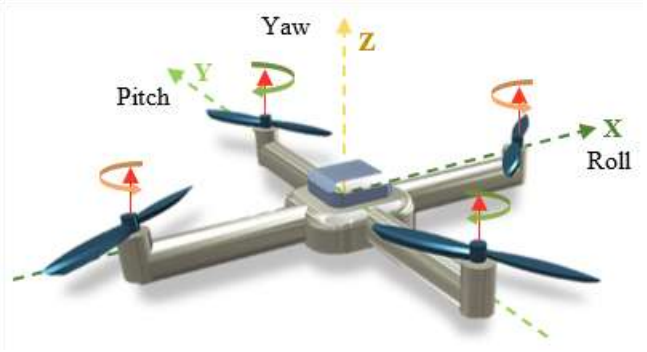


Figure 1. General structure of a quadrotor

In quadrotor modeling, using multiple coordinate frames is essential for accurately describing the motion and control of the vehicle. The three main coordinate frames are the inertial frame, vehicle frame, and body frame, each serving a specific purpose in the modeling process [14, 15]. These frames are illustrated in Figure 2.

- Inertial frame: A fixed reference frame (often earth-centered) used to describe the global position and orientation of the vehicle.
- Vehicle frame: A frame aligned with the inertial axes but moving with the UAV, useful for expressing navigation variables.
- Body frame: A frame attached to the UAV itself, used to describe forces, torques, and angular velocities relative to the vehicle.

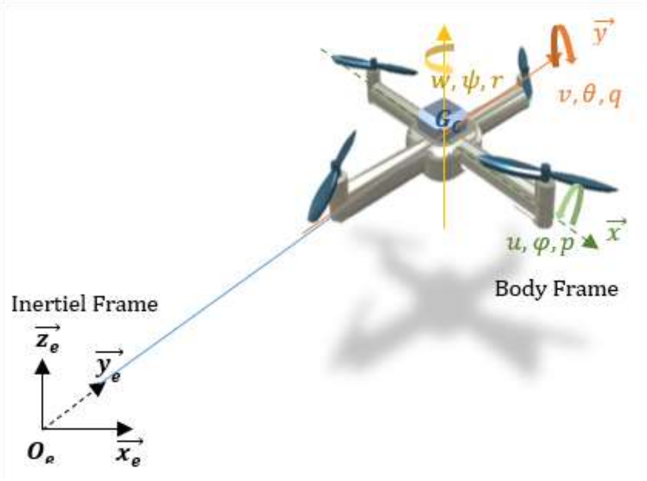


Figure 2. Coordinate frames used in quadrotor modeling

The attitude of the quadrotor relative to the earth-fixed frame is described by a commonly, used sequence of counter clockwise rotations about the axes of the earth-fixed coordinate system: yaw (ψ angle about the Z_I axis) , pitch (θ angle about the Y_I axis) and roll (φ angle about the X_I axis), with relative rotation matrices R_ψ, R_θ and R_ϕ, respectively [16], Then, the orthogonal rotation matrix $R(R^{-1} = R^T)$ that transforms a vector given in the inertial frame to its expression in the body-fixed reference frame, is given by (using the shorthand notation C for cosine and S for sine):

$$R(\psi,\theta,\phi) = \begin{bmatrix} C\psi C\theta & S\phi S\theta C\psi - S\psi C\phi & C\phi S\theta C\psi + S\psi S\phi \\ S\psi C\theta & S\varphi S\theta S\psi + C\psi C\theta & C\phi S\theta S\psi - S\phi C\psi \\ -S\theta & S\phi C\theta & C\phi C\theta \end{bmatrix} \quad (1)$$

To simplify the derivation of the quadrotor dynamic model and focus on the controller comparison, the following assumptions are made:

- Aerodynamic forces, which can be complex and difficult to model accurately, are neglected in this analysis.
- External aerodynamic forces are not included in the nominal model.
- Center of gravity is assumed perfectly centered.
- Rotor dynamics are simplified as first-order and symmetric, and the effect of motor saturation is neglected.
- Small-angle approximation is used in linearized models for attitude dynamics, which is valid for near-hover conditions.

The state, output and input vectors of the quadrotor model are chosen to describe all information about the system in the following way:

$$X = [x, \dot{x}, y, \dot{y}, z, \dot{z}, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T$$

where,

$[x, y, z]^T$: is the position in the inertial frame.

$[\dot{x}, \dot{y}, \dot{z}]^T$: is the linear velocity in the inertial frame.

$[\phi, \theta, \psi]^T$: is the Euler angles representing roll (ϕ), pitch (θ), and yaw (ψ).

$[\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$: Are the velocity angular.

The motion of a quadrotor is described by three translations (x, y, z) and three rotations (roll ϕ, pitch θ, yaw ψ) defined by Euler angles using the z-y-x sequence to express its orientation relative to the earth frame.

Under predefined assumptions, the dynamics of the quadrotor can be approximated as those of a rigid body in space, with additional aerodynamic forces generated by the rotation of the rotors. Using the newton–Euler formalism, the equations of motion can be expressed as follows:

$$\begin{cases} \dot{\xi} = \nu \\ m\ddot{\xi} = F_f + F_t + F_g \\ \dot{R} = RS(\Omega) \\ J\dot{\Omega} = \Gamma_f - \Gamma_a - \Gamma_{gm} - \Gamma_{gh} \end{cases} \quad (2)$$

$\xi = [x\, y\, z]^T$ is the translation movement vector. $\Omega = [\Omega_1\, \Omega_2\, \Omega_3]^T$ Contains the angular speed, m is the quadrotor mass while $j = diag[I_x I_y I_z]$ is the inertia matrix, v linear velocity, F force vector.

Eq. (3) calculates the translation movement based on factors including mass, aerodynamic forces, propulsion forces.

$$m\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} C(\phi)\, C\, S(\theta) + S(\phi)\, S\,(\psi) \\ C(\phi)\, S(\theta)\, S(\psi) - S(\phi)\, C(\psi) \\ C(\phi)\, C\,(\theta) \end{bmatrix} \sum_{i=1}^{4} F_i \begin{bmatrix} k_{ftx}\dot{x} \\ k_{fty}\dot{y} \\ k_{ftz}\dot{z} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (3)$$

Therefore, the equations describing the dynamics of translational motion are:

$$\begin{cases} \ddot{x} = \frac{1}{m}(C(\phi)\,C(\psi)\,S(\theta) + S(\phi)\,S(\psi))(\sum_{i=1}^{4} F_i) - \frac{K_{ftx}}{m}\dot{x} \\ \ddot{y} = \frac{1}{m}(C(\phi)\,S(\theta)\,S(\psi) - S(\phi)\,C(\psi))(\sum_{i=1}^{4} F_i) - \frac{K_{fty}}{m}\dot{y} \\ \ddot{z} = \frac{1}{m}(C(\phi)\,C(\theta))(\sum_{i=1}^{4} F_i) - \frac{K_{ftz}}{m}\dot{z} - g \end{cases} \tag{4}$$

We have the formula for rotational motion:

$$J\dot{\Omega} = -\Gamma_{gm} - \Gamma_{gh} - \Gamma_a + \Gamma_f \tag{5}$$

Replacing the moments with their corresponding formulas, we get:

$$\begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = -\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \wedge \left(\begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}\right) - \begin{bmatrix} J_r\bar{\Omega}_r\dot{\theta} \\ -J_r\bar{\Omega}_r\dot{\phi} \\ 0 \end{bmatrix} - \begin{bmatrix} K_{fax}\dot{\phi}^2 \\ K_{fay}\dot{\theta}^2 \\ K_{faz}\dot{\psi}^2 \end{bmatrix} + \begin{bmatrix} d(\omega_3^2 - \omega_1^2) \\ d(\omega_4^2 - \omega_2^2) \\ K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \tag{6}$$

where, $\bar{\Omega}_r = \omega_1 - \omega_2 + \omega_3 - \omega_4$.

Therefore, the equations of rotational motion are:

$$\begin{cases} \ddot{\phi} = \frac{1}{I_x}[\dot{\theta}\dot{\psi}(I_y - I_z) - K_{fax}\dot{\phi}^2 - J_r\bar{\Omega}_r\dot{\theta} + dU_2] \\ \ddot{\theta} = \frac{1}{I_y}[\dot{\phi}\dot{\psi}(I_z - I_x) - K_{fay}\dot{\theta}^2 + J_r\bar{\Omega}_r\dot{\phi} + dU_3] \\ \ddot{\psi} = \frac{1}{I_z}[\dot{\phi}\dot{\theta}(I_x - I_y) - K_{faz}\dot{\psi}^2 + U_4] \end{cases} \tag{7}$$

The complete dynamics of the quadrotor model, based on Eqs. (3)–(6) can be represented in Eq (8):

$$\begin{cases} \ddot{x} = \frac{1}{m}(C(\phi)\,C(\psi)\,S(\theta) + S(\phi)\,S(\psi))U_1 - \frac{K_{ftx}}{m}\dot{x} \\ \ddot{y} = \frac{1}{m}(C(\phi)\,S(\theta)\,S(\psi) - S(\varphi)\,C(\psi))U_1 - \frac{K_{fty}}{m}\dot{y} \\ \ddot{z} = \frac{1}{m}(C(\phi)\,C(\theta))U_1 - \frac{K_{ftz}}{m}\dot{z} - g \\ \ddot{\phi} = \frac{1}{I_x}[\dot{\theta}\dot{\psi}(I_y - I_z) - K_{fax}\dot{\phi}^2 - J_r\bar{\Omega}_r\dot{\theta} + dU_2] \\ \ddot{\theta} = \frac{1}{I_y}[\dot{\phi}\dot{\psi}(I_z - I_x) - K_{fay}\dot{\theta}^2 + J_r\bar{\Omega}_r\dot{\phi} + dU_3] \\ \ddot{\psi} = \frac{1}{I_z}[\dot{\phi}\dot{\theta}(I_x - I_y) - K_{faz}\dot{\psi}^2 + U_4] \end{cases} \tag{8}$$

System inputs are represented in Eq. (9). U_1, U_2, U_3, U_4 representing entries control for altitude, roll, pitch and yaw, respectively.

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} K_p & K_p & K_p & K_p \\ 0 & -dK_p & 0 & dK_p \\ -dK_p & 0 & dK_p & 0 \\ K_d & -K_d & K_d & -K_d \end{bmatrix} \times \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{9}$$

For a physical system, there are multitudes of state representations; it all depends on the choice of the state vector. For model represented by Eq. (8), developed previously and for a state vector.

$$X = [x, \dot{x}, y, \dot{y}, z, \dot{z}, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T$$
$$X = [x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9 x_{10} x_{11} x_{12}]^T$$

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{K_{ftx}}{m}x_2 + \frac{1}{m}U_xU_1 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -\frac{K_{fty}}{m}x_4 + \frac{1}{m}U_yU_1 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = -\frac{K_{ftz}}{m}x_6 + \frac{1}{m}(C(x_7)\,C(x_9))U_1 - g \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = -\frac{K_{fax}}{I_x}x_8^2 + \frac{(I_y - I_z)}{I_x}x_{10}x_{12} - \frac{J_r}{I_x}\bar{\Omega}_r x_{10} + \frac{d}{I_x}U_2 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = -\frac{K_{fay}}{I_y}x_{10}{}^2 + \frac{(I_z - I_x)}{I_y}x_8x_{12} + \frac{J_r}{I_y}\bar{\Omega}_r x_8 + \frac{d}{I_y}U_3 \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = -\frac{K_{faz}}{I_z}x_{12}{}^2 + \frac{(I_x - I_y)}{I_z}x_8x_{10} + \frac{1}{I_z}U_4 \end{cases} \tag{10}$$

We notice that the Euler angles and their time derivatives are independent of the translation components, unlike the translations, which depend on the Euler angles, the quadrotor is an underactuated system with four inputs vs six outputs, and strongly coupled. Therefore, it is important to introduce virtual inputs based like in Eq. (11), to get the desired angles; we can divide our system into two sub-systems [17]:

$$\begin{cases} U_x = C(x_7)\,C(x_{11})\,S(x_9) + S(x_7)\,S(x_{11}) \\ U_y = C(x_7)\,S(x_9)\,S(x_{11}) - S(x_7)\,C(x_{11}) \end{cases} \tag{11}$$

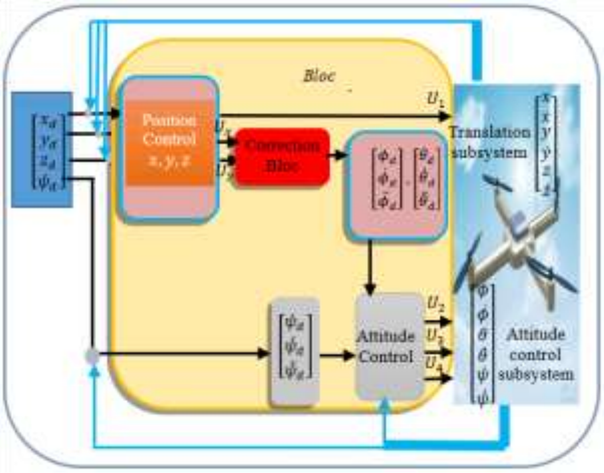

Bloc
x_d y_d z_d ψ_d
Position Control x, y, z
U_1
U_x
U_y
Correction Bloc
ϕ_d $\dot{\phi}_d$ $\ddot{\phi}_d$, θ_d $\dot{\theta}_d$ $\ddot{\theta}_d$
Translation subsystem
x $\dot{x}$ y $\dot{y}$ z $\dot{z}$
ψ_d $\dot{\psi}_d$ $\ddot{\psi}_d$
Attitude Control
U_2 U_3 U_4
ϕ $\dot{\phi}$ θ $\dot{\theta}$ ψ $\dot{\psi}$
Attitude control subsystem

Figure 3. Synoptic scheme of the adopted control quadrotor system

The scheme representing the adopted control quadrotor system is shown in Figure 3. Which is based on a two-loop cascade structure. From the desired positions x_d and y_d, the outer loop computes the desired roll (ϕ^d) and pitch (θ^d) angles, which are then provided as references to the inner loop, The advantage of this internal control loop is that it calculates the necessary thrusts to reach the desired altitude and attitude, then transmits them as command inputs to the system ($U_1, U_x, U_y, U_2, U_3, U_4$).

Table 1. The design parameters for the quadrotor model

Design Paramatre	Numerical Value	Name
m	1 kg	Quadrirotor mass
g	9.806 m/s^2	Gravity
K$_p$	2.9842*10^{-5} N/rad/s	Lift coefficient
K$_d$	2.2320*10^{-7} N.m/rad/s	Drag coefficient
Jr	2.8385*10^{-5} kg.m^2	Inertie of rotor
Km	4.3*10^3 N.m/A	Torque constant
Ke	2.16*10^{-3} N.m/A	Electrical torque constant
d	0.30 m	Distance between the center of gravity and the motor's axis of rotation

Quadrotor model parameters are as described in Table 1.

3. CONVENTIONAL CONTROL USING PID CONTROLLER

A PID controller is employed to stabilize the quadrotor. Its main advantages include straightforward implementation and simple tuning. The standard form (Figure 4) of the PID controller is expressed as follows [18-20]:

$$e(t) = x_d(t) - x(t) \quad (12)$$

$$U = K_p.e(t) + K_i.\int e(t).dt + K_d.\left(\frac{de(t)}{dt}\right) \quad (13)$$

The PID control strategy is certainly the most intuitive and the simplest to implement on a processor. It makes it easy to understand the physical role of each control term, which in turn facilitates the tuning of the gains accordingly [21].

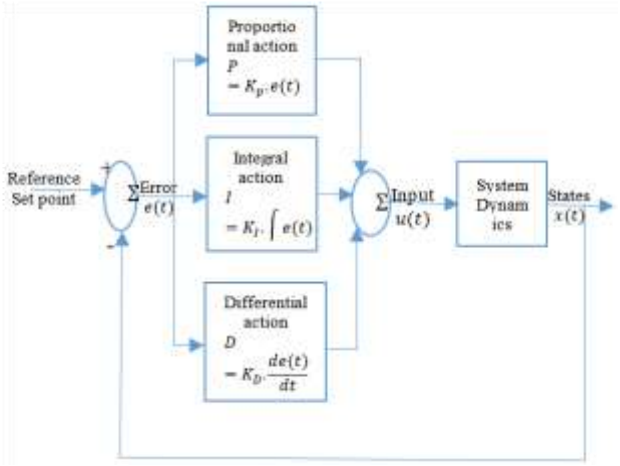


Figure 4. Structure controller PID

In a quadrotor system, the state variables must be precisely controlled to maintain stability during flight. Small deviations in the reference values can cause large variations in motion, leading to trajectory errors and reduced controllability. Although the pilot provides the main control inputs, external disturbances during flight can still destabilize the system. To address this, an internal flight controller is required to automatically adjust the rotor speeds and stabilize the vehicle. The system has four control inputs thrust, roll, pitch, and yaw that regulate six state variables. Thrust influences the vertical motion (z), roll affects the lateral motion (x) and ϕ, pitch controls the longitudinal motion (y) and pitch angle (θ), while yaw controls the heading angle (ψ). Therefore, six PID controllers are used, with a distribution of 1-2-2-1, corresponding to one controller for thrust, two for roll (position and angle), two for pitch (position and angle), and one for yaw respectively.

For the PID controllers, each channel (roll ϕ, pitch θ, yaw ψ, and altitude z) was tuned individually. The Ziegler–Nichols (Z-N) method was used as an initial approach: proportional gains were increased until sustained oscillations occurred, from which the ultimate gain (K_u) and oscillation period (T_u) were measured. These values were then used to compute the PID gains (K_p, K_i, K_d) according to the Z-N tuning rules. Minor adjustments were performed through simulations to account for the multivariable coupling of the quadrotor.

3.1 Attitude and altitude control

3.1.1 Altitude control

To regulate the quadrotor's altitude, a PID controller is designed. It generates the control input U_1, which governs the vertical motion of the quadrotor based on the altitude error. The resulting control law is derived as follows:

$$U_1 = K_p(z_d - z) + K_d(\dot{z}_d - \dot{z}) + K_i \int (z_d - z)\, dt \quad (14)$$

3.1.2 Attitude control

- Roll Controller

To control the roll angle ϕ of the quadrotor, an additional PID controller is used. It generates the input U_2, which adjusts the roll motion, as shown in Eq. (15).

$$U_2 = K_p(\phi_d - \phi) + K_d(\dot{\phi}_d - \dot{\phi}) + K_i \int (\phi_d - \phi)\, dt \quad (15)$$

- Pitch Controller

A PID controller is developed to regulate the quadrotor's pitch angle θ. The corresponding control law produces the input U_3, as given in Eq. (16).

$$U_3 = K_p(\theta_d - \theta) + K_d(\dot{\theta}_d - \dot{\theta}) + K_i \int (\theta_d - \theta)\, dt \quad (16)$$

- Yaw Controller

An additional PID controller is used to manage the yaw angle ψ of the quadrotor. It generates the input U_4 as shown in Eq. (17).

$$U_4 = K_p(\psi_d - \psi) + K_d(\dot{\psi}_d - \dot{\psi}) + K_i \int (\psi_d - \psi)\, dt \quad (17)$$

3.2 Proportional integral derivative controller simulation

The simulation results of the PID controller are presented in the following figures. Figure 5 illustrates the step input response of the PID controller. Table 2 summarizes the key performance indicators rise time, percentage overshoot, and settling time for the angler roll, pitch, yaw and the displacement z . These parameters provide essential information about the controller’s performance, accuracy, and stability.

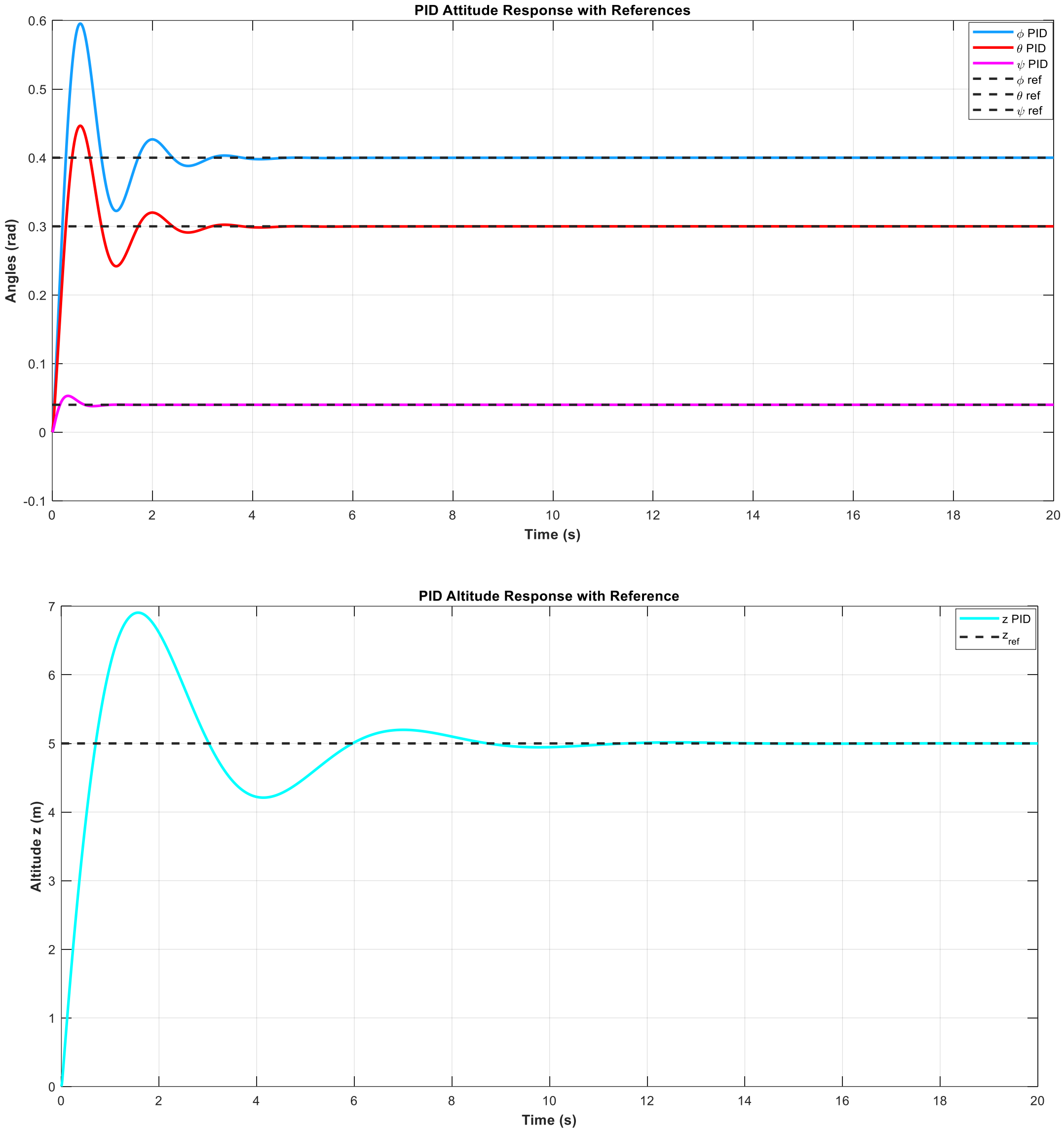


Figure 5. Step responses for PID controller

Table 2. Transient performance for step input when using PID controller

Response Parameter	Phi (Roll)	Theta (Pitch)	Psi (Yaw)	z (Thrust)
Settling time (s)	5 s	6 s	2.8 s	7 s
Max Overshoot M	38%	43%	30%	40%
Peak time (s)	0.5 s	1 s	0.6 s	1.5 s

The PID controller effectively stabilizes both the orientation (roll, pitch, and yaw) and the thrust (Z) of the system. Roll and pitch show moderate overshoot (about 38% and 43% respectively) and settle in about 5-6 s, yaw is the most stable, showing the lowest overshoot (30%) and fastest settling time (about 2.8 s) and thrust shows a relatively high overshoot (around 40%) but slowest settling time (about 7 s). Overall, the PID achieves fast convergence and acceptable transient performance; although the thrust axis could be tuned further to reduce overshoot.

4. LINEAR QUADRATIC REGULATOR

In recent years, modern control theory has exerted a significant influence on the aircraft industry. Among its methodologies, the LQR employs a state-space framework for system analysis and design. This approach facilitates the treatment of multi-output systems and enables stabilization through full-state feedback control [22, 23].

The LQR aims to minimize a cost function while constraining the state X of the system to follow the desired trajectory X_d, take into account the dynamic system [24, 25].

$$\begin{cases} \dot{x} = Ax + Bu \\ \quad y = Cx \end{cases} \tag{18}$$

Based on Eq. (18) the feedback signal used in this control system is the matrix K with optimal control, which is:

$$u = -Kx \tag{19}$$

To generate fresh equations:

$$\dot{x} = (A - BK)x + BKx \tag{20}$$

The cost function for this optimal problem is given by:

$$J(t) = \int_0^\infty x^T(t)Qx(t) + u^T(t)Ru(t)dt \tag{21}$$

Q and R are weighting matrices. They are defined as positive matrices, and their elements are selected to provide relative weighting to the individual state variables and to the control input of the system.

The first step is to choose the weight values of matrices of Q and R, using Eq. (18), the optimal full-state feedback gain K is obtained from the matrix algebraic equation.

$$K = -R^{-1}B^T P \tag{22}$$

where, P is derived from the Riccati equation:

$$PA + A^T + Q - PBR^{-1}B^T = 0 \tag{23}$$

Next, the functional diagram of the full-state feedback is shown in Figure 6.

We choose the matrices Q and R, and then implement the LQR control using the LQR function in MATLAB/Simulink R2024b.

Full-state feedback approach was adopted, where the control law is $u = -Kx$. The weighting matrices Q and R were initially chosen as diagonal matrices to optimize state deviations and control effort, respectively. The diagonal entries of Q were set higher for critical states (e.g., angles and altitude) to ensure fast and accurate tracking, while the entries of R were tuned to limit actuator saturation and ensure smooth control signals. Iterative simulations were used to fine-tune these matrices, balancing tracking performance and control effort.

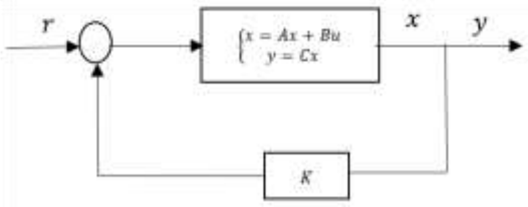


Figure 6. LQR block diagram

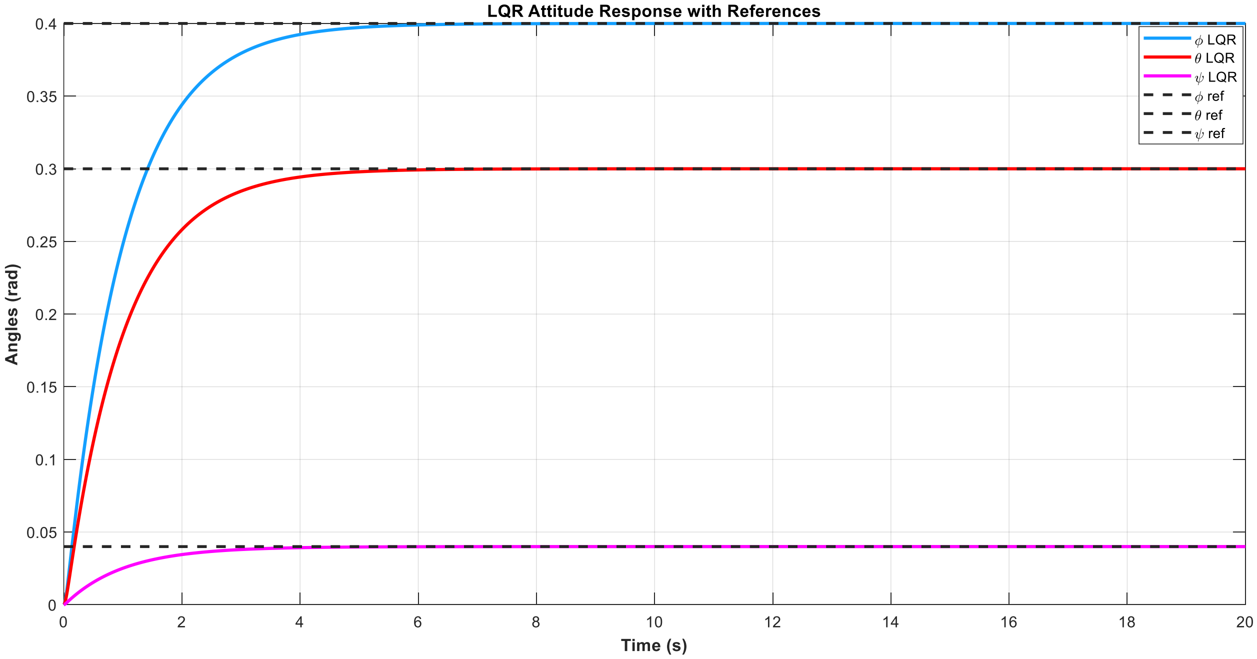
The tuned Q and R matrices which give the optimal response for the Quadrotor system parameters as defined before are:

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

And the feedback gain matrix K is computed as:

$$K = \begin{bmatrix} 1.0000 & 1.0321 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 1.0000 & 1.0321 & -0.0000 & -0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 & 1.0129 & 0.0000 & -0.0000 \\ -0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & 1.0000 & 1.5166 \end{bmatrix}$$



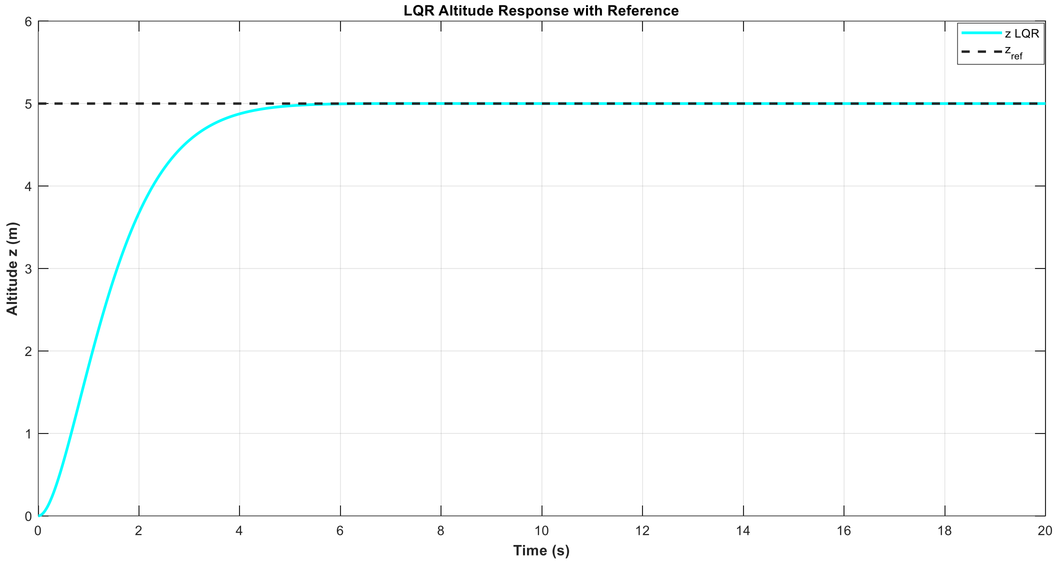


Figure 7. Step responses for LQR controller

Table 3. Transient performance for step input when using LQR controller

Response Parameter	Phi (Roll)	Theta (Pitch)	Psi (Yaw)	z (Thrust)
Settling time (s)	4 s	4.5 s	4.2 s	5 s
Max Overshoot M	0%	4%	0%	5%
Peak time (s)	4 s	1.5 s	2.5 s	2 s

4.1 LQR controller simulation

The simulation results obtained with the LQR controller are presented in the following figures. Figure 7 depicts the system response to a step input, highlighting the transient and steady-state behaviors. Table 3 provides a quantitative summary of the main performance indices, namely the rise time, overshoot, and settling time, evaluated for the angular motions roll, pitch, and yaw as well as the vertical displacement (z). These indicators offer valuable insights into the controller's dynamic performance, precision, and stability characteristics under the applied control strategy.

From the upper plot, it can be observed that the roll (ϕ), pitch (θ), and yaw (ψ) responses converge smoothly to their steady-state values within approximately 4-5 s. The overshoot is minimal, with only about 4% for pitch response and practically negligible for the roll and yaw response indicating a well-damped, and stable dynamic behavior. The system exhibits fast transient performance, with a peak time ranging between 1 and 2.5 s, which highlights the controller's ability to achieve accurate initial tracking. Likewise, the thrust (z) response shown in the lower plot settles in about 5 s, with a small overshoot (5%). These results collectively confirm the stability, smoothness, and effectiveness of the LQR controller in regulating both rotational motion and altitude system.

5. H_∞ CONTROLLER DESIGN

In the H_∞ control approach, the performance and robustness of the control system are mainly characterized by three closed-loop functions: the sensitivity function S, the complementary sensitivity function T, and the controller sensitivity function R which are defined by [26-28]:

$$\begin{cases} S(s) = [I + G(s)K(s)]^{-1} \\ T(s) = G(s)K(s)[I + G(s)K(s)]^{-1} \\ R(s) = K(s)[I + G(s)K(s)]^{-1} \end{cases} \tag{24}$$

As illustrated by the standard block diagram of Figure 8, $G(.)$ and $K(.)$ are the plant transfer matrix and feedback controller matrix, respectively. The functions $S(s)$, $T(s)$, and $R(s)$ represent, successively, the transfer matrix from the disturbance input d to the controlled system output y, the complement matrix $T = I - S$ and the transfer matrix from the disturbance d to the plant input u.

The design of the control system within the framework of H_∞ amounts to shaping S, T and R sensitivity functions to meet the following performance requirements:

To achieve overall disturbance attenuation together with good robustness against low frequency perturbations that may be caused by plant parameter uncertainty, small values of $S(j\omega)$ in a low frequency band are required. This is typically expressed by a 2-norm condition; that is, the largest singular value of the sensitivity function S, [26- 28]:

$$\bar{\sigma}\big(S(j\omega)\big) \ll 1 \tag{25}$$

The mitigation of disturbances and steady state requirements can be treated as follows:

$$\sigma(S(j\omega)) < \sigma(W_S^{-1}(j\omega)) \tag{26}$$

At high frequencies, the primary requirements involve rejecting measurement noise and ensuring robust stability against uncertainties arising from unmolded dynamics, nonlinearities, and system truncation effects, which typically become more pronounced as the operating frequency increases. Moreover, control effort should be minimized in this range, since no reference tracking is required. These objectives can be met by keeping the output control sensitivity as low as possible.

$$\sigma(K((j\omega)S(j\omega)) \ll 1 \tag{27}$$

By defining an upper bound on the maximum singular value of $K(s)\, S(s)$, the requirements above are satisfied if the function $W_{KS}(s) = 1/KS_{max(s)}$ expresses the desired shape for $K(s)\, S(s)$. Therefore, the following approaches can be used to handle the sensor noise rejection and bust stability [29, 30]:

$$\sigma(K((j\omega)S(j\omega)) < \sigma(W_{KS}^{-1}(j\omega)) \tag{28}$$

The control scheme for H_∞ is also shown in Figure 8. The problem robust performance is represented in mixed output sensitivity (S) , output control sensitivity (KS) , and complementary sensitivity (T) Often referred to as the H_∞ control problem to obtain an internally stabilizing controller.

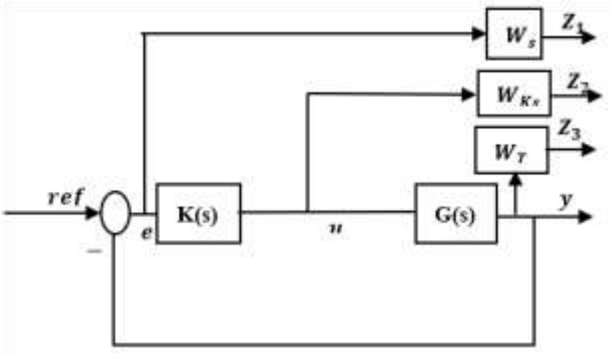


Figure 8. Diagram of H_∞ control

Consider the interconnection (illustrated in Figure 9) between the augmented plant P and the controller K. The closed-loop transfer function $T = f_l(P, K)$ from w to z can be determined by simple visual inspection.

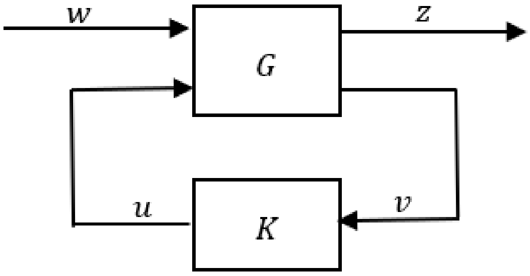


Figure 9. General control H_∞ problem

The system's input-output relationship in Figure 9 can be expressed as:

$$\begin{bmatrix} z(s) \\ y(s) \end{bmatrix} = P(s) \begin{bmatrix} w(s) \\ u(s) \end{bmatrix} = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} \begin{bmatrix} w(s) \\ u(s) \end{bmatrix} \tag{29}$$

where:

$$\begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} = \begin{bmatrix} W_s & -W_s * G \\ 0 & W_{ks} \\ 0 & W_T * G \\ 1 & -G \end{bmatrix} \tag{30}$$

The closed-loop transfer function from w to z can be found as:

$$f_l(P(s), K(s)) = P_{11} + P_{12} * K * (I - P_{22} * K)^{-1} * P_{21} \tag{31}$$

The mapping $f_l(P(s), K(s))$ is referred to as the linear fractional transformation [26-28]. The control objective is to determine a controller K such that the H_∞ norm of $f_l(P(s), K(s))$ remains below a specified constant γ. This constant γ defines the required performance level of the closed-loop system. In other words, the task is to design a controller K that ensures internal stability of the system while satisfying this performance bound [31].

$$\|f_l(P(s), K(s))\|_\infty < \gamma \tag{32}$$

A successful H_∞ controller synthesis minimizes the following H_∞ cost function and ensures an internally stable closed-loop system.

$$\|f_l(P(s), K(s))\|_\infty = \left\| \begin{matrix} W_s S \\ W_{ks} KS \\ W_T T \end{matrix} \right\| < \gamma = 1 \tag{33}$$

5.1 Selection of weighting function

One of the most critical and challenging tasks in robust controller design is selecting appropriate weighting functions. This process is often tedious and time-consuming, particularly when the system model includes complex nonlinearities that are neglected in the linearized version. The fundamental requirements for choosing weighting functions are outlined in Eqs. (25) and (28). Additionally, it is essential that the weighting functions are stable, compatible with nonminimum phase systems, and of reasonably low order; otherwise, the order of the resulting H_∞ controller will increase, as it is determined by the sum of the plant and the weighting function orders. Therefore, whenever possible, weighting functions of the lowest order that still satisfy the design specifications should be chosen.

After extensive simulation and fine tuning, the weighting functions W_s, W_{ks} and W_T are chosen as follows:

$$\begin{cases} W_s = \frac{0.6667s+0.5}{s+0.001} \\ W_{ks} = 0.0001 \\ W_T = \frac{0.5s+0.1}{s+10} \end{cases}$$

The Bode plots of the sensitivity function $S(s)$ and the complementary sensitivity function $T(s)$ are shown in Figure. 10, illustrating how the designed H_∞ controller meets the specified performance requirements.

5.2 H_∞ robust controller simulation

Figure 11 illustrates the tracking curve of reference input with H_∞ controller, Table 4 summarizes some transient performance parameters of the quadrirotor using the H_∞ controller. The simulation results indicate that the H_∞ controller eliminates overshoot, achieves a shorter settling time, and accurately tracks the desired response, demonstrating significant improvement in system performance.

From the upper plot, the roll (ϕ), pitch (θ), and yaw (ψ) responses under the H_∞ controller reach their steady state values within approximately 4 seconds, demonstrating a noticeably faster dynamic response compared to the LQR controller.

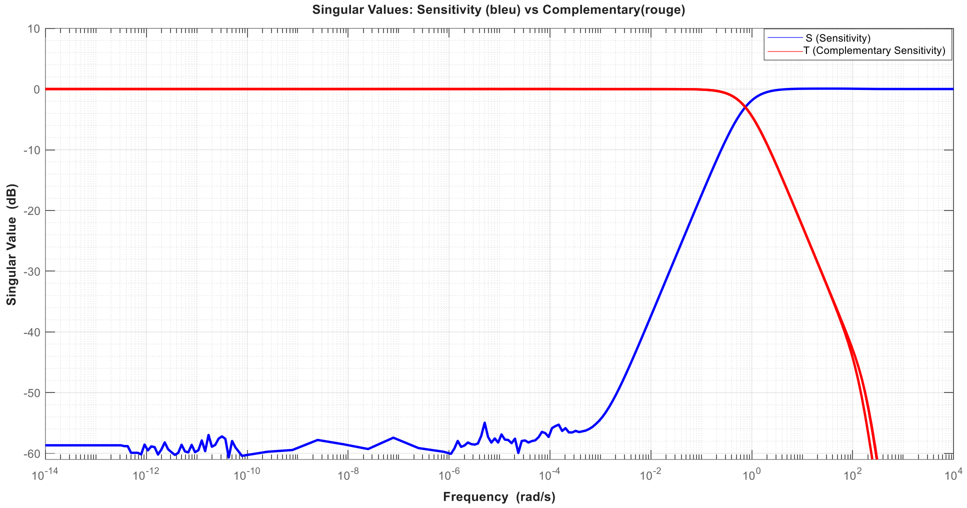


Figure 10. Bode plots of the sensitivity function $S(s)$ and complementary function $T(s)$ for the designed H_∞ controller

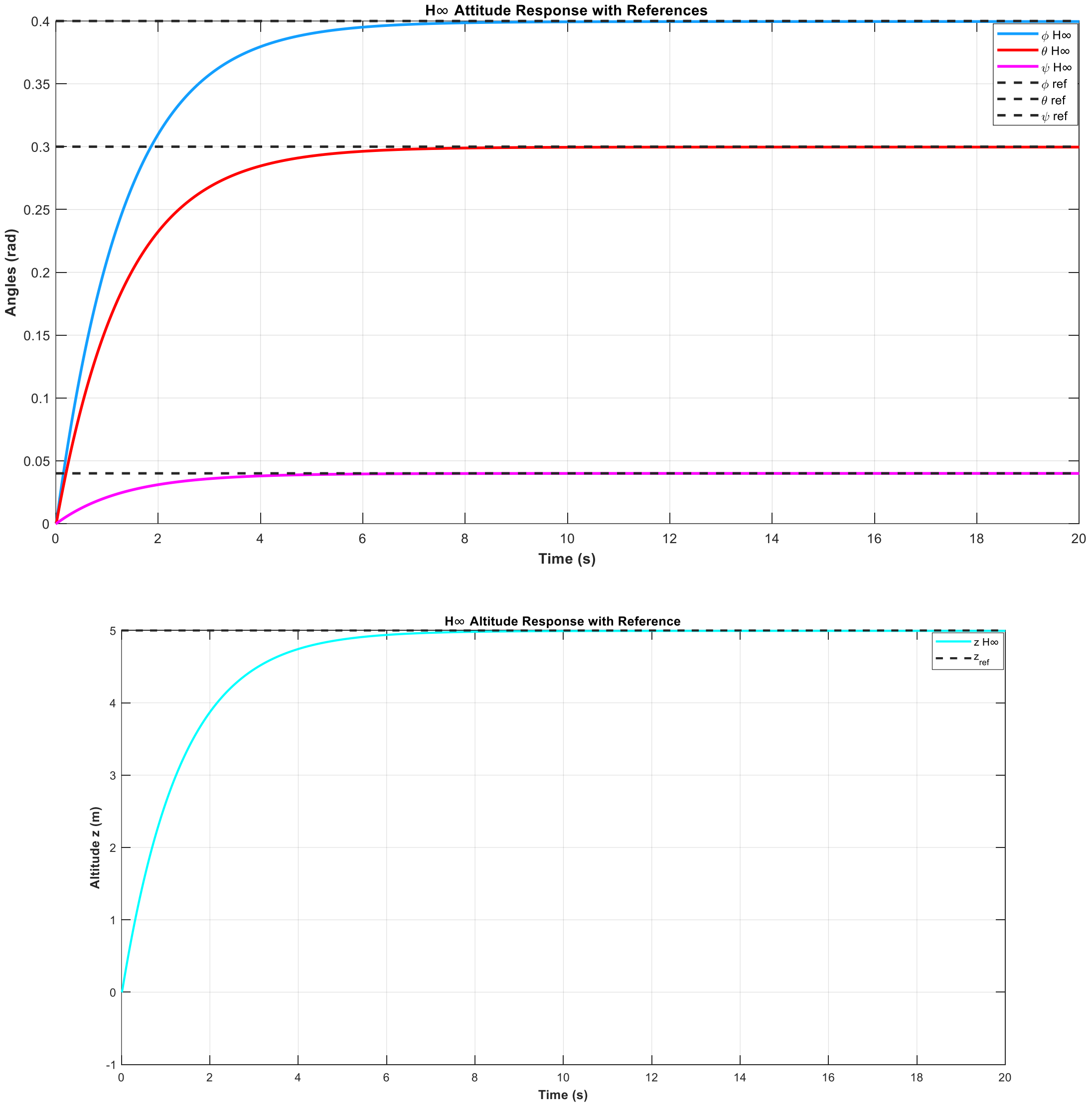


Figure 11. Step responses H_∞ controller

Table 4. Transient performance for step input when using H_∞ controller

Response Parameter	Phi (Roll)	Theta (Pitch)	Psi (Yaw)	z (Thrust)
Settling time (s)	3 s	9 s	3.6 s	9 s
Max overshoot M	1%	0%	0%	0%
Peak time (s)	1.2 s	0.3 s	1.8 s	0.2 s

The H_∞ controller provides highly damped and stable responses across all channels; the roll ϕ response shows a negligible overshoot 1% with a settling time 3 s, while both the pitch and yaw responses exhibit no measurable overshoot, indicating excellent damping, the corresponding peak times range from 0.3 s (pitch)to 1.8 s (yaw). The thrust (z) response is also free of overshoot but settles more slowly, around 9 s reflecting the controller's conservative and robust behavior. Overall, the H_∞ controller achieves strong robustness and smooth transients, with minimal overshoot at the expense of a slower altitude response.

5.3 Robust H_∞ with disturbance and noise

The robustness of the H_∞ controller was further evaluated by introducing external disturbance and measurement noise during the tracking task. A step disturbance was applied at t = 4 s, affecting all outputs simultaneously, while Gaussian measurement noise was added to simulate realistic sensor imperfections. The obtained results show that the H_∞ controller maintains stable tracking performance despite these adverse conditions. The attitude angles (roll, pitch, and yaw) rapidly reject the injected disturbances and converge back to their desired trajectories with minimal overshoot. Similarly, the altitude (z) response remains stable and exhibits strong disturbance attenuation. These results confirm that the designed H_∞ controller provides high robustness, ensuring reliable performance under model uncertainties, external perturbations, and noisy measurements, which is essential for real-time quadrotor operation. As shown in Figures 12 and 13.

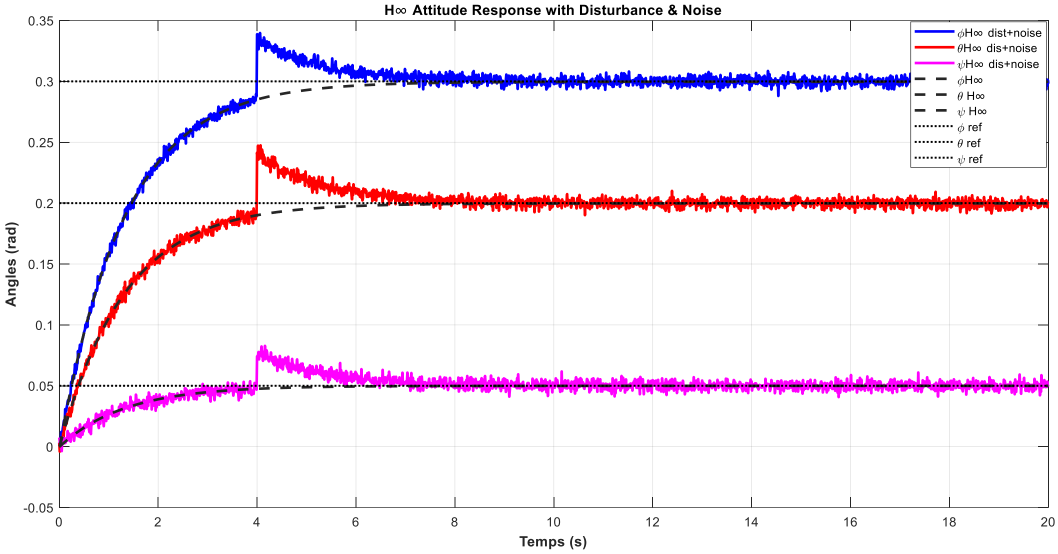


Figure 12. Attitude tracking under H_∞ control with disturbance and measurement noise

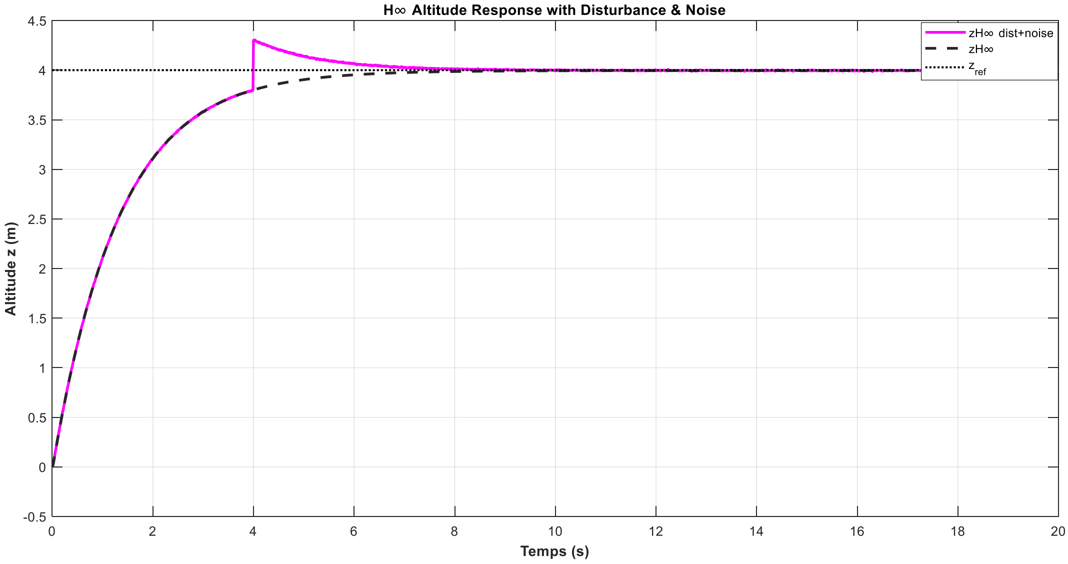


Figure 13. Altitude tracking under H_∞ control with disturbance and measurement noise

6. IMPLEMENTABILITY ON EMBEDDED SYSTEMS

The practical implementation of the three controllers is an important consideration on embedded hardware for coextensive quadrotor applications. Since the PID control only relies on basic arithmetic operations, namely addition, multiplication, and numeral differentiation. Therefore, it meets the requirements for small UAV platforms, since it can be implemented at high sampling frequencies (> 500 Hz) even on low-power micro controllers such as the STM32F1 or ATmega series. Moreover, at each sampling instant, the LQR control needs performing matrix multiplications between the state vector and the gain matrix. While it increases the computational load relative to the PID, the necessary operations remain moderate. Whereas, the H_∞ controller requires higher-order state-space models as well as evaluating multiple matrix operations online. Generally, these operations require more memory and processing power than the other two controllers. However, with the present embedded processors such as ARM Cortex-R7 or above, the H_∞ controller can be applied at medium sampling frequencies (around 100-200 Hz). The study of these three controllers showed that while PID efficiently provides embedded implementation, both LQR and H_∞ controllers remain practical with modern flight-control hardware, enabling a balance between computational complexity and robustness demands.

7. DISCUSSION AND RESULTS COMPARISON

The system response curves for the variables (ϕ, θ, ψ, z) under the influence of the three controllers (PID, LQR, H_∞) were represented superimposed, to facilitate an accurate comparison between the different control strategies.

Figure 14 illustrates the system's step response for the output variable ϕ when each of the different controllers is applied, with the aim of showing the impact of each control strategy on the system's dynamic performance. By comparing these responses, the response speed, stability, and amount of damping provided by each controller can be evaluated. Table 5 provides a detailed presentation of the distinctive dynamic characteristics of the step input response for the variable ϕ under the influence of different controllers, such as rise time, settling time, overshoot percentage, and other indicators for an accurate evaluation of each controller's performance.

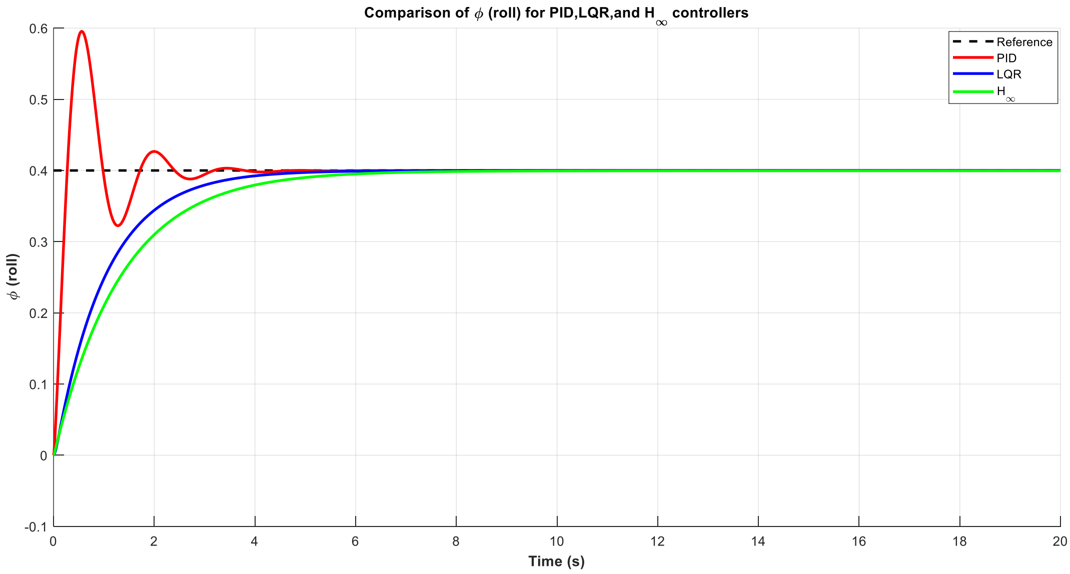


Figure 14. ϕ (roll) response to a step input using the different controllers

Table 5. The step response characteristics of the ϕ variable as produced by the different controllers

ϕ	PID	LQR	H_∞
Settling time (s)	5 s	4 s	3 s
Max overshoot M	38%	0%	1%
Peak time (s)	0.5 s	4 s	1.2 s

The roll tracking error for ϕ shown in Figure 15 indicates that the PID controller converges quickly but exhibits oscillations, the LQR provides a stable and smooth response with moderate convergence, while the H_∞ controller combines stability and robustness, keeping the error close to zero from the outset.

Figure 16 illustrates the system's step response for the output variable ψ when using different controllers, while Table 6 shows the dynamic characteristics of the step input response for the variable ψ under the influence of these controllers.

Table 6. The step response characteristics of the ψ variable as produced by the different controllers

ψ	PID	LQR	H_∞
Settling time (s)	2.8 s	4.2 s	3.6 s
Max overshoot M	30%	0%	0%
Peak time (s)	0.6 s	2.5 s	1.8 s

The yaw tracking error for ψ shown in Figure 17 indicates that the H_∞ offers the best combination of stability and robustness, PID is the fastest but with transient oscillations, and LQR constitutes a compromise between speed and stability.

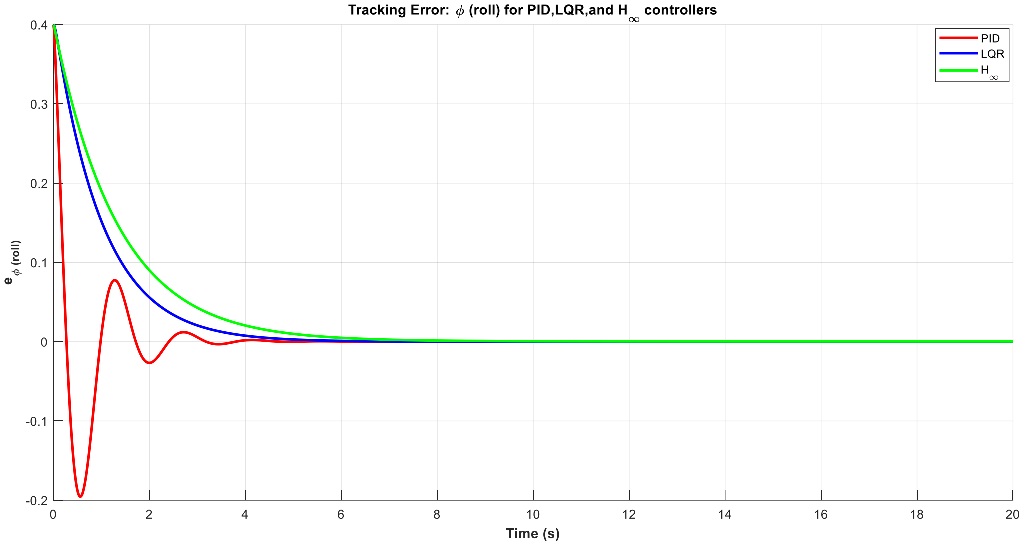


Figure 15. Tracking error ϕ (roll) using the different controllers

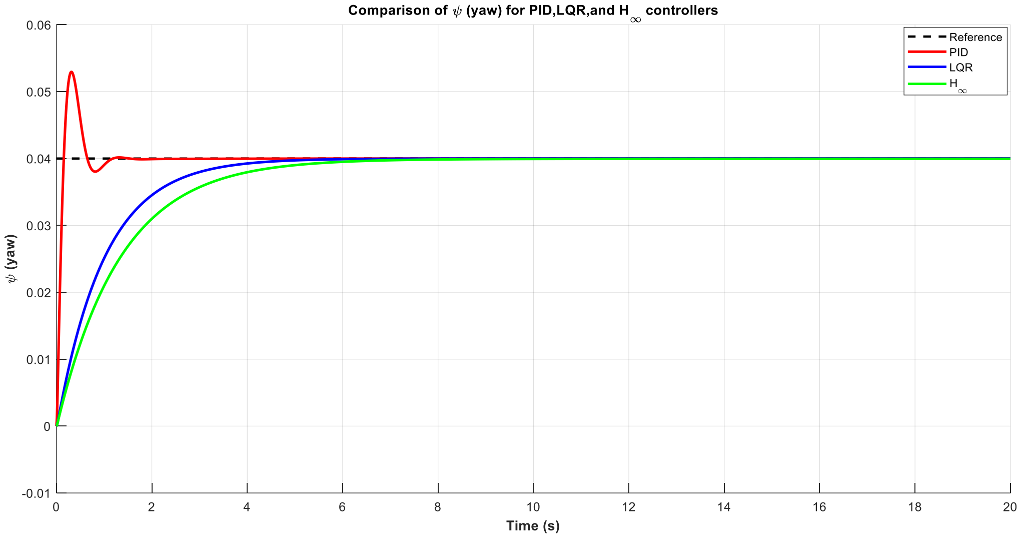


Figure 16. ψ (yaw) response to a step input using the different controllers

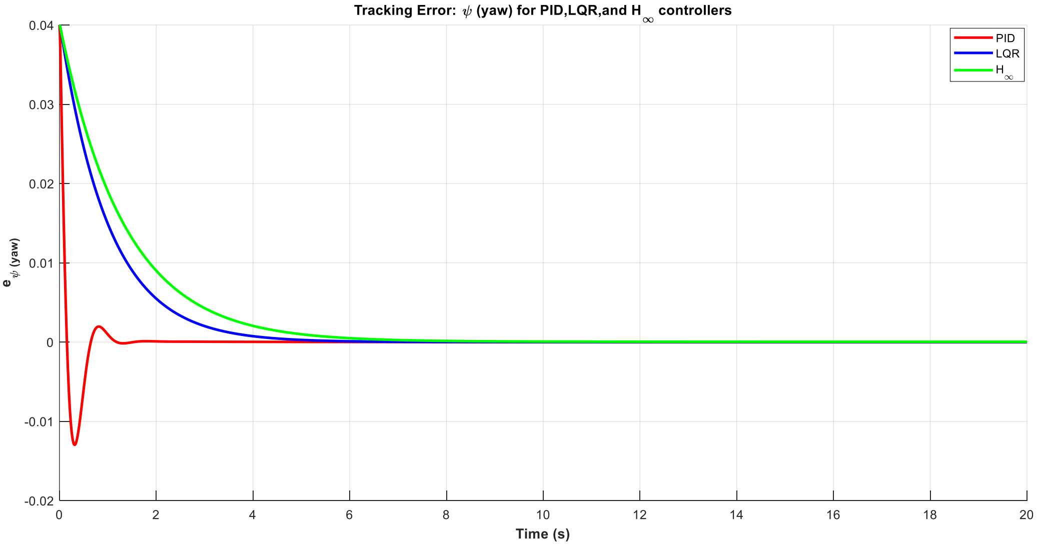


Figure 17. Tracking error ψ (yaw) using the different controllers

Figure 18 reflects how the system responds to a step in the output variable θ as a result of applying different controllers, while Table 7 illustrates the main dynamic parameters for the step input response of the same variable, such as peak time, settling time, and overshoot percentage.

The Figure 19 highlights that PID can respond quickly, while LQR and especially H_∞ provide better stability and smoother tracking of the pitch angle.

Figure 20 illustrates the step response of variable z using different controllers, and Table 8 presents the characteristics of the step input response for the same variable.

The altitude tracking error for z as shown in Figure 21 demonstrates that LQR represents a compromise between speed and stability, PID is the fastest but has oscillations in the transient regime, and H_∞ provides the optimum balance between stability and robustness.

Table 7. The step response characteristics of the θ variable as produced by the different controllers

$\boldsymbol{\theta}$	PID	LQR	$\boldsymbol{H_\infty}$
Settling time (s)	6s	4.5s	9s
Max overshoot M	43%	4%	0%
Peak time (s)	1 s	1.5 s	0.3 s

Table 8. The step response characteristics of the z variable as produced by the different controllers

$\boldsymbol{z}$	PID	LQR	$\boldsymbol{H_\infty}$
Settling time (s)	7 s	5 s	9 s
Max overshoot M	40%	5%	0%
Peak time (s)	1.5 s	2 s	0.2 s

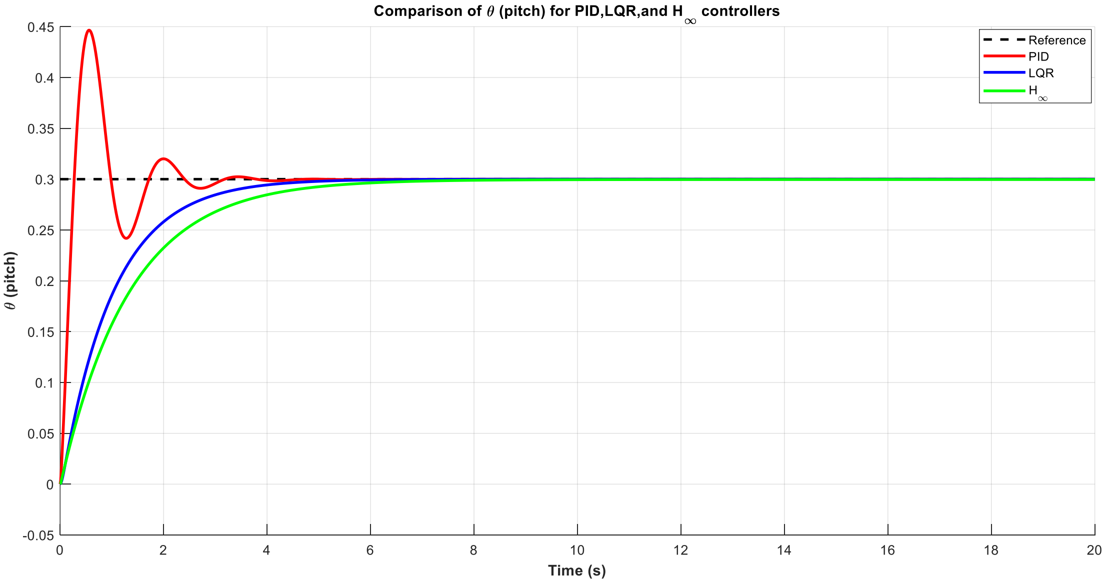


Figure 18. θ (pitch) response to a step input using the different controllers

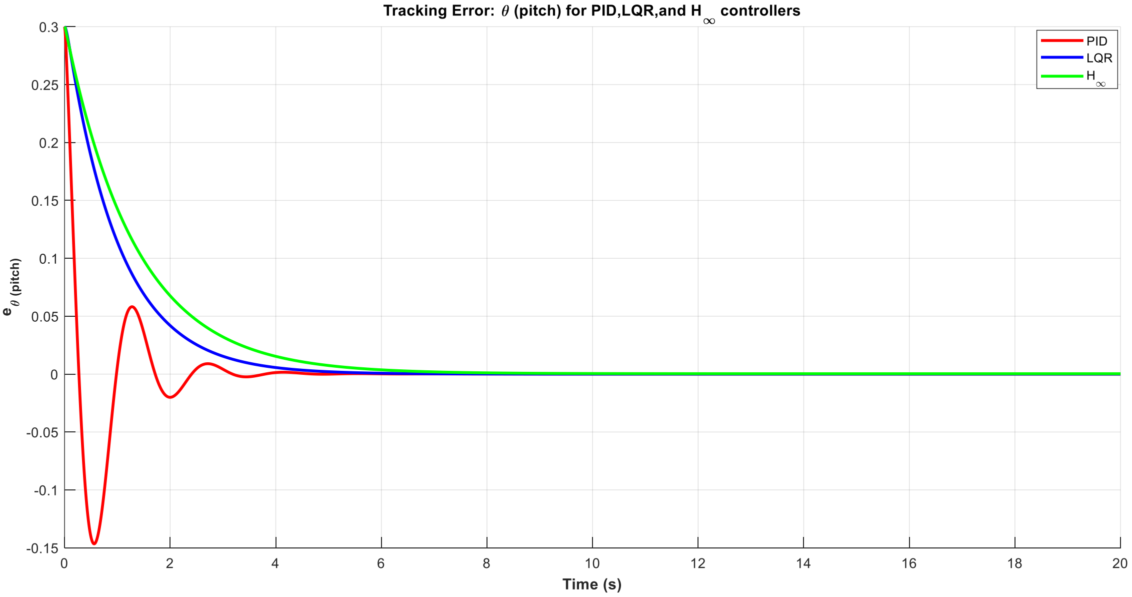


Figure 19. Tracking error θ (pitch) using the different controllers

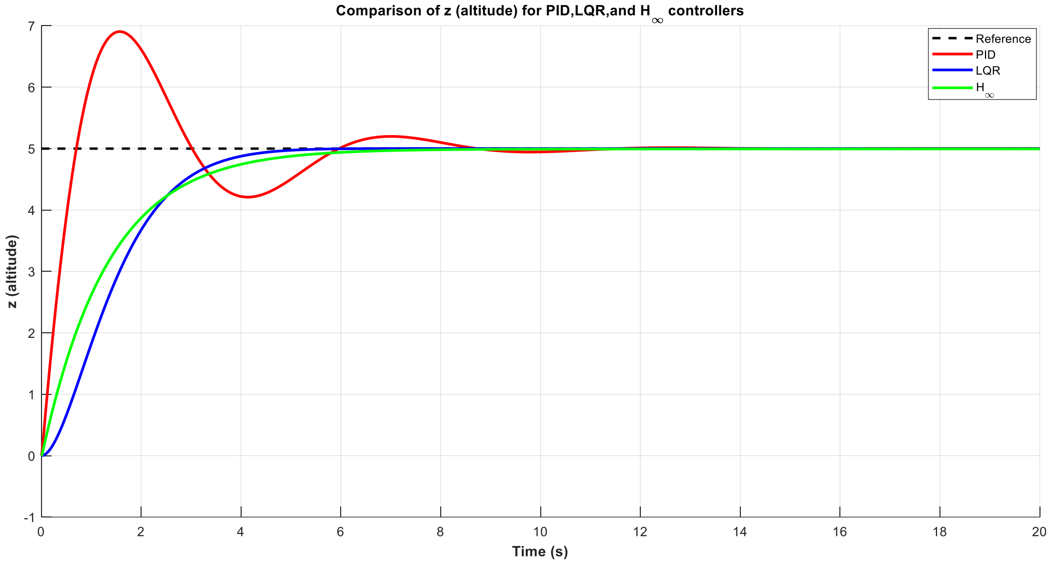


Figure 20. z (altitude) response to a step input using the different controllers

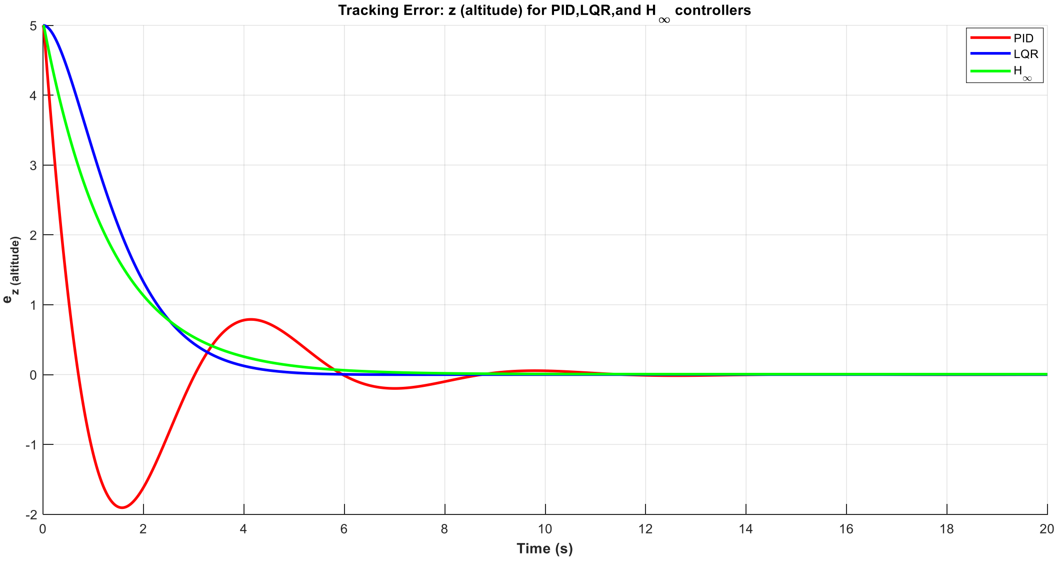


Figure 21. Tracking error z (altitude) using the different controllers

Table 9. Comparative analysis of control strategies for quadrotor systems: Advantages and limitations

Control Method	Advantages	Disadvantages
PID	Simple, effective for many problems, easy to implement	Manual tuning needed, poor with non-linearities and disturbances
LQR	Optimal for a specific cost, straightforward, effective for linear systems	Assumes perfect knowledge, poor with uncertainties, not for non-linear systems
H_∞ Control	Robust against uncertainties, handles multivariable, specific performance	Complex design, requires detailed model, high control effort

In conclusion, this paper offers a comprehensive evaluation of the strengths and weaknesses of the different control strategies analyzed, highlighting their relative effectiveness and suitability as revealed through the comparative study. The results demonstrate that while the conventional PID and LQR controllers provide satisfactory performance under nominal operating conditions, the H_∞ controller achieves superior robustness and faster dynamic response. This enhanced performance confirms its ability to effectively manage model uncertainties and external disturbances, making it a more reliable solution for ensuring stable and precise control of the quadrotor system. And provides a comprehensive assessment of the strengths and limitations of the three control strategies examined, as summarized in Table 9, The table clearly highlights the key differences between the controllers.

8. CONCLUSION

This study presented a comprehensive comparative analysis of three control strategies PID, LQR, and robust H_∞ applied to the attitude (ϕ, θ, ψ) and vertical position (z) control of a quadrotor unmanned aerial vehicle (UAV). The quadrotor, a multivariable and underactuated system capable of generating lift and control torques through its four rotors, represents a highly versatile platform for applications such as aerial surveillance, infrastructure inspection, and search-and-rescue operations.

The simulation results of this comparative analysis of three control PID, LQR, and robust H_∞ applied to the attitude (ϕ, θ, ψ) and vertical position (z) controller of quadrator unmanned aerial vehicle (UAV) presented the distinct strengths of each approach i.e. the PID control provides simple implementation and limited processing capacity, making it appropriate for real-time application. The LQR controller also excels in balancing between control effort and response quality, establishing stable and accurate tracking performance. However, by showing the most robust and reliable behavior under parameter variations and external disturbances, the H_∞ controller maintains stability and high precision even in uncertain and dynamic environments.

The comparative analysis demonstrates that the controllers are not considered absolutely optimal; instead, the choice is based on the operational objectives, system constraints, and environmental conditions. The results show that despite the complexity in design and computation, the H_∞ control approach ensures robustness and dynamic performance for applications requiring high reliability and persistence to uncertainty.

By providing a deeper comprehension of the comparative performance of classical, optimal, and robust control strategies for quadrotor systems, the study offers valuable insights and design guidelines for developing reliable and high-performance control architectures that extend to a wide range of other multi-input, multi-output (MIMO) aerospace and robotic systems under uncertain or challenging conditions.

REFERENCES

[1] Legowo, A., Sulaeman, E., Rosli, D. (2019). Review on system identification for quadrotor unmanned aerial vehicle (UAV). In 2019 Advances in Science and Engineering Technology International Conferences (ASET), Dubai, United Arab Emirates, pp. 1-8. https://doi.org/10.1109/ICASET.2019.8714531

[2] Jeler, G.E. (2019). Military and civilian applications of UAV systems. In International Scientific Conference Strategies XXI. The Complex and Dynamic Nature of the Security Environment-Volume 1, pp. 379-386.

[3] Utsav, A., Abhishek, A., Suraj, P., Badhai, R.K. (2021). An IoT based UAV network for military applications. In 2021 Sixth International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, pp. 122-125. https://doi.org/10.1109/WiSPNET51692.2021.9419470

[4] Bouaiss, O., Mechgoug, R., Ajgou, R. (2020). Modeling, control and simulation of quadrotor UAV. In 2020 1st International Conference on Communications, Control Systems and Signal Processing (CCSSP), El Oued, Algeria, pp. 340-345. https://doi.org/10.1109/CCSSP49278.2020.9151687

[5] Hamdy, O.M., Hassan, H.T. (2019). Modeling simulation and control of quadcopter using PID controller. In 4th International under-Graduate Research Conference, IUGRC.

[6] Rinaldi, M., Primatesta, S., Guglieri, G. (2023). A comparative study for control of quadrotor UAVs. Applied Sciences, 13(6): 3464. https://doi.org/10.3390/app13063464

[7] Elkhatem, A.S., Engin, S.N. (2022). Robust LQR and LQR-PI control strategies based on adaptive weighting matrix selection for a UAV position and attitude tracking control. Alexandria Engineering Journal, 61(8): 6275-6292. https://doi.org/10.1016/j.aej.2021.11.057

[8] Minervini, A., Godio, S., Guglieri, G., Dovis, F., Bici, A. (2021). Development and validation of a LQR-based quadcopter control dynamics simulation model. Journal of Aerospace Engineering, 34(6): 04021095. https://doi.org/10.1061/(ASCE)AS.1943-5525.0001336

[9] Saber, K., Elarkam, M., Zahir, A., Larabi, M.S., Colak, I. (2023). Robust H_∞ optimal control for longitudinal and lateral dynamics in small-scale helicopters. Journal Européen des Systèmes Automatisés, 56(6): 929-944. https://doi.org/10.18280/jesa.560604

[10] Hegde, N.T., George, V.I., Nayak, C.G., Vaz, A.C. (2021). Application of robust H-infinity controller in transition flight modeling of autonomous VTOL convertible Quad Tiltrotor UAV. International Journal of Intelligent Unmanned Systems, 9(3): 204-235. https://doi.org/10.1108/IJIUS-09-2020-0041

[11] Hamza, A., Mohamed, A.H., El-Badawy, A. (2022). Robust H-infinity control for a quadrotor UAV. In AIAA SCITECH 2022 Forum, AIAA 2022-2033. https://doi.org/10.2514/6.2022-2033

[12] Komiyama, S., Uchiyama, K., Masuda, K. (2025). Combined robust control for quadrotor UAV using model predictive control and super-twisting algorithm. Drones, 9(8): 576. https://doi.org/10.3390/drones9080576

[13] Ahmed, F., Kumar, P., Patil, P.P. (2016). Modeling and simulation of a quadcopter UAV. Nonlinear Studies, 23(4): 553-561.

[14] Ahmad, F., Kumar, P., Patil, P.P. (2018). Modeling and simulation of a quadcopter with altitude and attitude control. Nonlinear Studies, 25(2): 287-299.

[15] Chen, Y., He, Y., Zhou, M. (2013). Modeling and control of a quadrotor helicopter system under impact of wind field. Research Journal of Applied Sciences, Engineering and Technology, 6(17): 3214-3221.

[16] Castillo, P., Lozano, R., Dzul, A.E. (2005). Modelling and control of mini-flying machines. London: Springer London.

[17] Gherouat, O., Matouk, D., Hassam, A., Abdessemed, F. (2007). Sliding mode control for a quadrotor unmanned aerial vehicle. Journal of Automation & System Engineering, 10(3): 150-157.

[18] Bodrumlu, T., Soylemez, M.T., Mutlu, I. (2017). Modelling and control of the Qball X4 quadrotor system based on PID and fuzzy logic structure. In Journal of Physics: Conference Series, 783(1): 012039. https://doi.org/10.1088/1742-6596/783/1/012039

[19] Hadid, S., Zamoum, R.B., Refis, Y. (2025). Linear and nonlinear control design for a quadrotor. Bulletin of Electrical Engineering and Informatics, 14(2): 940-955. https://doi.org/10.11591/eei.v14i2.8234

[20] Şenkul, F., Altuğ, E. (2013). Modeling and control of a novel tilt—Roll rotor quadrotor UAV. In 2013 International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, pp. 1071-1076. https://doi.org/10.1109/ICUAS.2013.6564796

[21] Apriaskar, E., Fahmizal, F., Salim, N.A., Prastiyanto, D. (2019). Performance evaluation of balancing bicopter using P, PI, and PID controller. Jurnal Teknik Elektro, 11(2): 44-49.

[22] 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. https://doi.org/10.1016/j.matpr.2020.04.282

[23] Martins, L., Cardeira, C., Oliveira, P. (2019). Linear quadratic regulator for trajectory tracking of a quadrotor. IFAC-PapersOnLine, 52(12): 176-181. https://doi.org/10.1016/j.ifacol.2019.11.195

[24] Jafari, J., Ghazal, M., Nazemizadeh, M. (2014). A LQR optimal method to control the position of an overhead crane. IAES International Journal of Robotics and Automation, 3(4): 252-258.

[25] Acakpovi, A., Fifatin, F.X., Aza-Gnandji, M., Kpadevi, F., Nyarko, J. (2020). Design and implementation of a quadcopter based on a linear quadratic regulator (LQR). Journal of Digital Food, Energy & Water Systems, 1(1). https://doi.org/10.36615/digitalfoodenergywatersystems.v1i1.409

[26] Kwakernaak, H. (1993). Robust control and H_∞ -optimization-tutorial paper. Automatica, 29(2): 255-273.

[27] Khalil, I.S., Doyle, J.C., Glover, K. (1996). Robust and Optimal Control (Vol. 2). New York: Prentice Hall.

[28] Zhou, K., Doyle, J.C. (1998). Essentials of Robust Control (Vol. 104). Upper Saddle River, NJ: Prentice hall.

[29] Alotaibi, J., Morales, R. (2018). Mixed-sensitivity H_∞ on-blade control. Proceedings of the 44th 943 K(s) European Rotorcraft Forum, Delft, The Netherlands, pp 1-8.

[30] Choudhary, S.K. (2017). Optimal feedback control of a twin rotor MIMO system. International Journal of Modelling and Simulation, 37(1): 46-53. https://doi.org/10.1080/02286203.2016.1233008

[31] Noormohammadi-Asl, A., Esrafilian, O., Arzati, M.A., Taghirad, H.D. (2020). System identification and H∞-based control of quadrotor attitude. Mechanical Systems and Signal Processing, 135: 106358. https://doi.org/10.1016/j.ymssp.2019.106358

NOMENCLATURE

W_{ks}	Control Sensitivity Weight
W_s	Sensitivity Weight
W_T	Complementary Sensitivity Weight
K_{ftx}	Translational friction coefficient along the X-axis
K_{fty}	Translational friction coefficient along the Y-axis
K_{ftz}	Translational friction coefficient along the Z-axis
K_{fax}	Aerodynamic drag coefficient along the X-axis
K_{fay}	Aerodynamic drag coefficient along the Y-axis
K_{faz}	Aerodynamic drag coefficient along the Z-axis