ABUAD Journal of Engineering Research and Development (AJERD)
ISSN (online): 2645-2685; ISSN (print): 2756-6811

Volume 8, Issue 2, 185-208

Analysis of Nonlinear Control Strategies for Quadrotor Stability and Trajectory Tracking

Johnson ABIOLA[1], Oluwafemi AJEWOLE[1], Philip ADEWUYI[2]

[1]*Mechatronics Engineering Department, Bells University of Technology, Ota, Ogun, Nigeria*
joabiola@bellsuniversity.edu.ng/okajewole@bellsuniversity.edu.ng

[2]*Electrical/Mechatronics Engineering, Wigwe University, Isiokpo, Rivers, Nigeria*
solaadewuyi@gmail.com

Corresponding Author: abiola.johnson@outlook.com, +2348079593532

Date Submitted: 16/04/2025
Date Accepted: 22/07/2025
Date Published: 23/07/2025

Abstract: *The quadrotor control as nonlinear and underactuated systems is a huge challenge in getting a fine stabilization and trajectory tracking. The present work seeks to introduce a novel hybrid control structure composed of an Internal Model Control-based Proportional-Integral (IMC-PI) controller, with the integration of a Sliding Mode Controller (SMC) and an Extended State Observer (ESO). The main purpose is to achieve improvement of stability and tracking performance under dynamic flight operations. A six degrees of freedom (6-DoF) dynamic model is developed under rigid body structure and proportional thrust-drag assumptions. An IMC-PI controller is initially employed to stabilize the Euler angles and vertical position but is not able to control the lateral positions (x and y). For this, a hybrid SMC-ESO method is incorporated for enhanced robustness along with precise path tracking. Simulation results reveal that while the stand-alone IMC-PI controller can stabilize altitude and orientation within 3 seconds, it cannot converge to the complete position. Nevertheless, the hybrid controller achieves very precise moving in complex trajectories (helical and figure-eight) within 25 seconds, which outperforms the stand-alone solution. The results confirm the viability of the hybrid controller in autonomous UAV operations where stability and precision are of particular importance.*

Keywords: *Quadrotor Control, IMC, PI Controller, Nonlinear Dynamic Inversion, SMC, ESO*

1. INTRODUCTION

A quadrotor is a type of unmanned aerial vehicle (UAV) having four motors in its design, and it is capable of vertical take-off and landing (VTOL) and making sharp turns. The design is generally in the form of two clockwise motors and two counterclockwise motors, providing equal torques to stabilize the plane when it is flying [1]. While structurally straightforward and mechanically balanced, the quadrotor is underactuated and significantly nonlinear with six degrees of freedom (DoF) and four control inputs (motor thrusts), thus stabilizing and controlling the quadrotor is a nontrivial issue [2], [3].

Precise control of roll, pitch, yaw, and translational motion is achieved by varying the rotational speeds of the individual motors. Such motions facilitate translation along the directions x–y–z as well as rotation about its three principal axes [4], [5]. Nevertheless, stable and precise control of these motions, particularly in the presence of external disturbances or model uncertainties, is a pervasive challenge in current UAV research. Advances in embedded systems, sensors, and control theory have pushed the application of quadrotors to new areas such as surveillance, delivery, precision agriculture, and monitoring of the environment [6]. As the number of applications continues to rise with added complexity, there is a pressing need for advanced control mechanisms that can cope with limitations of traditional controllers such as PID and LQR, which have a tendency to assume linearity and are not robust enough to handle dynamic nonlinearities or external disturbances in real-world environments [7].

While different nonlinear control techniques have been proposed, most of them either involve high computational complexity or lack robustness against system uncertainties and unmodeled dynamics [8]. Surprisingly, the traditional controllers such as PID are very efficient when dealing with near-hover maneuvers but are not up to the mark when aggressive maneuvers or operating on a trajectory tracking mission is involved [9]. On the other hand, sliding mode control (SMC) is disturbance robust but normally sensitive to chattering and may require accurate system modeling [10]. Besides, single internal model control (IMC)-based methods perform well for systems whose models are available but less so with unmodeled dynamics or disturbance rejection [11].

To address these concerns, in this work, a hybrid control scheme comprising an IMC-based PI controller and an SMC augmented by an Extended State Observer (ESO) is adopted. The IMC-based PI controller provides fast and stable stabilization of the Euler angles and vertical motion with relatively low complexity [12], [13]. The SMC augmented with

ESO is capable of overcoming external disturbances and model uncertainties effectively and can thus be utilized for accurate tracking of trajectories [14], [15]. The hybrid method benefits from the good points of both methods: IMC's predictive and tuning simplicity and SMC-ESO's robustness and disturbance rejection capability.

Accordingly, in this paper, a comprehensive mathematical model of the quadrotor is formulated, nonlinear vehicle dynamics are simulated, and the hybrid control approach is utilized for enhanced stabilization and precise path tracking during intricate maneuvers. This extends new insight to how hybrid control structures enhance trajectory tracking in underactuated UAV systems, and will be used as a foundation for future practical implementation under disturbance and noise.

2. LITERATURE REVIEW

Quadrotors or quadrotors have evolved significantly since their invention in the early 20th century. Manned vertical take-off and landing (VTOL) vehicles were the first successful models, but their performance was compromised due to lack of adequate stability augmentation and high pilot workload [7]. As time elapsed, advances in computational power, modeling techniques, and control theory enabled the transition from rudimentary linear control methods to sophisticated nonlinear methods, culminating in the creation of modern unmanned aerial vehicles (UAVs) [6]. The historical development of quadrotor technology is described in this paper, recent mathematical modeling and control algorithm advances are reviewed, and their applications in various fields are discussed.

2.1 Historical Development and Early Challenges

The pioneering quadrotors of the 1920s and 1930s were among the first VTOL aircraft to enjoy success. Instability and workload for the pilots were the prominent problems of early prototypes, which limited their deployment in practice [3]. It was only for the first time that linear control techniques were implemented to minimize computations and offer stability in flight. Although these approaches worked well for small perturbations near equilibrium points, they tended to miss the complete nonlinear dynamics of the system and thus performed sub-optimally in real-world applications [8].

2.2 Modeling of Quadrotor Dynamics

A full understanding of quadrotor dynamics requires complex mathematical modeling, both kinematics and dynamics. Kinematics prescribes the quadrotor motion without considering the acting forces, while dynamics prescribes the forces and moments which create this motion [9]. Nonlinear state equations were recently built on both Newton-Euler and Euler-Lagrange formulations, providing a robust basis for the study of quadrotor behavior subject to external disturbances [10]. For instance, [11] provided a lower-order model that did not capture all aerodynamic effects but was still able to stabilize and track using proportional-derivative (PD) controllers. The authors did note that unmodeled effects such as wind could compromise performance and that more robust techniques are required.

2.3 Stability and Trajectory Tracking Control Algorithms

The choice of the control algorithm has a direct bearing on determining the performance of a quadrotor. Traditional methods, i.e., proportional-integral-derivative (PID) control, are also common due to simplicity and robustness near hover states [12]. However, PID controllers suffer with large disturbances and nonlinear dynamics. To combat such limitations, newer methods such as sliding mode control (SMC), backstepping control, and adaptive control have been explored [13]. For example, [14] had blended a PD controller with heuristics for minimizing the influence of external perturbations so that the system tracks the path more effectively. Similarly, [15] recommended a hybrid controller that incorporates optimal elements of some algorithms, despite acknowledging that any one method may not be best under all aspects of performance metrics such as robustness, flexibility, and following accuracy. Yan et al. [28] proposed a multi-constraint model predictive control (MPC) approach that incorporates a Kalman-consistent filter alongside a fixed-time disturbance observer. This combination improves the controller’s robustness and suitability for complex environments, while also enhancing computational performance.

To create a closer comparison between other control methods, Table 1 summarizes popular control methods used in the application of quadrotor systems, evaluated with respect to stability, robustness, and computational load. This comparison highlights the importance of pursuing a hybrid IMC-based PI and SMC-ESO method since it achieves the best balance of simplicity, robustness, and performance in the regulation of the quadrotor's nonlinear and underactuated dynamics.

2.4 Applications and Future Directions

Quadrotors have also been discovered to exhibit various applications such as in surveillance, agriculture, and environment observation. An example is [14] developing a quadrotor for games count in African national parks in line with such properties as durability, agility, and minimal noise production. It was discovered by the study that the use of hybrid control systems presented a balance of performance against flexibility, while sacrifices are inevitable according to application. In addition, [16] developed a nonlinear flight mechanics model and implemented it with MATLAB/Simulink and demonstrated the effectiveness of automatic tuning methods, of Gradient Descent algorithm, compared to manual tuning. Existing work has also explored autonomous flight capabilities, with [17] proposing an integrated PID-backstepping collision avoidance system. This proposed solution enabled the quadrotor to perform complex maneuvers like take-off, hovering, and collision avoidance, which was a major advancement in UAV technology. Optimization of performance across various environmental conditions remains to be achieved, calling for further research into adaptive and robust control methods

Table 1: Comparison of various control techniques

Control Method	Stability	Robustness to Disturbance/Uncertainty	Computational Complexity	Remarks
PID	Moderate	Low to Moderate	Low	Works well near hover; poor performance under strong nonlinearities
Backstepping	High	Moderate	High	Handles nonlinearity but complex for multi-DoF systems
Sliding Mode Control (SMC)	High	High	Moderate	Robust but prone to chattering; effective for underactuated systems
Adaptive Control	High	High	High	Adjusts to parameter variations; may require persistent excitation
Model Predictive Control (MPC)	High	High	Very High	Handles constraints well; computationally expensive
IMC	Moderate	Low to Moderate	Low	Simple to implement; poor disturbance rejection without augmentation
ESO	High	Very High	Moderate to High	Combines robustness with online disturbance estimation

3. METHODOLOGY

3.1 Mathematical Modelling

To fully comprehend the quadrotor system dynamics, understanding the concept of six degrees of freedom (6-DoF) that outlines the position and orientation of the vehicle in three-dimensional (3D) space is critical. The 6-DoF model comprises three translational degrees of freedom motion along the x, y, and z axes, and three rotational degrees of freedom, roll, pitch, and yaw as shown in Figure 1. These freedoms of the quadrotor altogether decide the quadrotor's ability to navigate, stabilize, and perform complex flight maneuvers within a 3D space.

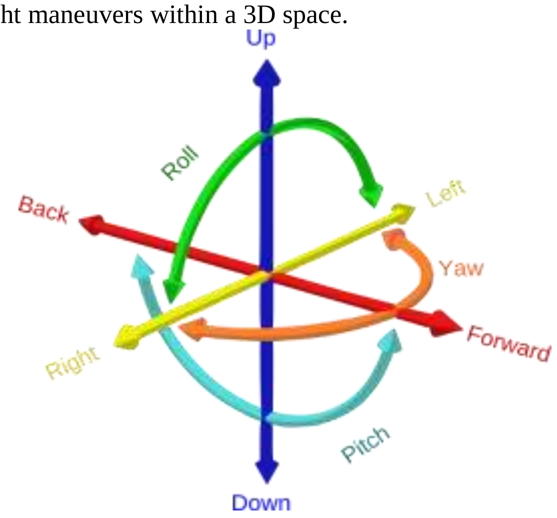


Figure 1: Six degrees of freedom [18]

6-DoF is the orientation and body position described in six coordinates divided into two frames of reference. The inertial frame (or earth frame), a stationary coordinate system defined by the axes x, y, and z, usually pointing in cardinal directions North, East, and Down, is the first one. The second is the body frame, which is a translating coordinate system relative to the quadrotor's center of gravity, specified by the roll (ϕ), pitch (θ) and yaw (ψ) angles. These are the orientation angles of the quadrotor relative to its body frame. This complexity of the system requires simplifying assumptions so that the system can be modeled and controlled effectively. Common assumptions are [19]:

i. Rigid structure: The frame of the quadrotor is considered rigid, with no deformation under flight.
ii. Axis symmetry: Symmetry occurs on the axes of the structure to facilitate easier analysis for forces and moments.
iii. Center of gravity alignment: The center of gravity is at the origin of the body-fixed frame.
iv. Rigid propellers: The rigid assumption is that the propellers don't bend or flex under operational conditions.
v. Thrust and drag proportionality: Both thrust and drag forces are proportional to the square of the rotational speed of the propeller.

3.1.1 Euler angles

Euler angles, which were named by Leonhard Euler, are three angles giving a description of the rigid body's orientation in a reference system. Euler angles tend to be useful in finding the link between two reference systems and changing the coordinates from one frame to another. For quadrotors, the Euler roll (ϕ), pitch (θ) and yaw (ψ) angles represent rotation around the x, y, and z axes of the body frame, respectively. These angles indicate by how much the quadrotor is tilted or rotated away from the initial position. Rigid body orientation can be found by multiplying these three Euler angles together. The rotation matrices, which transform coordinates from one frame to another, i.e the rotation about the x, y, and z-axes, are given by [20], [21]:

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi) & cos(\phi) \end{bmatrix} \tag{1}$$

$$R_y(\theta) = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -s(\theta) & 0 & c(\theta) \end{bmatrix} \tag{2}$$

$$R_z(\psi) = \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 \\ sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

The rotation matrix R, describes the link between the earth frame (inertial frame) and the body frame of the quadrotor is constructed using the Euler angles ϕ (roll), θ (pitch), and ψ (yaw) given as Equation (4).

$$R = R_z(\psi) \times R_y(\theta) \times R_x(\phi) \tag{4}$$

$$R = \begin{bmatrix} cos(\theta)cos(\psi) & sin(\phi)sin(\theta)cos(\psi) - cos(\phi)sin(\psi) & cos(\phi)sin(\theta)cos(\psi) + sin(\phi)sin(\psi) \\ cos(\theta)sin(\psi) & sin(\phi)sin(\theta)sin(\psi) + cos(\phi)cos(\psi) & cos(\phi)sin(\theta)sin(\psi) - sin(\phi)cos(\psi) \\ -sin(\theta) & sin(\phi)cos(\theta) & cos(\phi)cos(\theta) \end{bmatrix} \tag{5}$$

R is orthogonal such that $R^{-1} = R^T$.

3.1.2 Reference frame transformation

Let $[x\ y\ z\ \phi\ \theta\ \psi]^T$ represent a vector containing the positions in the inertial frame, Also, let $[u\ v\ w\ p\ q\ r]^T$ denote a vector of the velocities in the quadrotor body frame. Generally, the derivative of the angular positions should yield angular velocities [3]. However, since the angular positions and the velocities mentioned above are defined in different inertial and body frames (reference frames), a transformation matrix is required to convert quantities from one frame to the other. This transformation will ensure consistency and accuracy when analysing the quadrotor's motion. Also, let $\xi = [x\ y\ z]^T\ and\ \eta = [\phi\ \theta\ \psi]^T$, let $v_I = [\dot{x}\ \dot{y}\ \dot{z}]^T\ and\ \omega_I = [\dot{\phi}\ \dot{\theta}\ \dot{\psi}]^T$ and let $v_B = [u\ v\ w]^T\ and\ \omega_B = [p\ q\ r]^T$.

From this statement, $v_I \neq v_B\ and\ \omega_I \neq \omega_B$, instead:

$$v_I = R \cdot v_B \tag{6}$$

$$\omega_I = W_\eta^{-1} \cdot \omega_B \tag{7}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & sin(\phi)t(\theta) & cos(\phi)t(\theta) \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & \frac{sin(\phi)}{cos(\theta)} & \frac{cos(\phi)}{cos(\theta)} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{8}$$

Conversely,

$$\omega_B = W_\eta \cdot \omega_I \tag{9}$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -sin(\theta) \\ 0 & cos(\phi) & sin(\phi)cos(\theta) \\ 0 & -sin(\phi) & cos(\phi)cos(\theta) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{10}$$

Because rotational dynamics are computed in the body frame, but measurements (like position and velocity) are mostly in the inertial frame, a transformation is needed. The rotation matrix that can be derived from Euler angles offers this transformation so that control laws are aligned with physical orientation.

3.1.3 Rotational motion

The quadrotor is modeled as a rigid body, its dynamics can be described using Euler's equations for rigid body motion. These equations relate the angular velocities and moments acting on the body to its rotational dynamics. In the body-fixed frame, the dynamics equation is expressed as [22]:

$$I\dot{\omega}_B + [\omega_B \times (I\omega_B)] + \Gamma = \tau_B \tag{11}$$

It's assumed that the quadrotor is in a symmetric structure with the four arms closely with the body x and y-axes. Therefore, the inertia tensor I becomes diagonal and the moments of inertia about the x and y axes are equal ($I_{xx} = I_{yy}$).

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{12}$$

The gyroscopic movement (forces) Γ arise due to the combined rotational motion of the four motors and the body of the quadrotor, this is given in equation (13).

$$\Gamma = \sum_{i=1}^{4} J_r(\omega_B \wedge \hat{e}_3)(-1)^{i+1}\omega_{r_i} \tag{13}$$

In matrix form,

$$\Gamma = J_r\omega_B\omega_r \qquad where\ \omega_r = -\omega_1 + \omega_2 - \omega_3 + \omega_4 \tag{14}$$

The external torque,

$$\tau_B = [\tau_\phi\ \tau_\theta\ \tau_\psi]^T \tag{15}$$

The torque of the roll (τ_ϕ), and that of the pitch torque (τ_θ), are generated by the differential thrust of the motors. typically motor $i = 1$ and $i = 3$ are aligned with the roll-axis (x-axis) while motors $i = 2$ and $i = 4$ are aligned with the pitch-axis (y-axis).

$$\tau_\phi = \Sigma r \times T = l(-k_t\omega_2^2 + k_t\omega_4^2) = lk_t(-\omega_2^2 + \omega_4^2) \tag{16}$$

$$\tau_\theta = \Sigma r \times T = l(-k_t\omega_1^2 + k_t\omega_3^2) = lk_t(-\omega_1^2 + \omega_3^2) \tag{17}$$

For the yaw-axis, the torque that is created around the motor axis is a result of the reactive torques generated by the spinning propellers. Since the motor axis is aligned with the z-direction in the quadrotor body frame, the yaw torque (τ_ψ) is influenced by the speed of rotation of the motors and their direction of rotation. The yaw torque expression is given by equation (18).

$$\tau_\psi = (-1)^{i+1}k_b\omega_i^2 + I_m\dot{\omega}_i \tag{18}$$

where the term $(-1)^{i+1}$ is introduced to account for the clockwise (positive) and anticlockwise (negative) rotation of the i^{th} propeller. Additionally, $I_m\dot{\omega}_i$ is the moment of inertia of the motor and propeller, this can be ignored in steady-state condition, because $\dot{\omega}_i \approx 0$.

The overall torque about the z-axis is therefore simplify as equation (19).

$$\tau_\psi = k_b(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \tag{19}$$

The torque matrix is also therefore written as equation (20). Physically, each motor contributes to torque about the axes of the quadrotor based on its position and direction of spin. Roll and pitch torques result from differences in thrust between opposing motors, and yaw torque results from direction of spin and drag. Torque distribution equation expresses these as a matrix equation between motor angular velocities and net torque productions.

$$\tau_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} Lk_t(-\omega_2^2 + \omega_4^2) \\ Lk_t(-\omega_1^2 + \omega_3^2) \\ k_b(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \end{bmatrix} \tag{20}$$

where k_t denotes the coefficient of thrust, k_b denotes drag's coefficient, while L is distance between the quadrotor motor and the mass center.

3.1.4 Translational motion

To model, the linear dynamic of a quadrotor, Newtonian mechanics are employed, the extraneous forces against the system include gravitational force, thrust forces from the motors, and aerodynamic drag forces [3], [29] equation (21).

$$\dot{m}v_1 = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + RT_B + F_D \tag{21}$$

The drag force F_D represents the resistance experienced by the quadrotor as it moves through the air. This force is due to air resistance or fluid friction, and acts opposite to the direction of motion. To simplify the modeling process, the drag force is often approximated to be proportional to the linear quadrotor velocity in all the three spatial directions.

$$F_D = -k_d v_I \tag{22}$$

The i^{th} motor angular velocity generates a force F_i, the axis of the motor aligned in the z-direction. When these individual forces are combined, they produce a total thrust T acting in the z-axis of the body frame given in equation (23).

$$T = \sum_{i=1}^{4} F_i = k_t \sum_{i=1}^{4} \omega_i^2 \tag{23}$$

Since the thrust acts in the direction of the z-axis:

$$T_B = \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} = k_t \begin{bmatrix} 0 \\ 0 \\ \Sigma_{i=1}^{4} \omega_i^2 \end{bmatrix} \tag{24}$$

Based on the earlier discussed rotational dynamics, Equation (11) can be rephrased as equation (25) in state-space.

$$\dot{\omega}_B = I^{-1}(-\omega_B \times (I\omega_B) - \Gamma + \tau_B) \tag{25}$$

$$\dot{\omega}_B = I^{-1}\left(\begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} I_{xx}p \\ I_{yy}q \\ I_{zz}r \end{bmatrix} - J_r \begin{bmatrix} p \\ q \\ r \end{bmatrix} \omega_r + \tau_B \right) \tag{26}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{(I_{yy}-I_{zz})qr}{I_{xx}} \\ \frac{(I_{zz}-I_{xx})pr}{I_{yy}} \\ \frac{(I_{xx}-I_{yy})pq}{I_{zz}} \end{bmatrix} - J_r \begin{bmatrix} \frac{q}{I_{xx}} \\ \frac{-p}{I_{yy}} \\ 0 \end{bmatrix} \omega_r + \begin{bmatrix} \frac{\tau_\phi}{I_{xx}} \\ \frac{\tau_\theta}{I_{yy}} \\ \frac{\tau_\psi}{I_{zz}} \end{bmatrix} \tag{27}$$

From the earlier discussed linear dynamics, equation 21 can now be expressed as equation (28).

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} cos(\psi)sin(\theta)cos(\phi) + sin(\psi)sin(\phi) \\ sin(\psi)sin(\theta)cos(\phi) + cos(\psi)sin(\phi) \\ cos(\theta)cos(\phi) \end{bmatrix} - \frac{1}{m} \begin{bmatrix} k_{dx} & 0 & 0 \\ 0 & k_{dy} & 0 \\ 0 & 0 & k_{dz} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{28}$$

3.2 Quadrotor System Stabilization

A quadrotor is an underactuated system because it has only four motor speeds (control inputs) but six outputs of interest: position [x, y, z] and orientation [ϕ, θ, ψ]. To address this challenge, the control problem is typically divided into two separate control loops, which are the attitude control and position control loops: Manages the linear motion states. The overall control architecture for the quadrotor is given in Figure 2, while the simulation Simulink is as shown in Figure 3.

3.2.1 Internal model control-based proportional integral derivative control

The Internal Model Control (IMC) framework is a strategy that is model-based, which uses a process model to improve controller performance. The general structure of an IMC-based PID controller contains process model, controller, feedback filter, desired state and the current state. The process model $G_p(s)$ represents the dynamic relationship between the control inputs (motor angular velocities) and the outputs (attitude and position), this was to predict the system's behavior [23], [24].

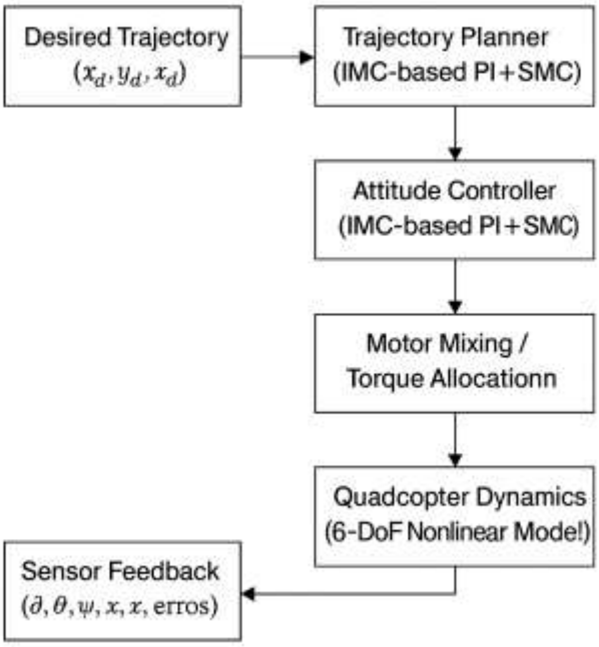


Figure 3: Control Architecture

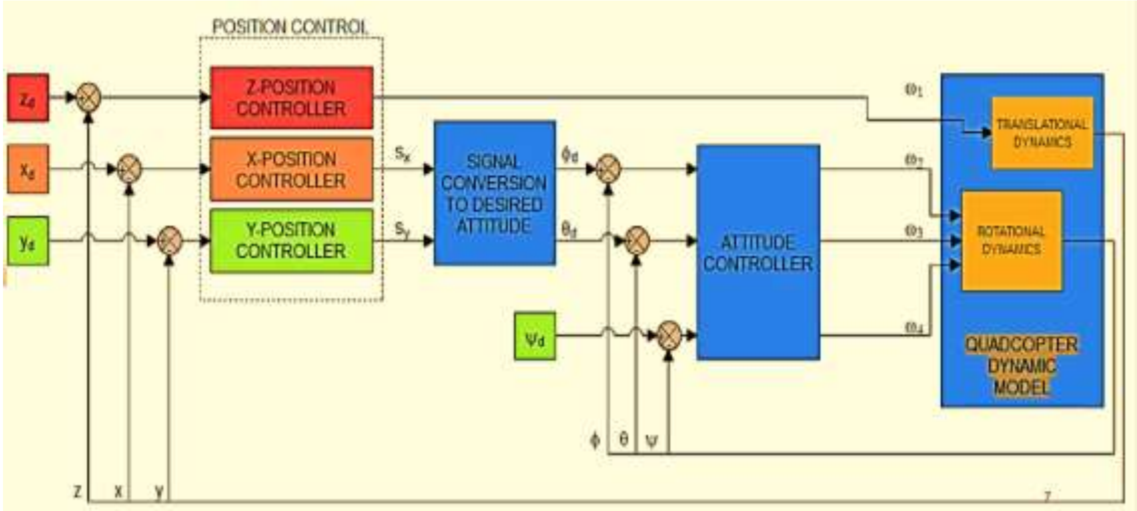


Figure 4: Simulation Simulink

The controller, Q(s) in equation (29) is designed based on the inverse of the process model in equation (30), while the low-pass filter F is given as equation (31). By comparing equation (30) with the standard PID controller, it shows that PI controller was enough to give the best response, and $T_i = T_p$.

$$Q(s) = G_p^{-1}(s) \times F(s) \tag{29}$$

$$G_p^{-1} = \frac{K_p}{[T_p s]+1}(s) \times F(s) \tag{30}$$

$$F(s) = \frac{1}{(\lambda s+1)^n} \tag{31}$$

The error $e(t)$ is computed as the difference between the desired state $r(t)$ and the predicted output of the process model $y(t)$, this is given as Equation (32). The control input $u(t)$ in Equation (33) is also computed using the IMC controller $Q(s)$.

$$e(t) = r(t) - y(t) \tag{32}$$

$$u(t) = Q(s) \times e(t) \tag{33}$$

where $F(s)$ is a low-pass filter used to ensure robustness and proper controller behavior, λ is the filter time constant and n is the order of the filter, r(t) depicts the desired state, u(t)is the control input, and y(t) is the current or actual state. T_p and K_p are the proportional time constant and the proportional gain of the resulting PID controller through the IMC procedures.

For the control of the quadrotor, the proportional and the integral terms were used. The generated torque is then proportional to the velocities. The torques are therefore set to be equal to the controller's output given in Equation (34) to (35).

$$\tau = I \times u(t) \tag{34}$$

$$\begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} I_{xx}\left(k_{P\phi}(\phi_{des} - \phi) + k_{I\phi}(\dot{\phi}_{des} - \dot{\phi})\right) \\ I_{yy}\left(k_{P\theta}(\theta_{des} - \theta) + k_{I\theta}(\dot{\theta}_{des} - \dot{\theta})\right) \\ I_{zz}\left(k_{P\psi}(\psi_{des} - \psi) + k_{I\psi}(\dot{\psi}_{des} - \dot{\psi})\right) \end{bmatrix} \tag{35}$$

The angular velocity of its motors is the control input to the quadrotor system. Equation (20) establishes the relationship between torque and the square of the motor angular velocities, with three equations with four unknowns. To simplify the system, the total thrust, that influences the acceleration along the direction z axis, is set to be $T = mg$. This constraint ensures the quadrotor remains airborne by transforming the thrust equation into the proper reference frame and applying a IMC-based PI controller to minimize errors in the z-axis, the system can achieve stable and controlled flight:

$$T = \left(g + k_{Pz}(z_{des} - z) + k_{Iz}(\dot{z}_{des} - \dot{z})\right)\frac{m}{cos(\phi)cos(\theta)} \tag{36}$$

We find the angular motor velocities $\omega_{m1}^2, \omega_{m2}^2, \omega_{m3}^2, and\ \omega_{m4}^2$ in equation (37) by determine first Equation (35) and subject it to Equation (20). By simplifying Equation (37) gives Equations (38), (39), (40), and (41)

$$\begin{bmatrix}\omega_{m1}^2\\ \omega_{m2}^2\\ \omega_{m3}^2\\ \omega_{m4}^2\end{bmatrix}=\begin{bmatrix}k_t & k_t & k_t & k_t\\ 0 & -lk_t & 0 & lk_t\\ -lk_t & 0 & lk_t & 0\\ -k_b & k_b & -k_b & k_b\end{bmatrix}^{-1}\begin{bmatrix}T\\ \tau_\phi\\ \tau_\theta\\ \tau_\psi\end{bmatrix} \tag{37}$$

$$\omega_{m1}^2=\frac{T}{4k_t}-\frac{\tau_\theta}{2lk_t}-\frac{\tau_\psi}{4k_b} \tag{38}$$

$$\omega_{m2}^2=\frac{T}{4k_t}-\frac{\tau_\phi}{2lk_t}+\frac{\tau_\psi}{4k_b} \tag{39}$$

$$\omega_{m3}^2=\frac{T}{4k_t}+\frac{\tau_\theta}{2lk_t}-\frac{\tau_\psi}{4k_b} \tag{40}$$

$$\omega_{m4}^2=\frac{T}{4k_t}+\frac{\tau_\phi}{2lk_t}+\frac{\tau_\psi}{4k_b} \tag{41}$$

3.2.2 Sliding mode control (SMC) with extended state observer (ESO)

Sliding Mode Control (SMC) is a robust control strategy that drives the system states to a preset sliding surface and maintains them on this surface despite uncertainties and disturbances [25]. The Extended State Observer (ESO) is used to estimate and improve for the unmodeled dynamics, external disturbances, and uncertainties [26]. The sliding surface (SS) is defined as Equation (42).

$$S=\dot{e}+\lambda e \tag{42}$$

where $e=x_{des}-x$ is the error of tracking, x_{des} is the desired state x is the actual state, and λ is the positive constant which determines the rate of convergence. The sliding surfaces for the x, y, and z axes of the quadrotor are given in Equations (43), (44), and (45).

$$S_x=\dot{e}_x+\lambda_x e_x \tag{43}$$

$$S_y=\dot{e}_y+\lambda_y e_y \tag{44}$$

$$S_z=\dot{e}_z+\lambda_z e_z \tag{45}$$

where $e_x=x_{des}-x$, $e_y=y_{des}-y$, and $e_z=z_{des}-z$.

The control law for SMC consists of two parts: the equivalent control which is the part that compensates for the known dynamics of the system, and switching control which is also the part that ensures robustness against uncertainties and disturbances. The control input u is given by Equation (46).

$$u=u_{eq}+u_{sw} \tag{46}$$

where u_{eq} is the equivalent control, $u_{sw}=-K\times sin(S)$, K is a positive gain, and $sin(S)$ is the signum function.
The control inputs u_x, u_y, and $u_z$for the positions in x, y, and z axes of the quadrotor, are given as Equations (47), (48), (49).

$$u_x=u_{eq,x}-K_x\times sin(S_x) \tag{47}$$

$$u_y=u_{eq,y}-K_y\times \sin(S_y) \tag{48}$$

$$u_z=u_{eq,z}-K_z\times sin(S_z) \tag{49}$$

The ESO is used to estimate the total disturbance d (including unmodeled dynamics and external disturbances) and compensate for it in the control law. The ESO dynamics are given in Equations (50), and (51).

$$\dot{x}=A\hat{x}+Bu+L(x-\hat{x}) \tag{50}$$

$$\hat{d}=C\hat{x} \tag{51}$$

where $\hat{x}$ is the estimated state, $\hat{d}$ is the estimated disturbance, A, B and C are system matrices, and L is the observer gain matrix.
The ESO estimates the disturbances in the positions x, y, and z axes of the quadrotor are given as Equations (52), (53), and (54).

$$\hat{d}_x=C_x\hat{x}_x \tag{52}$$

$$\hat{d}_y=C_y\hat{x}_y \tag{53}$$

$$\hat{d}_z=C_z\hat{x}_z \tag{54}$$

The torques τ_ϕ, τ_θ, and τ_ψ are computed using the SMC control law and the estimated disturbances as Equations (55) to (57).

$$\tau_\phi = I_{xx}(u_{eq,\phi} - K_\phi \times \sin(S_\phi) - \hat{d}_\phi \tag{55}$$

$$\tau_\theta = I_{yy}(u_{eq,\theta} - K_\theta \times \sin(S_\theta) - \hat{d}_\theta \tag{56}$$

$$\tau_\psi = I_{zz}(u_{eq,\psi} - K_\psi \times \sin(S_\psi) - \hat{d}_\psi \tag{57}$$

The total thrust T is computed to maintain the quadrotor's altitude as equation (58).

$$T = \left(g + k_{eqz} - k_{Iz} \times \sin(S_z) - \hat{d}_z\right)\frac{m}{cos(\phi)cos(\theta)} \tag{58}$$

The angular velocities of the motors ω_{m1}^2, ω_{m2}^2, ω_{m3}^2, and ω_{m4}^2 are computed also for the SMC with ESO by solving the following system in Equation (59).

$$\begin{bmatrix} \omega_{m1}^2 \\ \omega_{m2}^2 \\ \omega_{m3}^2 \\ \omega_{m4}^2 \end{bmatrix} = \begin{bmatrix} k_t & k_t & k_t & k_t \\ 0 & -lk_t & 0 & lk_t \\ -lk_t & 0 & lk_t & 0 \\ -k_b & k_b & -k_b & k_b \end{bmatrix}^{-1} \begin{bmatrix} T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \tag{59}$$

By simplifying equation (59), gives the individual motor angular velocities as Equations (60 to (62).

$$\omega_{m1}^2 = \frac{T}{4k_t} - \frac{\tau_\theta}{2lk_t} - \frac{\tau_\psi}{4k_b} \tag{60}$$

$$\omega_{m2}^2 = \frac{T}{4k_t} - \frac{\tau_\phi}{2lk_t} + \frac{\tau_\psi}{4k_b} \tag{61}$$

$$\omega_{m3}^2 = \frac{T}{4k_t} + \frac{\tau_\theta}{2lk_t} - \frac{\tau_\psi}{4k_b} \tag{62}$$

$$\omega_{m4}^2 = \frac{T}{4k_t} + \frac{\tau_\phi}{2lk_t} + \frac{\tau_\psi}{4k_b} \tag{63}$$

3.3 Trajectory Tracking

The primary objective of trajectory tracking is to guide the quadrotor from its present position to a desired one by adjusting the angular velocities of its motors [27]. Achieving this however is highly challenging due to the dynamic complexity of the quadrotor itself, the challenge still remains finding the correct path.

3.3.1 IMC-based PI

To relate the uncontrollable position x and y to the controllable angles of ϕ and θ, two assumptions were made:

i. $\sin(x) \approx x$ and $\cos(x) \approx 1$
ii. The yaw angle $\psi = 0$

These assumptions simplify Equation (28) as Equations (64) to (66).

$$\ddot{x}_{eq} = \frac{T}{m}\theta_{des} - \frac{k_{dx}}{m}\dot{x}_{eq} \tag{64}$$

$$\ddot{y}_{eq} = \frac{T}{m}\phi_{des} - \frac{k_{dy}}{m}\dot{y}_{eq} \tag{65}$$

$$\ddot{z}_{eq} = -g + \frac{T}{m} - \frac{k_{dz}}{m}\dot{z}_{eq} \tag{66}$$

Equations (64), (65) and (66) can be rewritten as:

$$\theta_{eq} = \frac{1}{T}\left[m\ddot{x}_{eq} + k_{ix}\dot{x}_{eq} + k_{Pxy}\left(x_{eq} - x\right)\right] \tag{67}$$

$$\phi_{eq} = -\frac{1}{T}\left[m\ddot{y}_{eq} + k_{iy}\dot{y}_{eq} + k_{Pxy}\left(y_{eq} - y\right)\right] \tag{68}$$

$$T_{eq} = m\left(\ddot{z}_{eq} + g\right) + k_{dz}\dot{z}_{eq} \tag{69}$$

ϕ_{eq}, θ_{eq} and T_{eq} described the desired values of the roll angle, pitch angle, and thrust, respectively.

3.3.2 SMC-ESO

Two assumptions were made in order to equate the uncontrollable positions x and y to the controllable angles ϕ and θ using SMC-ESO control strategy:

i. There is no approximation of small angle
ii. The desired angle $\psi = 0$

3.4 Simulation Parameters

The quadrotor is simulated using Table 2 parameter, and the initial conditions of the system is in Table 3, while that of Table 3 and Table 4 are the parameters that define the gains and settings for the IMC-based PID and SMC with ESO controllers, respectively.

Table 2: Simulation parameters

Parameter	Value	Unit	Description
G	9.81	m/s^2	Acceleration due to gravity
M	0.558	kg	Total mass of the quadrotor
L	0.225	m	Distance from the center to each
$\mathbf{J_r}$	3.357×10^{-5}	kg m^2	Motors' moment of inertia
$\mathbf{k_t}$	1.980×10^{-6}	N/(rad/s)^2	Motor speed to thrust.
$\mathbf{k_b}$	3.140×10^{-7}	N.m/(rad/s)^2	Motor speed to torque
$\mathbf{I_{xx}}$	5.856×10^{-3}	Kg.m^2	Moments of inertia about the x, y, and z axes.
$\mathbf{I_{yy}}$	5.856×10^{-3}		
$\mathbf{I_{zz}}$	8.801×10^{-3}		
$\mathbf{k_{dx}}$	0.25	Kg.m^2	Drag coefficients in the x, y, and z directions.
$\mathbf{k_{dy}}$	0.25		
$\mathbf{k_{dz}}$	0.25		

Table 3: Initial simulation conditions

State	Value	State	Value
X	-1	ϕ	6
Y	1.5	θ	-5
Z	0.5	ψ	2

Table 4: IMC-based PI controller gain

Parameter	Value	Parameter	Value
$\boldsymbol{k_P}$	10.5	k_I	0.001
$\boldsymbol{k_{P\phi}}$	12.5	$k_{I\phi}$	13.0
$\boldsymbol{k_{P\theta}}$	10.0	$k_{I\theta}$	11.6
$\boldsymbol{k_{P\psi}}$	10.0	$k_{I\psi}$	11.0
$\boldsymbol{k_{Px}}$	0.04	k_{Py}	9.5
$\boldsymbol{k_{Pz}}$	0.02	k_{Iz}	0.05

Table 5 SMC with ESO gains

Parameter	Value	Description
$\boldsymbol{Smc\_lambda}$	20.0	Sliding surface gain
$\boldsymbol{smc\_eta}$	1.2	Switching control gain
$\boldsymbol{eso\_alpha}$	1.8	ESO Disturbance Estimation Gain
$\boldsymbol{eso\_beta}$	1.2	ESO Disturbance Rejection Gain

4. SIMULATION, RESULTS AND DISCUSSION

4.1 Simulation and Results

The quadrotor dynamics is simulated using MATLAB 2023a. The simulation is timed to start at 0.2 second to 8 seconds total simulation time. The uncontrolled Euler angles of ϕ, θ *and* ψ were plotted in Figure 4 and the quadrotor positions x, y, z are plotted in figure 5. Also, the angular velocity of each motor was plotted in Figure 6.

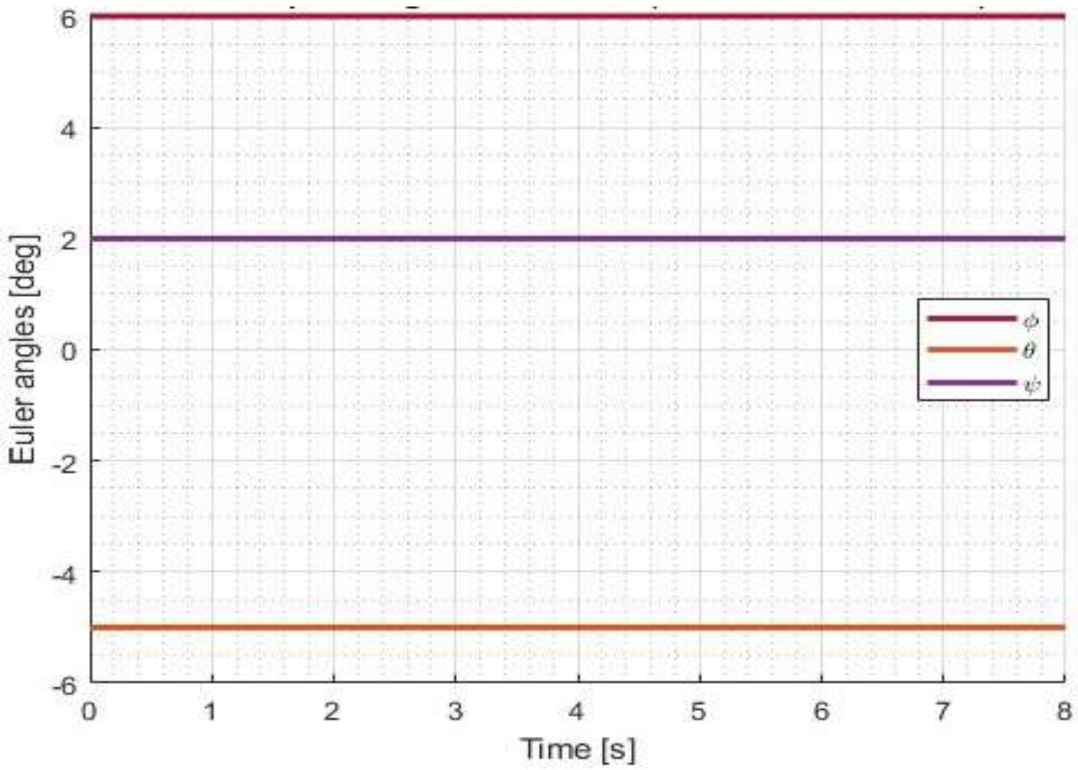


Figure 4: Uncontrolled quadrotor euler angles ϕ, θ *and* ψ per time

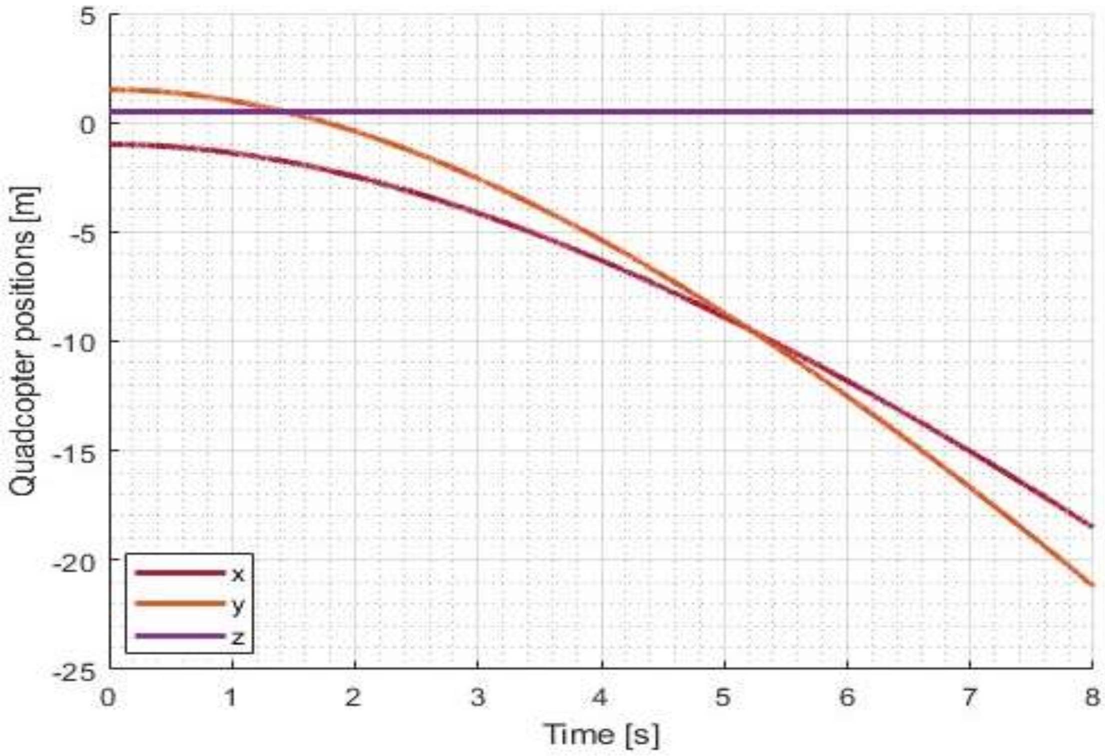


Figure 5: Uncontrolled quadrotor positions x, y, z per time

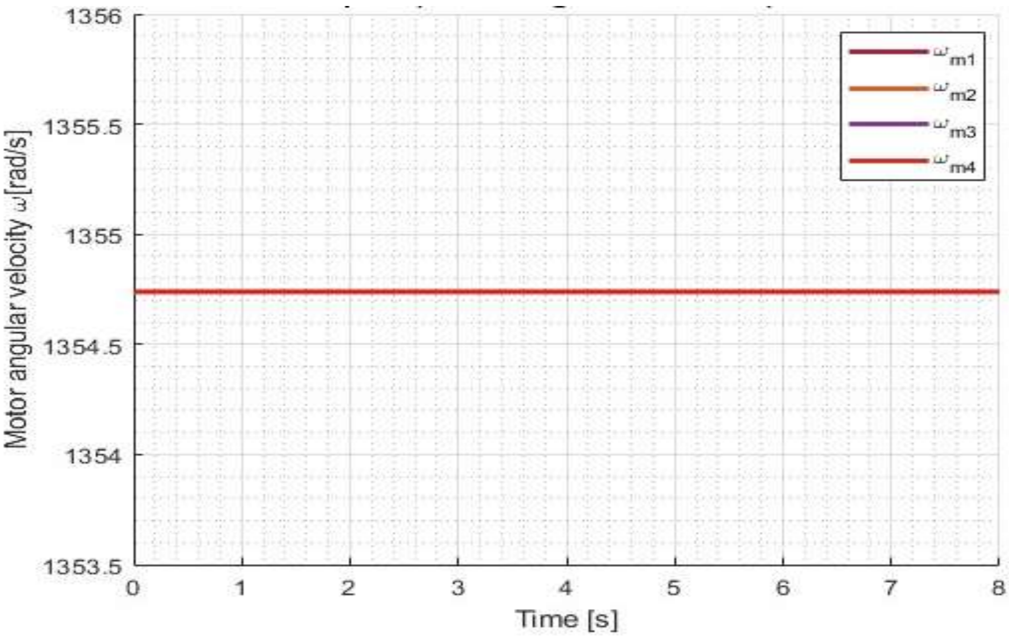


Figure 6: Quadrotor control input per time.

From Figure 4 to Figure 6, its evident that the control input is kept constant with orientation angles unchanged. However, the positions x, y, and z tend to be unstable. Therefore, by continuously adjusting the control inputs is important to stabilize both quadrotor's orientation and position.

4.1.1 Attitude stabilization

An IMC-based PI controller is applied to the quadrotor system to achieve stabilization in driving the quadrotor states to target (zero). The system's initial conditions are specified in Table 2 remain unchanged. The control gains for the IMC-based PI controller are manually tuned, and their values are provided in Table 3. The simulation runs with a time interval step of 0.01 seconds over a total duration of 8 seconds. The results are illustrated Figure 7 which shows the Euler angles ϕ, θ, and ψ; Figure 8 which also shows the angular velocities of the four motors, while Figure 9 shows the quadrotor's positional states x, y, and z. From the results, it's noted that the Euler angles stabilize to zero within 3 seconds, while the z-position reaches zero after 2.4 seconds. However, the positions x and y fail to stabilize at zero due to the lack of observability in these states.

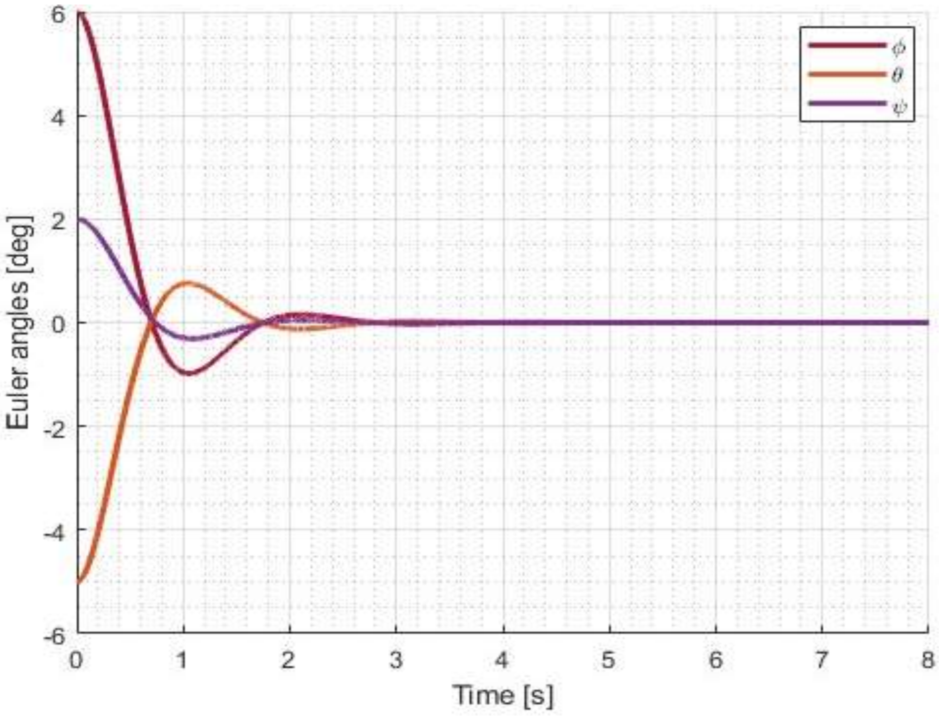


Figure 7: Euler angles per time

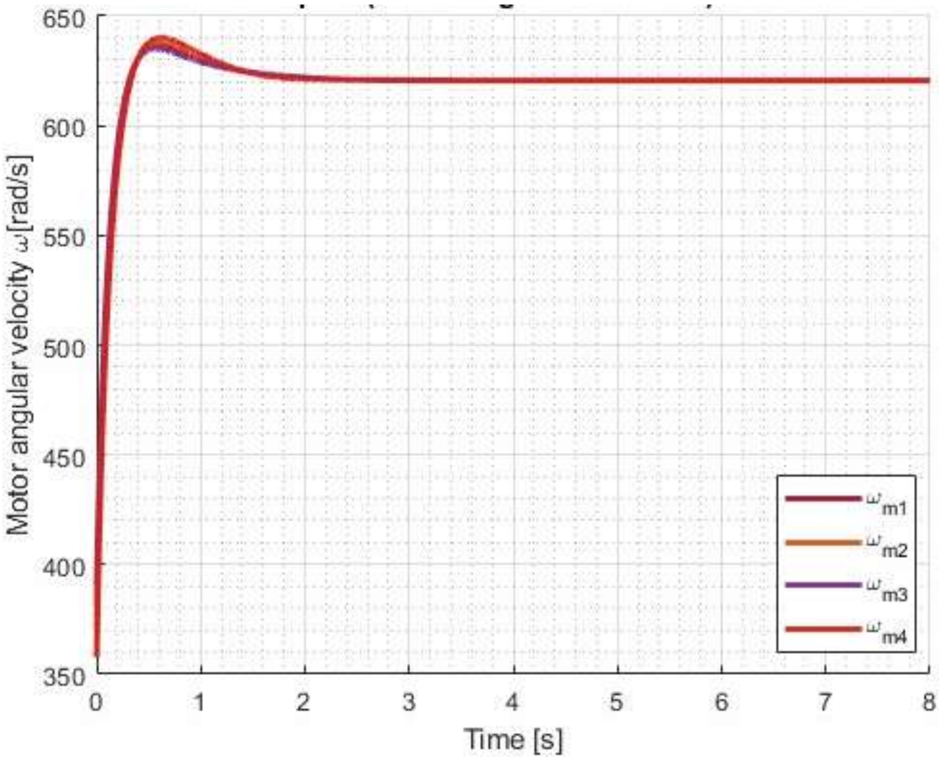


Figure 8: Motor angular velocity per time

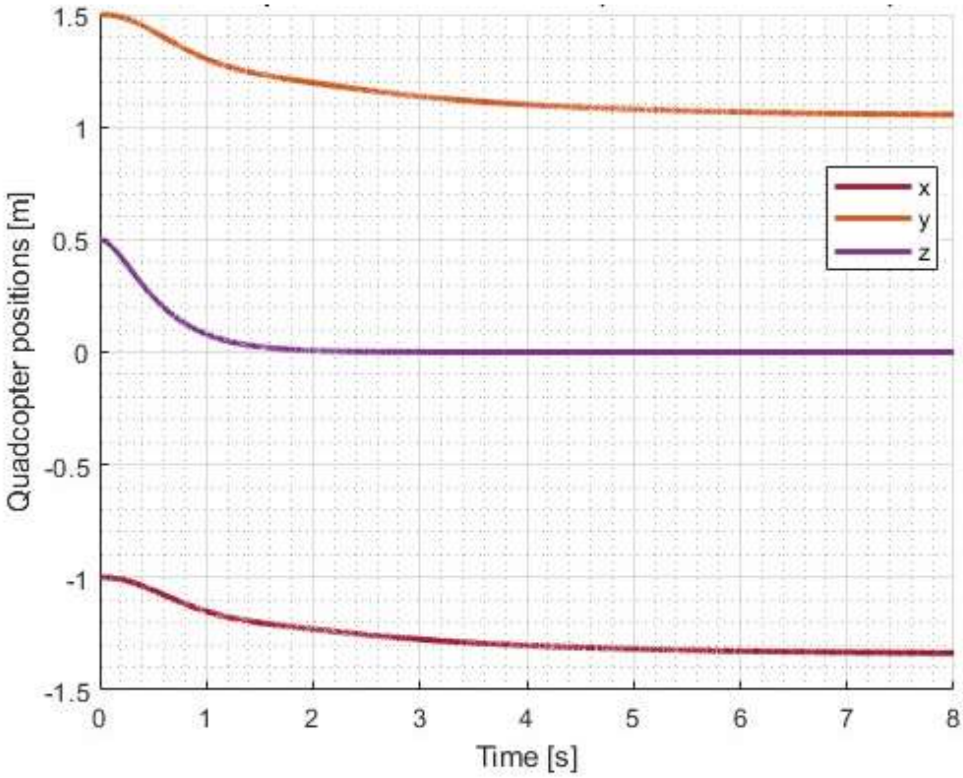


Figure 9: Quadrotor positions per time

To address the issue of stability of the x and y, the trajectory tracking concept was employed, to adjust the two positions to move to zero. This involved adjusting the desired zero states x and y to the controllable orientation angles ϕ and θ states as outlined in equation (47), (48) and (44). An additional proportional controller with gain $k_{Pxy} = 0.0495$, introduced to move the quadrotor states to the target (zero), and the simulation was conducted with a time step interval of 1 second over a total duration of 35 seconds. The results were plotted in Figure 10 as the quadrotor's motor angular velocities, Figure 11 also plotted the Euler angles ϕ, θ *and* ψ, while Figure 12 shows the modified positions x, y and z.

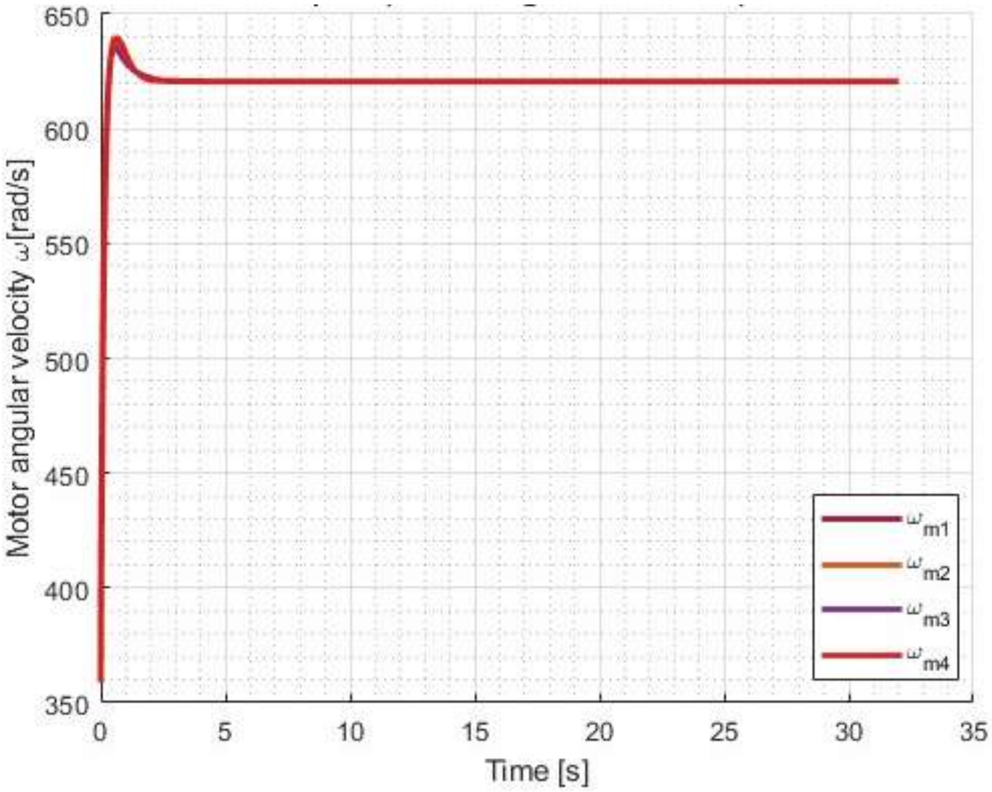


Figure 10. Adjusted motor angular velocity per time

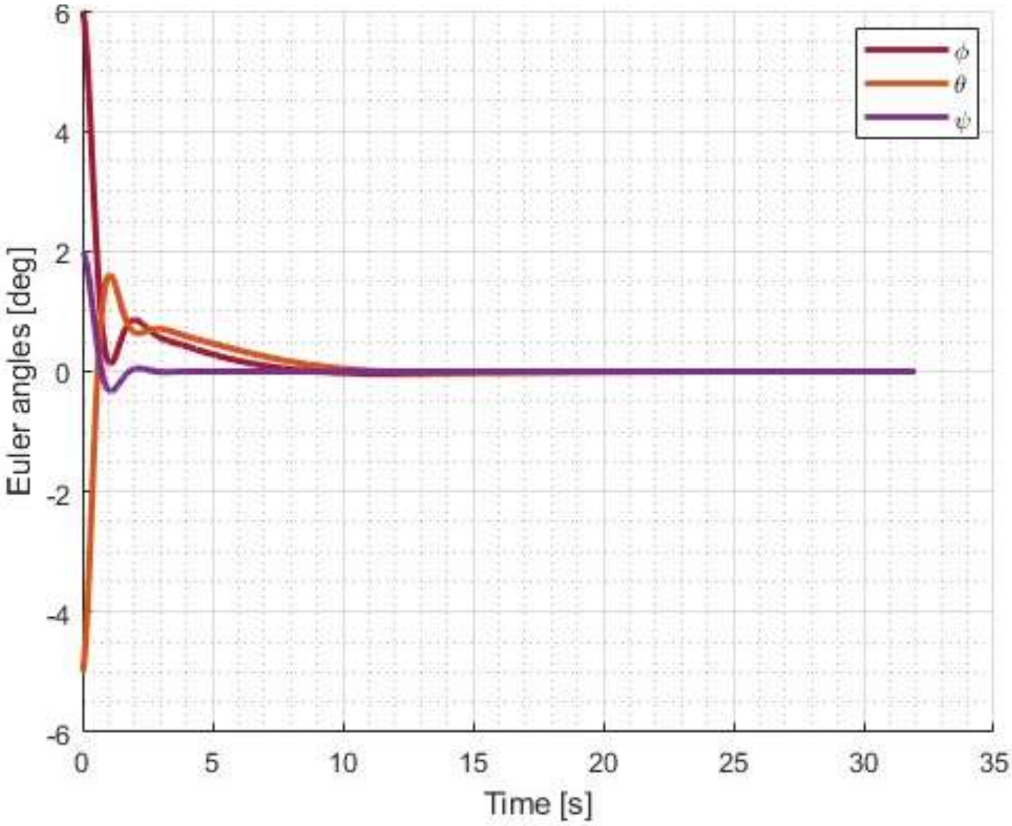


Figure 11: Adjusted Euler angles per time

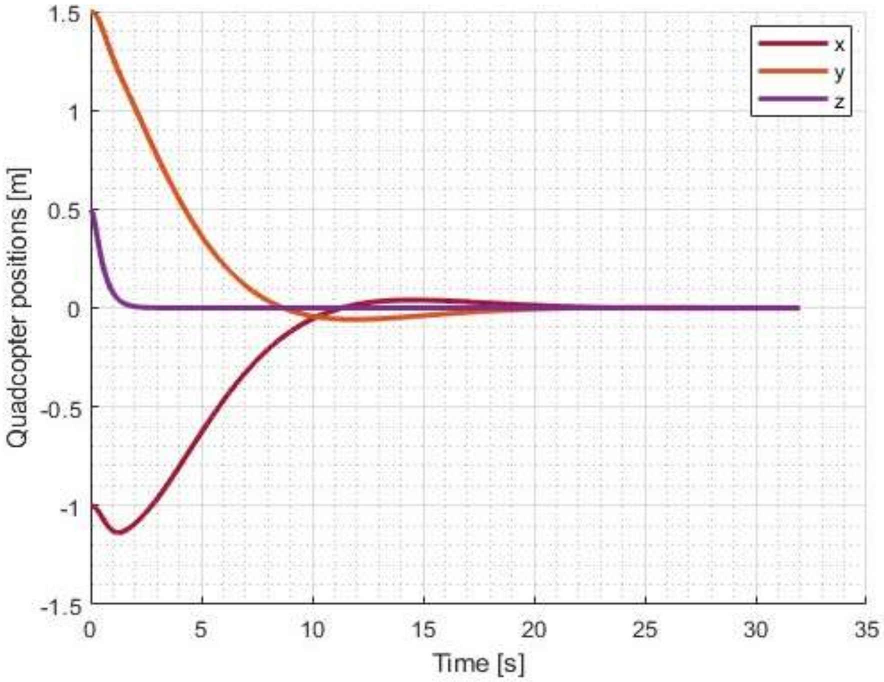


Figure 12: Adjusted quadrotor positions per time

The result of the adjustment shows that Euler angles stabilized at zero immediately after 12 seconds, while the positional states x, y, and z also reach zero after 23 seconds. The result indicates that the system achieves attitude stabilization earlier than positional stabilization, reflecting the inherent challenges with quadrotor dynamics and control.

4.1.2 Quadrotor trajectory tracking

Two distinct control strategies are employed and simulated for the helical and figure-eight shape tracks. The first strategy employed the IMC-based PI controller, used for the tracking, while the second strategy combined the IMC-based PI controller with SMC and ESO. In both of these cases, the simulation is conducted over a duration of 60 seconds to analyze and compare their performance.

A. Trajectory tracking with IMC-based PI controller

The desired and the corresponding actual trajectories comparison for the Helical trajectory is given in Figure 13. Figure 14 and Figure 15 describe the coordinate variables (positions x and y), respectively per time.

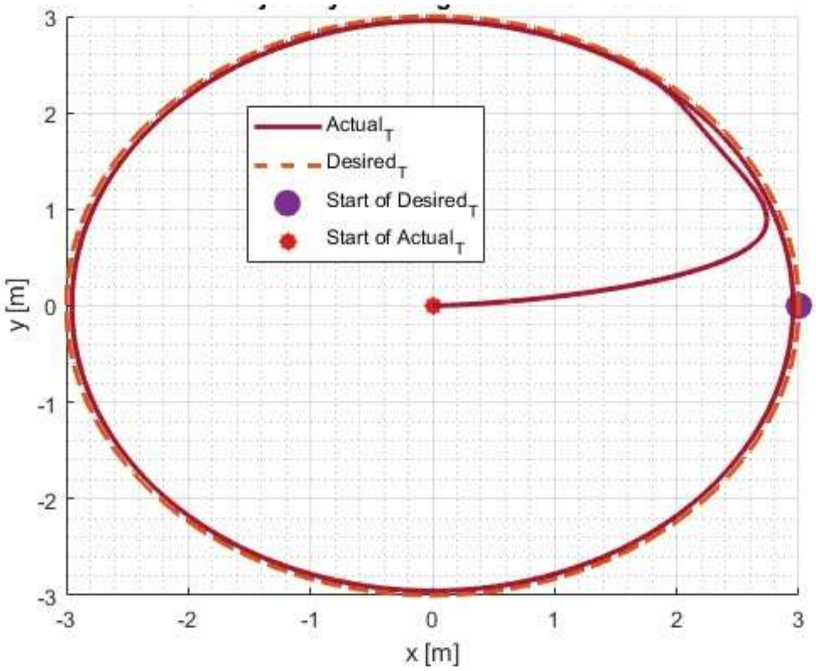


Figure 13: Desired and actual helical trajectory

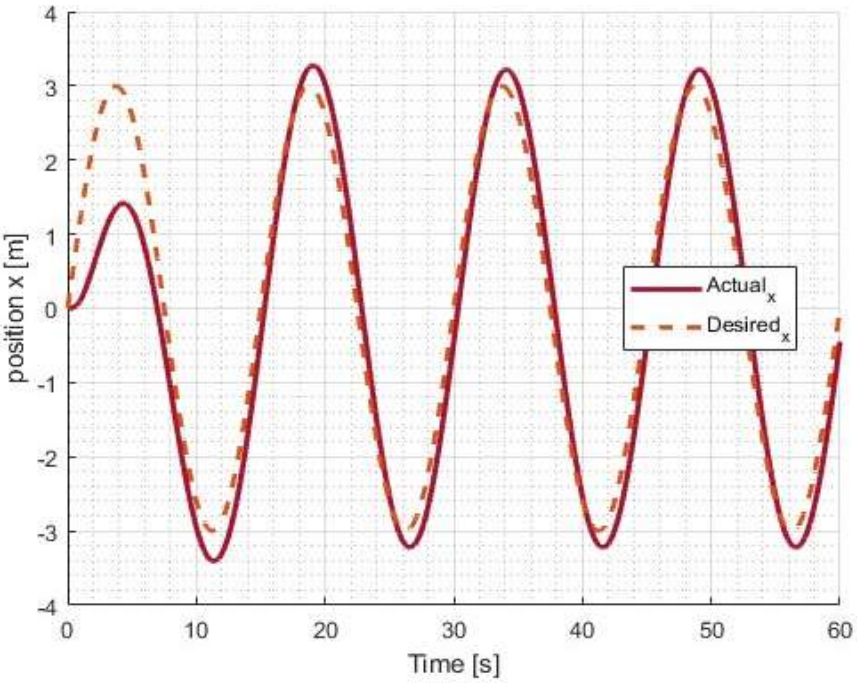


Figure 14: Position x desired and actual helical trajectory

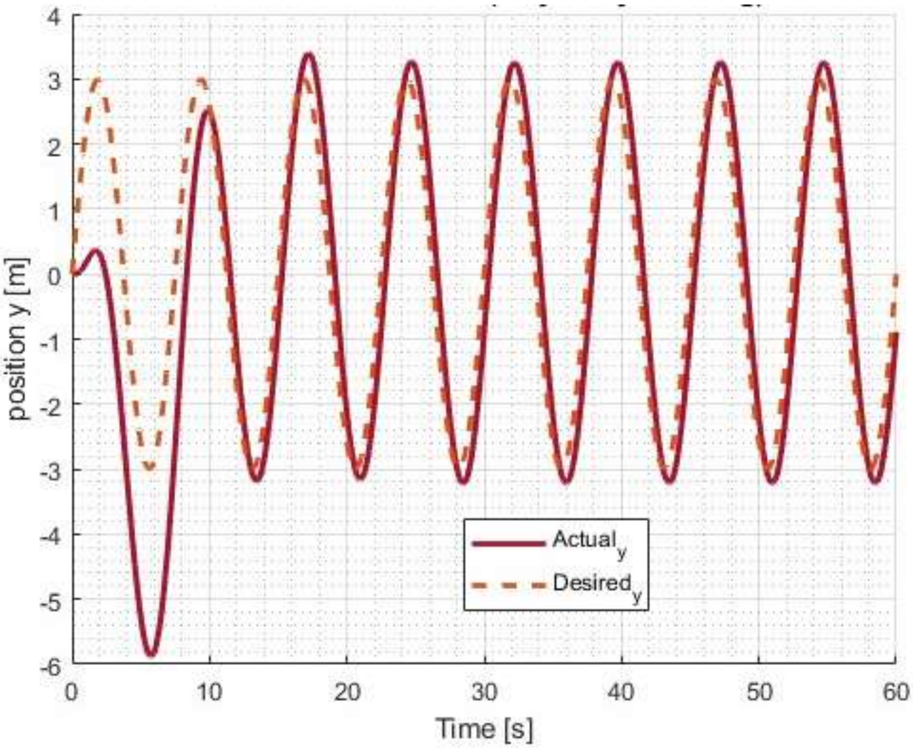


Figure 15: Position y desired and actual helical trajectory

Another trajectory in the shape of a figure-eight is simulated, and the results are shown in Figure 16. The graphs depicting the position variables x and y over a period of 60 seconds for the figure-eight trajectory are presented in Figure 17 and Figure 18, respectively. These plots shows the quadrotor's ability to follow complex paths and the performance of the strategies in tracking such trajectories.

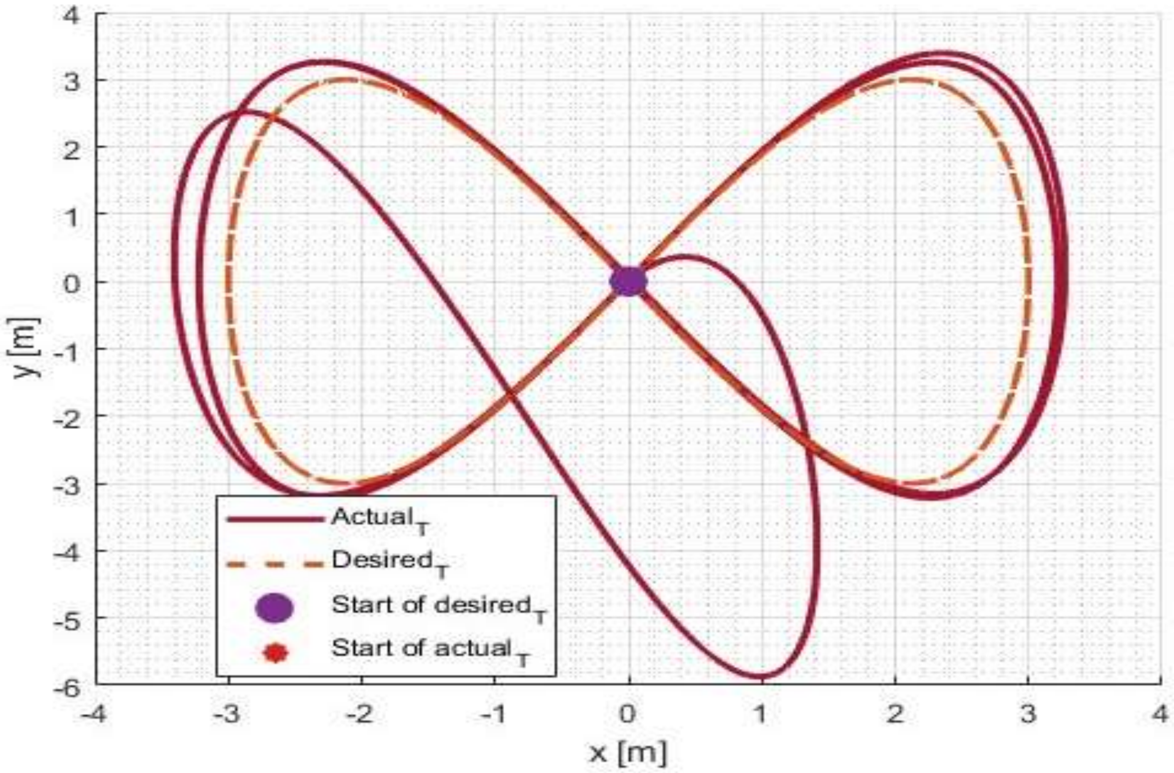


Figure 16: Desired and actual figure-eight trajectory

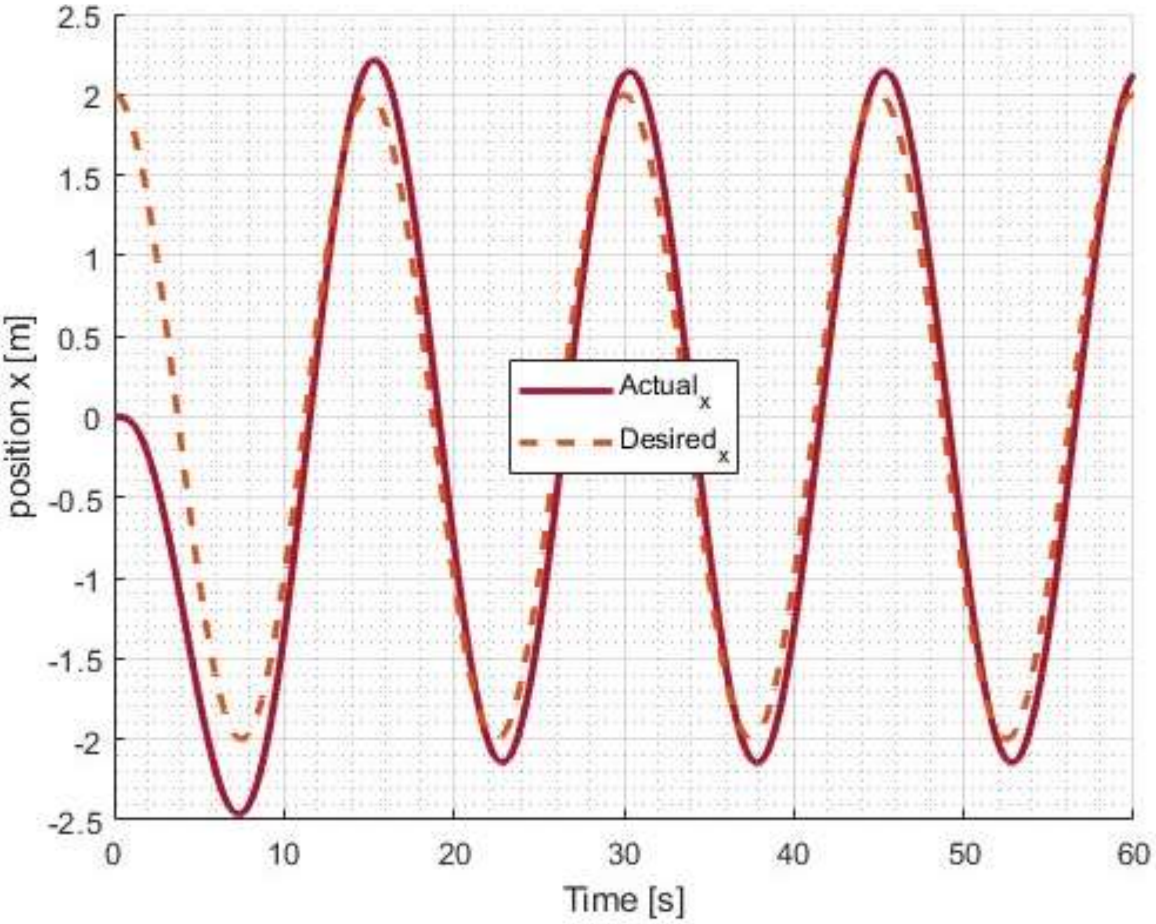


Figure 17: Position x desired and actual figure-eight trajectory

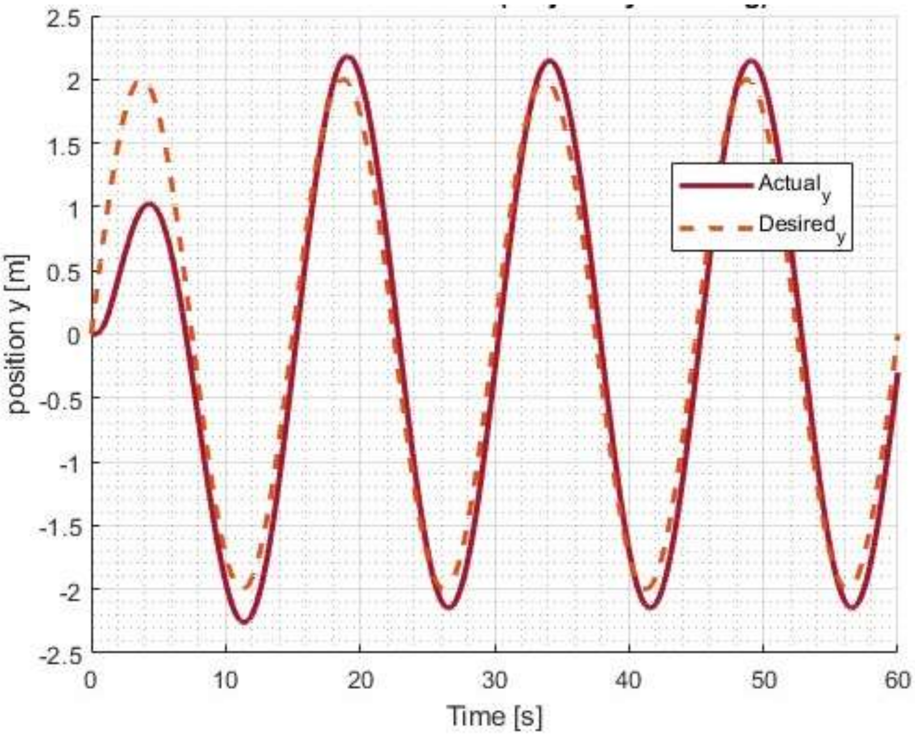


Figure 18: Position y desired and actual figure-eight trajectory

Just like Figure 17 and Figure 18, a similar result is also seen in Figure 14 and Figure 15 where the quadrotor is unable to track the desired position trajectory with high precision.

B Trajectory tracking with SMC-ESO and PI controller

The SMC-ESO strategy were simulated with the model of the quadrotor alongside an IMC-based PI controller. The results for the helical path are presented in Figure 19, which compares the desired trajectory with the actual trajectory by the quadrotor. This demonstrates the effectiveness of the hybrid control strategy in tracking complex paths.

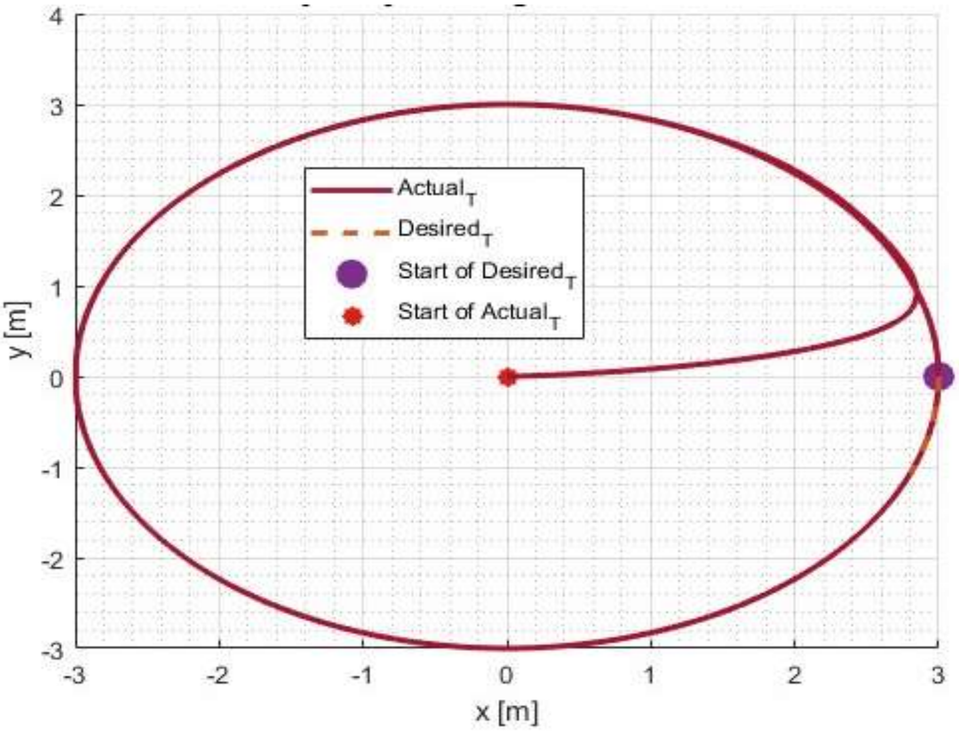


Figure 19: Desired and actual helical trajectory

Another trajectory path for the figure-eight shape is also simulated and the result is given in Figure 20.

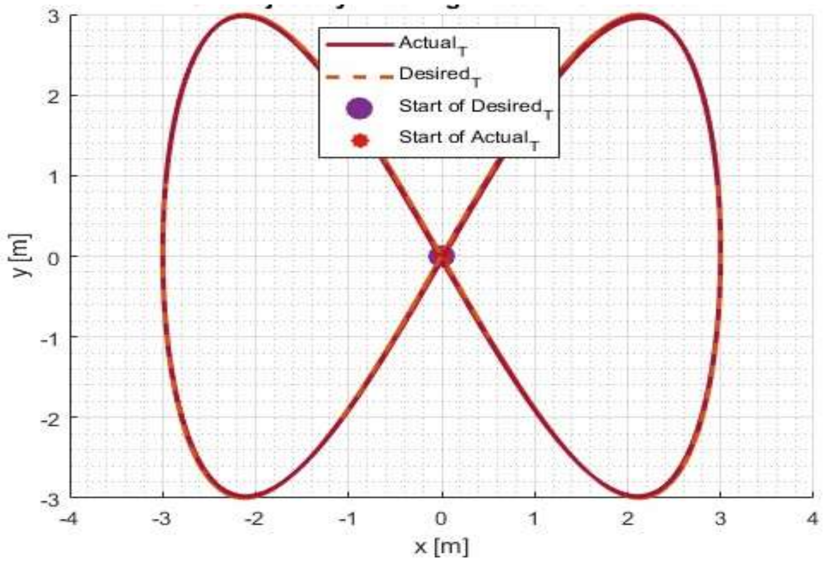


Figure 20: Desired and actual figure-eight trajectory

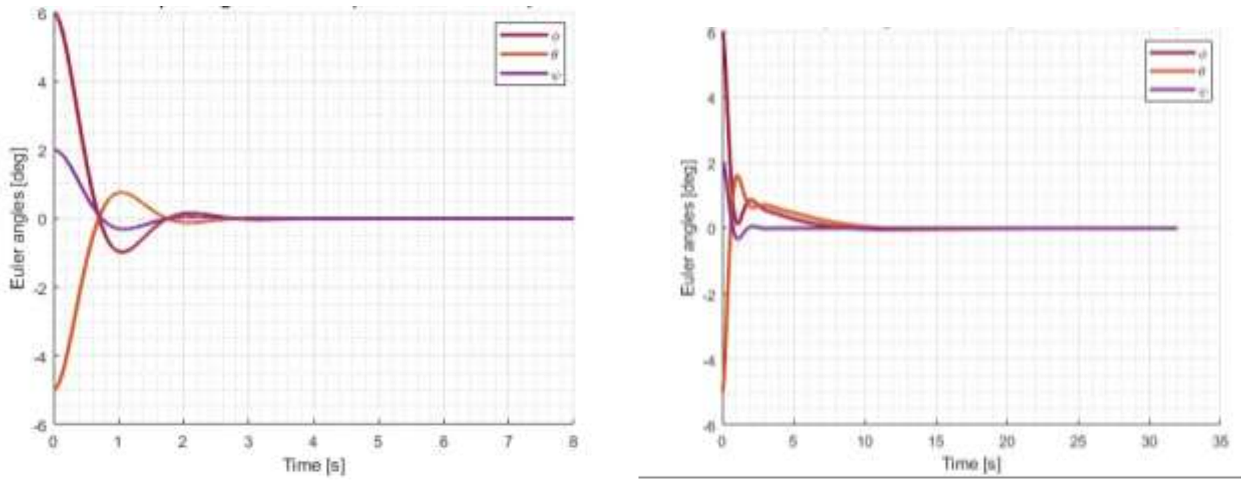
In comparison to the result as seen in Figure 13 and Figure 16 using IMC-based PI, the quadrotor in the SMC-ESO with IMC-based PI is able to locked accurately on to the desired path.

4.2 Result Discussions

A thorough analysis of the IMC-based PI controller as well as its integration with the SMC and ESO's performance for stabilization and trajectory tracking is discussed here to identify the most effective control strategy.

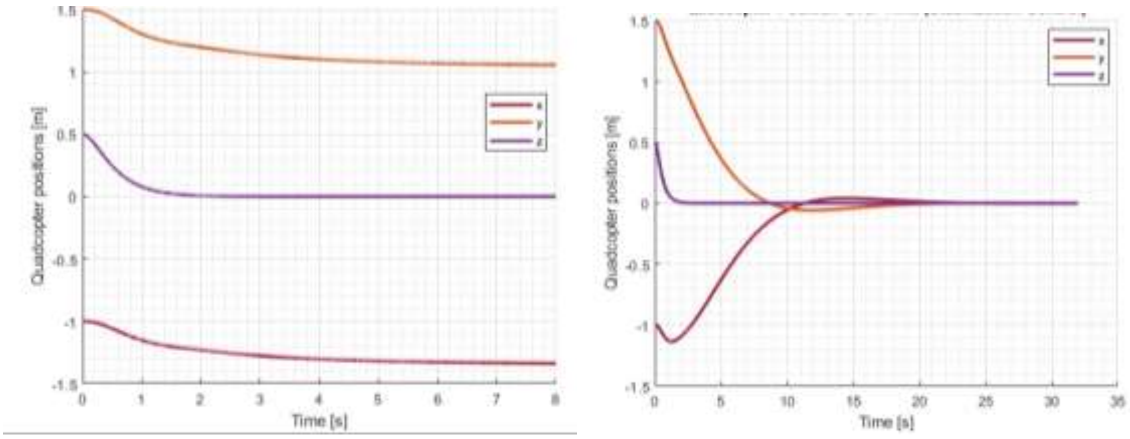
4.2.1 Quadrotor attitude stability

As discussed previously, the IMC-based PI controller with their control parameters computed in Table 3 successfully stabilized the angles ϕ, θ, and ψ hits the desired state within 3 seconds, also the z-position also hit target in 2.4 seconds. However, both positions of x and y did not stabilize to the desired state using this control strategy. To address this, the desired x and y states were mapped to the controllable orientation angles ϕ and θ, and also proportional controller with gain k_{Pxy} was introduced as additional control scheme to move to the desired state. While this approach achieved stabilization, it took significantly more time to stabilize after 12 seconds, and the x, y, z positions also stabilizes after 23 seconds. During this process, the ϕ, θ, x, and the y states exhibited some overshoots before eventually settling to zero. A comparison of the original and adjusted IMC-based PI controlled angle and position variables are shown in Figure 21 and 22, respectively.



(a) **(b)**

Figure 21: (a) Angles $\boldsymbol{\phi, \theta}$ ***and*** $\boldsymbol{\psi}$ (b) Adjusted Angles $\boldsymbol{\phi, \theta}$ ***and*** $\boldsymbol{\psi}$

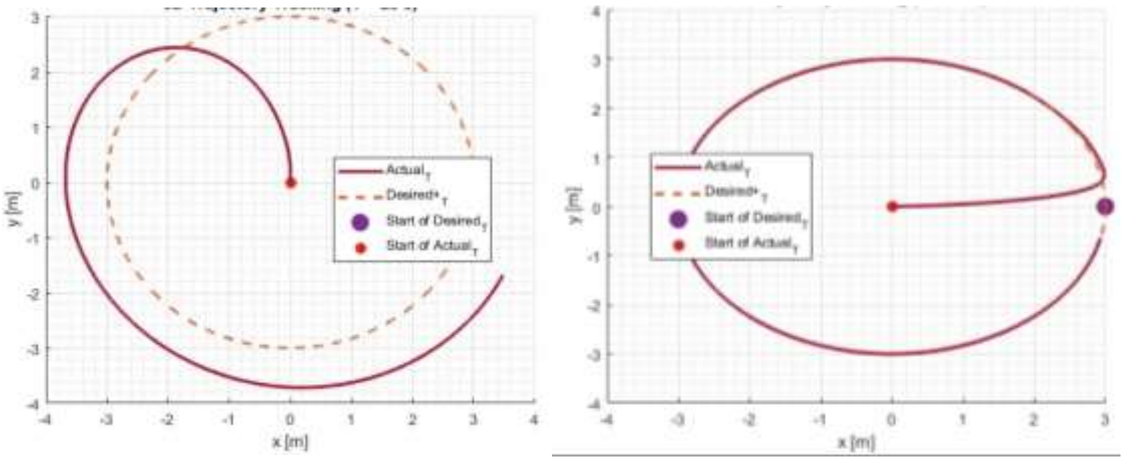


(a) (b)

Figure 22: Unstable positions x, y and z (b) Adjusted positions x, y and z

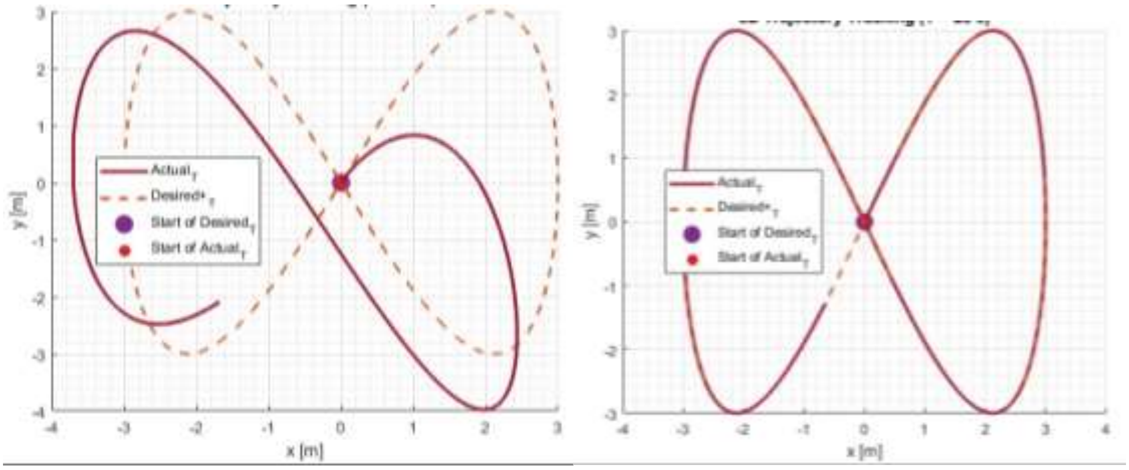
4.2.2 Quadrotor trajectory tracking

The performance with the two tracking strategies were analyzed, and it was observed that the SMC-ESO with IMC-based PI control demonstrated superior results. This approach enabled the quadrotor to accurately lock onto the path and follow the desired trajectory in a shorter time period compared to the other strategy of only IMC-based PI. Figure 23, 24, 25, 26, 27 and 28 show the plots of the desired and actual trajectory for 25, 45 and 60 seconds respectively for both the IMC-based PI and SMC-ESO with IMC-based PI control using helical and figure-eight trajectory.



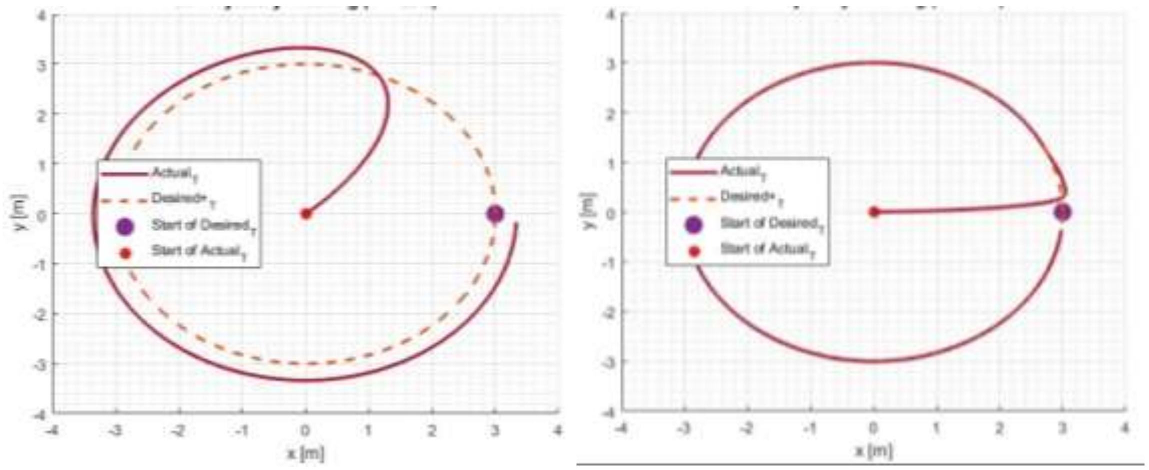
(a) (b))

Figure 23: Helical (a) IMC-based PI (b) SMC-ESO with IMC-based PI control after 25 seconds



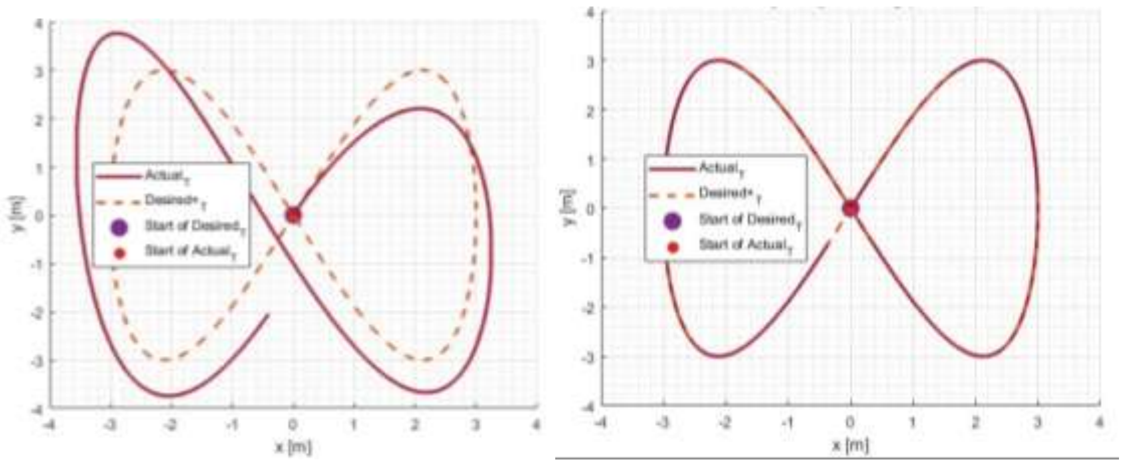
(a) (b)

Figure 24: Figure-eight (a) IMC-based PI (b) SMC-ESO with IMC-based PI control after 25 seconds



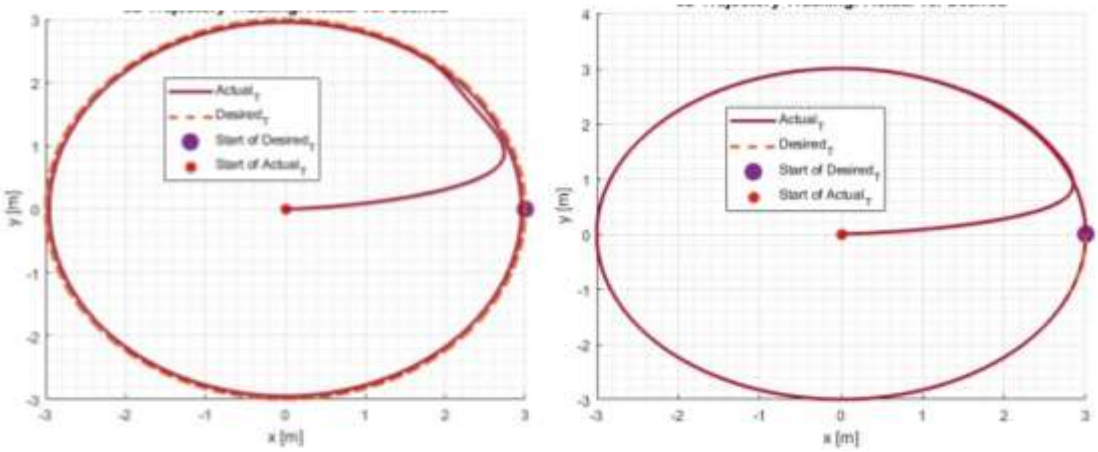
(a) (b)

Figure 25: Helical (a) IMC-based PI (b) SMC-ESO with IMC-based PI control after 45 seconds



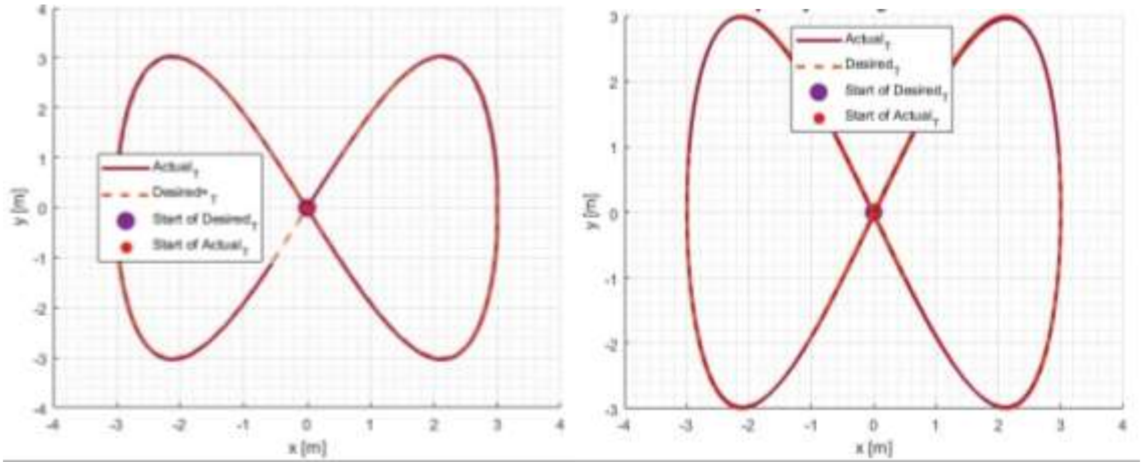
(a) (b)

Figure 26: Figure-eight (a) IMC-based PI (b) SMC-ESO with IMC-BASED PI control after 45 seconds



(a) (b)

Figure 27: Helical (a) IMC-based PI (b) SMC-ESO with IMC-BASED PI control after 60 seconds



(a) (b)

Figure 28: Figure-eight (a) IMC-based PI (b) SMC-ESO with IMC-BASED PI control after 60 seconds

These plots highlighted the improved accuracy and efficiency of the SMC-ESO with IMC-based PI control in tracking complex paths. The SMC-ESO with IMC-based PI control was able to accurately hit the target (desired) path after the first 25 seconds for both helical and figure-eight trajectories, while the system with only the IMC-based PI control is unable to lock on the target (desired) path until after 60 seconds for both the helical and figure-eight trajectories. This shows that the SMC-ESO with IMC-based PI control is more effective when it comes to tracking paths than the IMC-based PI control.

Table 6: Control strategies performance comparison

Metric	**IMC-based PI**	**SMC-ESO + IMC-based PI**
Euler Angle Settling Time (ϕ, θ, ψ)	3.0 s	2.5 s
z-Position Settling Time	2.4 s	2.2 s
x, y Position Settling Time	23.0 s	9.5 s
Trajectory RMSE (x-axis)	0.68 m	0.23 m
Trajectory RMSE (y-axis)	0.73 m	0.26 m
Overshoot in ϕ or θ	~10%	<5%

To quantitatively compare the two control strategies, their performance is reported in Table 6 based on standard control measures such as settling time and root mean square error (RMSE) for trajectory tracking. The hybrid controller (SMC-ESO + IMC-based PI) achieved faster stabilization with Euler angles stabilizing within 2.5 seconds and x, y position within 9.5 seconds. In contrast to the stand-alone IMC-based PI controller, which stabilized position at most within 23 seconds. Accuracy of trajectory following was also significantly improved: RMSE of x and y locations reduced by more than 60% based on the hybrid method. Attitude angle overshooting was also reduced, representing smoother convergence and better robustness under trajectory constraints. These improvements supplement the visual outcomes and fully illustrate the benefit of the hybrid control technique.

5. CONCLUSION

This study explored the mathematical modeling, stabilization, and trajectory tracking of a quadrotor using advanced control strategies. The quadrotor's dynamics were analyzed using the six degrees of freedom (6-DoF) framework, which describes its translational and rotational motion in three-dimensional space. Rigid structure, axis symmetry, and thrust-drag proportionality, were the key assumptions employed to simplify the modeling process. Rotation matrices Euler angles were utilized to define the quadrotor's orientation and to transform coordinates between reference frames.

There are two control strategies involved in this implementation, these are IMC-based PI controller and an approach combining IMC-based PI control with SMC-ESO. The IMC-based PI controller demonstrated effective stabilization of the Euler angles ϕ, θ, and ψ within 3 seconds and the z-position within 2.4 seconds. However, positions x and y required additional control adjustments, including a proportional controller, to achieve stabilization, albeit with longer settling times of 12 seconds for angles and 23 seconds for positions. Overshoots were observed in the ϕ, θ, x, and y states before stabilization.

For trajectory tracking, the hybrid SMC-ESO with IMC-based PI control outperformed the standalone IMC-based PI controller. The hybrid approach enabled the quadrotor to accurately lock onto and follow complex trajectories, such as helical and figure-eight paths, with greater precision and in shorter timeframes. Simulations showed that the hybrid controller achieved accurate trajectory tracking within 25 seconds, while the IMC-based PI controller struggled to match the desired paths even after 60 seconds. SMC-ESO with IMC-based PI control proved to be a robust and efficient strategy for both stabilization and trajectory tracking, addressing the challenges posed by the quadrotor's underactuated and nonlinear.

While current simulations are optimistic for the viability of the suggested hybrid controller, efforts should be put in place by the control engineer to guarantee its use in real-world conditions. The future work can focus on experimental validation using a physical quadrotor testbed, controller performance testing against external disturbance such as wind gusts and payload changes, and integrating sensor noise, communication latency, and actuator constraints into simulation and control design.

These advances will be essential to enabling bridging from simulation to real-world deployment in high-precision, demanding, and autonomous UAV missions.

REFERENCES

[1] Elnady, A., Osama, J., Osama, I., & Khalil, A. (2019). Design and manufacturing of quadcopter. □ *The 4th International under-Graduate Research Conference, IUGRC 2019, July 29th -August 1st, 2019 At: Military Technical College, Cairo, Egypt.*

[2] Pappalardo, C. M., Del Giudice, M., Oliva, E. B., Stieven, L., & Naddeo, A. (2023). Computer-aided design, multibody dynamic modeling, and motion control analysis of a quadcopter system for delivery applications. *Machines*, *11*(4), 464. https://doi.org/10.3390/machines11040464

[3] Abdulkareem, A., Oguntosin, V., Popoola, O. M., & Idowu, A. A. (2022). Modeling and nonlinear control of a quadcopter for stabilization and trajectory tracking. *Journal of Engineering, 2022*, 1–19. https://doi.org/10.1155/2022/2449901

[4] Sampaio, D. D., & Silva, W. L. S. (2021). Sistema nebuloso para navegação autônoma de veículo aéreo não tripulado / Fuzzy system for autonomous unmanned aerial vehicle navigation. *Brazilian Journal of Development*, *7*(4), 34520–34536. https://doi.org/10.34117/bjdv7n4-086

[5] Muhamad, A., Panjaitan, S. D., & Yacoub, R. R. (2024). Design and development of flight controller for quadcopter drone control. *Telecommunications Computers and Electricals Engineering Journal*, *1*(3), 279. https://doi.org/10.26418/telectrical.v1i3.73681

[6] Telli, K., et al. (2023). A comprehensive review of recent research trends on unmanned aerial vehicles (UAVs). *Systems*, *11*(8), 400. https://doi.org/10.3390/systems11080400

[7] Polvara, R., Sharma, S., Wan, J., Manning, A., & Sutton, R. (2018). Vision-based autonomous landing of a quadrotor on the perturbed deck of an unmanned surface vehicle. *Drones*, *2*(2), 15. https://doi.org/10.3390/drones2020015

[8] How, J. P., Frazzoli, E., & Chowdhary, G. V. (2014). Linear flight control techniques for unmanned aerial vehicles. In *Springer eBooks* (pp. 529–576). https://doi.org/10.1007/978-90-481-9707-1_49

[9] Elkholy, H., & Habib, M. K. (2015). Dynamic modeling and control techniques for a quadrotor. In *Advances in Computational Intelligence and Robotics Book Series* (pp. 408–454). https://doi.org/10.4018/978-1-4666-7387-8.ch014

[10] Thanh, H. L. N. N., et al. (2022). Quadcopter UAVs extended states/disturbance observer-based nonlinear robust backstepping control. *Sensors*, *22*(14), 5082. https://doi.org/10.3390/s22145082

[11] Bouabdallah, S., Murrieri, P., & Siegwart, R. (2004). Design and control of an indoor micro quadrotor. In *IEEE International Conference on Robotics and Automation (ICRA '04)* (Vol. 5, pp. 4393–4398). https://doi.org/10.1109/ROBOT.2004.1302409

[12] Sahrir, N. H., & Basri, M. A. M. (2022). Modelling and manual tuning PID control of quadcopter. In *Lecture Notes in Electrical Engineering* (pp. 346–357). https://doi.org/10.1007/978-981-19-3923-5_30

[13] Maaruf, M., Abubakar, A. N., & Gulzar, M. M. (2024). Adaptive backstepping and sliding mode control of a quadrotor. *Journal of the Brazilian Society of Mechanical Sciences and Engineering, 46*(11). https://doi.org/10.1007/s40430-024-05188-z

[14] Alexis, K., Tzes, A., & Nikolakopoulos, G. (2012). Model predictive quadrotor control: Attitude, altitude and position experimental studies. *IET Control Theory and Applications, 6*(12), 1812–1827. https://doi.org/10.1049/iet-cta.2011.0348

[15] Lim, Y., Ramasamy, S., Gardi, A., Kistan, T., & Sabatini, R. (2017). Cognitive human-machine interfaces and interactions for unmanned aircraft. *Journal of Intelligent & Robotic Systems, 91*(3–4), 755–774. https://doi.org/10.1007/s10846-017-0648-9

[16] Gopalakrishnan, E. (2017). *Quadcopter flight mechanics model and control algorithms* (Master's thesis, Czech Technical University in Prague).

[17] Lee, D., Kim, H. J., & Sastry, S. (2009). Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter. *International Journal of Control Automation and Systems, 7*(3), 419–428. https://doi.org/10.1007/s12555-009-0311-8

[18] Uzair, M., Sipra, K., Waqas, M., & Tu, S. (2020, October 16). Applications of MyO armband using EMG and IMU signals. In *2020 IEEE 3rd International Conference on Mechatronics, Robotics and Automation (ICMRA)*. https://doi.org/10.1109/ICMRA51221.2020.9398375

[19] Gao, Y., Lin, J., Yang, L., & Zhu, J. (2016). Development and calibration of an accurate 6-degree-of-freedom measurement system with total station. *Measurement Science and Technology, 27*(12), 125103. https://doi.org/10.1088/0957-0233/27/12/125103

[20] Lee, D., Burg, T., Dawson, D., Shu, D., Xian, B., & Tatlicioglu, E. (2009). Robust tracking control of an underactuated quadrotor aerial-robot based on a parametric uncertain model. In *2009 IEEE International Conference on Systems, Man and Cybernetics (SMC)* (pp. 3187–3192).

[21] Sabatino, F. (2015). *Quadcopter control: Modeling, nonlinear control design, and simulation*.

[22] Amer, T. S., & Abady, I. M. (2017). Solutions of Euler's dynamic equations for the motion of a rigid body. *Journal of Aerospace Engineering, 30*(4). https://doi.org/10.1061/(asce)as.1943-5525.0000736

[23] H. P. R., M. S. Y., & Lakshmanaprabu, S. K. (2019). Internal model controller based PID with fractional filter design for a nonlinear process. *International Journal of Electrical and Computer Engineering (IJECE), 10*(1), 243. https://doi.org/10.11591/ijece.v10i1.pp243-254

[24] Saxena, S., & Hote, Y. (2012). Advances in internal model control technique: A review and future prospects. *IETE Technical Review*, *29*(6), 461. https://doi.org/10.4103/0256-4602.105001

[25] Bucak, İ. Ö. (2020). An in-depth analysis of sliding mode control and its application to robotics. In *IntechOpen eBooks*. https://doi.org/10.5772/intechopen.93027

[26] Yang, X., & Huang, Y. (2009). Capabilities of extended state observer for estimating uncertainties. In *American Control Conference* (pp. 3700–3705). https://doi.org/10.1109/acc.2009.5160642

[27] Li, Y., Zhu, Q., & Elahi, A. (2024). Quadcopter trajectory tracking based on model predictive path integral control and neural network. *Drones*, *9*(1), 9. https://doi.org/10.3390/drones9010009

[28] Yan, D., Zhang, W., Chen, H., & Shi, J. (2022). Robust control strategy for multi-UAVs system using MPC combined with Kalman-consensus filter and disturbance observer. *ISA Transactions*, *135*, 35–51. https://doi.org/10.1016/j.isatra.2022.09.021

[29] Li, B., Liu, H., Ahn, C. K., Wang, C., & Zhu, X. (2024). Fixed-Time tracking control of wheel mobile robot in slipping and skidding conditions. *IEEE/ASME Transactions on Mechatronics*, 1–11. https://doi.org/10.1109/tmech.2024.3401069