Sliding Mode Control based on Backstepping Approach for an UAV Type-Quadrotor

Bhavana B Rao, Sriram Prasad Kothapalli

Abstract—**This paper delves into the development of a sliding mode controller through the application of the backstepping approach. This controller is strategically employed for the synthesis of tracking errors and Lyapunov functions. To provide a comprehensive framework, a novel state-space representation is meticulously formulated by incorporating the dynamics of the quadrotor while accounting for non-holonomic constraints. The presented sliding mode controller is designed to proficiently address system non-linearities and enhance the tracking of predefined trajectories. The ensuing simulation results, presented graphically, offer a nuanced examination of the controller's performance.**

This report provides an intricate examination of the dynamic equations governing the quadrotor, elucidating the subsequent derivation of the state-space equation. A comprehensive exposition is dedicated to explicating the novel control law proposed. MATLAB was employed as the computational tool for conducting simulations. The primary outcomes encompass the dynamic evolution and stabilization of the quadrotor in space, tracking of desired trajectories along the yaw angle ψ and the (X, Y, Z) axes, and an analysis of errors incurred during the trajectory tracking process.

Index Terms—**Quadrotor, Sliding mode, non-holonomic constraints.**

I. Introduction

UNMANNED aerial vehicles (UAVs) have garnered increasing attention due to recent advancements in technology, particularly in instrumentation. This progress has enabled the development of robust systems, such as mini-drones, with actual capabilities for autonomous navigation at a reasonable cost.

Despite notable advancements, researchers still face significant challenges, especially in controlling these systems, particularly when dealing with atmospheric turbulences. Additionally, navigating UAVs is a complex task that involves perceiving often restricted and changing environments, especially during low-altitude flights.

Mini-drones have now permeated various application domains, including airspace and traffic monitoring for safety, supervision of natural risks like volcanic activities, environmental conservation through air pollution measurement and forest monitoring, handling tasks in hazardous environments such as radioactive workspaces and mine clearance, overseeing large infrastructures like dams, power lines, and pipelines, as well as contributing to agriculture and film production through aerial shooting[2].

Unlike ground-based robots, where a kinematics model may often suffice, the control of aerial robots, specifically quadrotors, necessitates dynamics to account for gravitational effects and aerodynamic forces.

In one study, a control law is proposed based on selecting a stabilizing Lyapunov function to achieve desired tracking trajectories along the (X, Z) axis and roll angle, yet it neglects consideration of nonholonomic constraints. Another investigation lacks the inclusion of frictions resulting from aerodynamic torques, drag forces, and nonholonomic constraints while presenting a control law based on backstepping to stabilize the entire system, including translation and orientation. Meanwhile, another work acknowledges gyroscopic effects and demonstrates the applicability of a classical model-independent PD controller for asymptotically stabilizing the quadrotor aircraft's attitude. This study introduces a new Lyapunov function[4], incorporating PD^2 and compensating for coriolis and gyroscopic torques, and a different work focuses on developing a PID controller specifically for altitude stabilization.

Additional research efforts introduce sliding mode and high-order sliding mode as observers to estimate unmeasured states and counteract external disturbances like wind and noise. In the current paper, the primary emphasis is on quadrotor modeling, considering various parameters influencing dynamics, such as frictions from aerodynamic torques, drag forces along (X, Y, Z) axes, gyroscopic effects identified in an experimental quadrotor, and high-order nonholonomic constraints[1]. This approach aims to establish a more comprehensive and realistic state-space representation, addressing the complexities of control law synthesis for such intricate systems.

In order to estimate unmeasured states and compensate for external disturbances like wind and noise, more research efforts use sliding mode and high-order sliding mode as observers[3]. The focus of the present study is quadrotor modeling with different characteristics taken into account that affect dynamics: drag forces along (X, Y, Z) axes, frictions from aerodynamic torques, gyroscopic effects found in an experimental quadrotor, and high-order nonholonomic constraints. With this method, the difficulties of control law synthesis for such complex systems will be addressed, and a more thorough and realistic state-space representation will be established[5].

II. Implementation

A. Quadrotor Dynamic Modeling

The quadrotor is equipped with four propellers arranged in a cross configuration. Specifically, the propellers are arranged in pairs (1,3) and (2,4), turning in opposite directions, as illustrated in Fig. 1. By adjusting the speed of the rotors, it is possible to control the lift force and induce motion. Altering the speeds of all four propellers simultaneously enables vertical motion, while changing the speeds of propellers 2 and 4 induces roll rotation coupled with lateral motion. Similarly, adjusting the speeds of propellers 1 and 3 leads to pitch rotation and corresponding lateral motion. Yaw rotation is a more nuanced effect, arising from the difference in counter-torque between each pair of propellers.

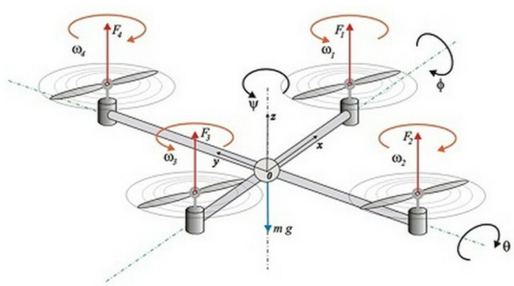

Fig. 1: Pitch

The dynamics of the quadrotor are defined considering a few assumptions, which are as listed below:

- The quadrotor frame is both rigid and symmetrical.
- The center of mass aligns with point o'.
- The propellers maintain rigidity.
- The thrust and drag exhibit a proportional relationship to the square of the propeller speed.

According to the Newton-Euler formulation of Dynamic model:

$$\dot{\xi} = v \tag{1}$$

where,

$\dot{\xi}$ = position vector of quadrotor's center of mass relative to the inertial frame.

v = Linear velocity of the Quadrotor.

According to the Newton's second law of motion:

$$m\ddot{\xi} = F_f + F_t + F_g \tag{2}$$

where,

$\ddot{\xi}$ = Acceleration of the system.

F_f = external forces acting on the system, such as applied forces, thrust, or other non-gravitational forces.

F_t = thrust force generated by the system.

F_g = gravitational force acting on the system.

$$\dot{R} = RS(\Omega) \tag{3}$$

In the above equation R is the rotation matrix and S(Ω) is the skew symmetric matrix with the angular velocity of the frame.

We have obtained skew symmetric matrix from 3D angular velocity vector $\Omega = [\Omega_1 \quad \Omega_2 \quad \Omega_3]^T$
where Ω_1,Ω_2 and Ω_3 are the angular velocities in x,y and z axes repecitvely.
A skew symmetric matrix is obtained as :

$$\begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \tag{4}$$

$$J\dot{\Omega} = -\Omega \wedge J\Omega + \tau_f - \tau_a - \tau_g \tag{5}$$

The above equation is the dynamic equation that describes the rotational motion of the quadrotor, where

J = Inertial tensor, an Identity 3x3 matrix representing how the mass is distributed around the rotational axes.

τ_f = Torque due to external forces.

τ_a = Torque due to acceleration.

τ_g = Torque due to gravitational force.

The J matrix is given as

$$J = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{6}$$

Ω is the angular velocity of the airframe expressed in B and given as:

$$\Omega = \begin{bmatrix} 1 & 0 & -S\theta \\ 0 & C\phi & C\theta S\phi \\ 0 & -S\phi & C\phi C\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{7}$$

The Quadrotor will perform many angular motions and to capture them, a homogeneous transformation matrix is derived by taking the product of rotation matrices of Euler angles as shown below:

$$R = R_\phi * R_\theta * R_\psi \tag{8}$$

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\phi & -S\phi \\ 0 & -S\phi & C\phi \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix} \begin{bmatrix} C\psi & -S\psi & 0 \\ S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{9}$$

The final transformation matrix is given as:

$$R = \begin{bmatrix} C\theta C\psi & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ C\theta S\psi & S\psi S\theta S\phi - C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & S\phi C\theta & C\phi C\theta \end{bmatrix}$$

where, $S \to \sin$ and $C \to \cos$

The resultant force generated by the four rotors

$$R = \begin{bmatrix} C\phi C\psi S\theta + S\theta S\psi \\ C\phi S\theta S\psi - S\phi C\psi \\ C\phi C\theta \end{bmatrix} \sum_{i=1}^{4} F_i$$

The general equation of thrust is given as

$$F = K\omega^2 \tag{10}$$

where, k = Lift constant

Thrust produced by Quadrotor

$$F_z = K(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \tag{11}$$

To determine the total moment τ_f generated by a quadrotor, a systematic approach involves isolating the moments produced in each axis and subsequently combining them into a concise matrix representation. The individual moments in the body-fixed frame are expressed as:

$$\tau_f = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \tag{12}$$

ω_2 and ω_4 spin along the X axis, one of the ω_3 and ω_1 must be greater for the spin so the moment in the X axis is

$$\tau_x = dk(F_3 - F_1) \tag{13}$$

where, d = distance between both the rotors Similarly, for the Y-axis

$$\tau_y = dk(F_4 - F_2) \tag{14}$$

Finally, in Z-direction from Eq (11)

$$\tau_z = k(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \tag{15}$$

Combining Eq (14),(15) and (16)

$$\tau_f = \begin{bmatrix} d(F_3 - F_1) \\ d(F_4 - F_2) \\ K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \tag{16}$$

K_d = drag coefficient

Drag forces along (X,Y,Z) axis

$$F_t \, \alpha \, \dot{\xi}$$

$$F_t = -K_{ft}\dot{\xi}$$

Writing in a matrix form

$$F_t = \begin{bmatrix} -K_{ftx} & 0 & 0 \\ 0 & -K_{fty} & 0 \\ 0 & 0 & -K_{ftz} \end{bmatrix} \dot{\xi} \tag{17}$$

K_{ftx}, K_{fty} and K_{ftz} are the translation drag coefficients along (X,Y,Z) directions. F_g is the gravity force.

$$F_g = \begin{bmatrix} 0 & 0 & -mg \end{bmatrix}^T \tag{18}$$

Resultant aerodynamic friction torques is given by the relation

$$\tau_a \, \alpha \, \Omega^2$$

$$\tau_a = -K_{fa}\Omega^2$$

Writing in a matrix form

$$\tau_a = \begin{bmatrix} -K_{fax} & 0 & 0 \\ 0 & -K_{fay} & 0 \\ 0 & 0 & -K_{faz} \end{bmatrix} \Omega^2 \tag{19}$$

K_{fax}, K_{fay} and K_{ftz} are the friction aerodynamics coefficients along (X,Y,Z) directions.

Defining the applied forces,

$$F_i \, \alpha \, \omega_i^2 \tag{20}$$

$$F_i = K_P \times \omega_i^2 \tag{21}$$

$$M_i = K_d \times \omega_i^2 \tag{22}$$

$$\tag{23}$$

$$\begin{aligned} &\text{Net force:} \quad F = \sum F_i - mgb_3 \\ &\text{Net moment:} \quad M = \sum (r_i \times F_i) + \sum M_i \\ &\text{Newton-Euler equations:} {}^A\omega^B = pb_1 + qb_2 + rb_3 \end{aligned}$$

where p,q,r are angular velocities in body frame

In inertial frame,

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R_B^A \begin{bmatrix} 0 \\ 0 \\ F_1 + F_2 + F_3 + F_4 \end{bmatrix} \tag{24}$$

$$F_1 + F_2 + F_3 + F_4 = U_1 \,\text{Net thrust}$$

Euler's rotation equation:

$$M = I\dot{\omega} + \omega \times (I\omega)$$

Now, in body frame:

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} d(F_3 - F_1) \\ d(F_2 - F_4) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{25}$$

$$\begin{aligned} &d(F_3 - F_1) = U_2 \,\text{ Moment in x-direction} \\ &d(F_2 - F_4) = U_3 \,\text{ Moment in y-direction} \\ &M_1 - M_2 + M_3 - M_4 = U_4 \,\text{ Moment in z-direction} \end{aligned}$$

d is the arm length. And, U1, U2, U3 and U4 are control inputs

$$Let \, \gamma = \frac{K_d}{K_P} = \frac{M_i}{F_i} \tag{26}$$

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} 0 & d & 0 & -d \\ -d & 0 & d & 0 \\ \gamma & -\gamma & \gamma & -\gamma \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{27}$$

From the control inputs defined in equations (24) and (25), we get

$$U = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -d & 0 & d & 0 \\ 0 & d & 0 & -d \\ \gamma & -\gamma & \gamma & -\gamma \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} \tag{28}$$

From equation (21)

$$U = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -d & 0 & d & 0 \\ 0 & d & 0 & -d \\ \gamma & -\gamma & \gamma & -\gamma \end{bmatrix} \begin{bmatrix} K_P\omega_1^2 \\ K_P\omega_2^2 \\ K_P\omega_3^2 \\ K_P\omega_4^2 \end{bmatrix} \tag{29}$$

$$= \begin{bmatrix} K_P & K_P & K_P & K_P \\ -K_Pd & 0 & K_Pd & 0 \\ 0 & K_Pd & 0 & -K_Pd \\ K_P\gamma & -K_P\gamma & K_P\gamma & -K_P\gamma \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{30}$$

$$K_P\gamma = K_P\frac{K_d}{K_P} = K_M \quad \text{From} \quad \text{Eqn.} \quad (26)$$
$$K_d \rightarrow \text{drag} \quad \text{coefficient}$$
$$K_M \rightarrow \text{lift} \quad \text{coefficient}$$

Hence,

$$U = \begin{bmatrix} K_F & K_F & K_F & K_F \\ -K_F d & 0 & K_F d & 0 \\ 0 & K_F d & 0 & -K_F d \\ K_M & -K_M & K_M & -K_M \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{31}$$

We know that,

$$R_B^A = \begin{bmatrix} C\theta C\psi & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ C\theta S\psi & S\psi S\theta S\phi - C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & S\phi C\theta & C\phi C\theta \end{bmatrix}$$

Substituting the rotation matrix in equation (8), we get

$$(24) \rightarrow m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} +$$
$$\begin{bmatrix} C\theta C\psi & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ C\theta S\psi & S\psi S\theta S\phi - C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & S\phi C\theta & C\phi C\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ U_1 \end{bmatrix}$$
$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} U_1(C\psi S\theta C\phi + S\psi S\phi) \\ U_1(S\psi S\theta C\phi - C\psi S\phi) \\ U_1 C\phi C\theta - mg \end{bmatrix}$$

$$\ddot{x} = \frac{U_1(C\psi S\theta C\phi + S\psi S\phi) - K_f tx\dot{x}}{m} \tag{32}$$
$$\ddot{y} = \frac{U_1(S\psi S\theta C\phi - C\psi S\phi) - K_f ty\dot{y}}{m} \tag{33}$$
$$\ddot{z} = \frac{U_1 C\phi C\theta - mg - K_f tz\dot{z}}{m} = \frac{U_1 C\phi C\theta - K_f tz\dot{z}}{m} - g \tag{34}$$

Where $K_f tx, K_f ty, K_f tz$ are drag forces (35)

Next, we define the rotational dynamics in the body frame,

$$J_b\dot{\omega} = \tau_m - \tau_g - (\omega \times J_b\omega)$$

where J_b is inertia matrix of quadrotor in body-frame
τ_m is motor torque
τ_g is gyroscopic torque

$$J_b = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}, \dot{\omega} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix}, J_m\dot{\omega} = \tau_m - \tau_\psi$$

J_b matrix represents the inertia of the quadrotor about its principal axes (X, Y, Z). The diagonal elements are the moments of inertia about the X, Y, and Z axes, respectively.

$$\text{At hover}, \dot{\omega} = 0 \,\text{so}\, \tau_\psi = \tau_D$$
$$\tau_\psi = \tau_D = (-1)^{i+1} K_d \omega_i^2 = K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2)$$

At hover, τ_ψ is the torque due to drag τ_D which is proportional to the square of the angular velocities of the four rotors. K_d is the drag torque coefficient.

$$\tau_{\phi,\theta} = \sum r \times T$$
$$\tau_\phi = dK_T(\omega_4^2 - \omega_2^2)$$
$$\tau_\theta = dK_T(\omega_3^2 - \omega_1^2)$$

These equations represent the torques τ_ϕ, τ_θ, about the roll and pitch axes, respectively. They are influenced by the difference in squared angular velocities of the corresponding rotor pairs. d is the distance from the center of mass to each rotor, and K_T is the thrust coefficient.

$$\tau_m = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \\ dK_T(\omega_4^2 - \omega_2^2) \\ dK_T(\omega_3^2 - \omega_1^2) \end{bmatrix}$$

$$\tau_g = \begin{bmatrix} \dot{\theta} \\ -\dot{\phi} \\ 0 \end{bmatrix} J_r \sum_{i=1}^{4} (-1)^{i+1}\omega_i = \begin{bmatrix} J_r\dot{\theta}(\omega_1 - \omega_2 + \omega_3 - \omega_4) \\ -J_r\dot{\phi}(\omega_1 - \omega_2 + \omega_3 - \omega_4) \\ 0 \end{bmatrix}$$

The first component $J_r\dot{\theta}(\omega_1 - \omega_2 + \omega_3 - \omega_4)$ represents the gyroscopic torque about the yaw axis (θ) due to the angular velocity differences between the rotors. The second component $-J_r\dot{\phi}(\omega_1 - \omega_2 + \omega_3 - \omega_4)$ represents the gyroscopic torque about the roll axis (ϕ) due to the angular velocity differences between the rotors. The third component is zero, indicating no gyroscopic torque about the pitch axis.

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

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{1}{I_x}[qr(I_y - I_z) - J_r q\Omega + dK_T(\omega_4^2 - \omega_2^2)] \\ \frac{1}{I_Y}[pr(I_z - I_x) - J_r p\Omega + dK_T(\omega_1^2 - \omega_3^2)] \\ \frac{1}{I_z}[pq(I_x - I_y) + K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2)] \end{bmatrix}$$

$$\ddot{\phi} = \frac{1}{I_x}\dot{\theta}\dot{\psi}(I_y - I_z) - J_r\bar{\Omega}\dot{\phi} + LU_2 - K_f ax\dot{\phi}^2 \tag{36}$$
$$\ddot{\theta} = \frac{1}{I_y}\dot{\phi}\dot{\psi}(I_z - I_x) - J_r\bar{\Omega}\dot{\theta} + LU_3 - K_f ay\dot{\theta}^2 \tag{37}$$
$$\ddot{\psi} = \frac{1}{I_z}\dot{\phi}\dot{\theta}(I_x - I_y) + K_d U_4 - K_f azp\dot{s}i^2 \tag{38}$$
$$\text{where}\, \Omega = \omega_1 - \omega_2 + \omega_3 - \omega_4 \tag{39}$$

Equation (32), (33), (34), (36), (37) and (38) define the complete quadrotor dynamic equations.

B. Rotor Dynamics

The rotor comprises a D.C motor that drives a propeller through a reducer. The dynamic equations governing the D.C motor are as follows: Applying KVL to the circuit in fig 2,

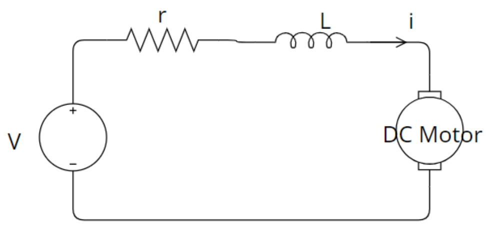

Fig. 2: Circuit diagram of a motor

$V = ri + L\dfrac{di}{dt} + e$

e is the back-emf and is given by

$e = k_e\omega$

$V = riL + \dfrac{di}{dt} + k_e\omega$

k_c is , a constant related to motor construction

Using Newton's second law for rotational motion and the electrical equation for a DC motor, we get
Torque = Inertia × Angular Acceleration + Damping Torque

$$k_m i = J_r\frac{d\omega}{dt} + C_s + k_r\omega^2$$

Where,
k_m : mechanical torque constant
k_r : load constant torque
r : motor internal resistance
J_r : rotor inertia
C_s : solid friction

From the above equation,
$\dot{\omega} = \frac{k_m i}{J_r} - \frac{C_s}{J_r} - \frac{k_r\omega^2}{J_r}$

Substituting this in voltage equation,
$\dot{\omega}_i = bV_i - \beta_0 - \beta_1\omega_i - \beta_2\omega_i^2$
where, $\beta_0 = \frac{C_s}{J_r}$, $\beta_1 = \frac{k_e k_m}{rJ_r}$, $\beta_2 = \frac{k_r}{J_r}$, $b = \frac{k_m}{rJ_r}$

C. State-Space Equation

Considering, the state space equation to be

$$X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{bmatrix} = \begin{bmatrix} \phi \\ \dot{\phi} \\ \theta \\ \dot{\theta} \\ \psi \\ \dot{\psi} \\ x \\ \dot{x} \\ y \\ \dot{y} \\ z \\ \dot{z} \end{bmatrix}$$

The state space form is as given ,$\dot{X} = f(X) + g(X,U) + \delta$

From the dynamic equations we derive the state space representation:

$$\dot{x}_1 = x_2 \tag{40}$$
$$\dot{x}_2 = a_1x_4x_6 + a_2x_2^2 + a_3\bar{\Omega}x_4 + b_1U_2 \tag{41}$$
$$\dot{x}_3 = x_4 \tag{42}$$
$$\dot{x}_4 = a_4x_2x_6 + a_5x_4^2 + a_6\bar{\Omega}x_2 + b_2U_3 \tag{43}$$
$$\dot{x}_5 = x_6 \tag{44}$$
$$\dot{x}_6 = a_7x_2x_4 + a_8x_6^2 + b_3U_4 \tag{45}$$
$$\dot{x}_7 = x_8 \tag{46}$$
$$\dot{x}_8 = a_9x_8 + U_x\frac{U_1}{m} \tag{47}$$
$$\dot{x}_9 = x_{10} \tag{48}$$
$$\dot{x}_{10} = a_{10}x_{10} + U_y\frac{U_1}{m} \tag{49}$$
$$\dot{x}_{11} = x_{12} \tag{50}$$
$$\dot{x}_{12} = a_{11}x_{12} + U_1\frac{Cx_1Cx_3}{m} - g \tag{51}$$

The above equations are framed, considering the below defined constants

$$a_1 = [\frac{I_y - I_z}{I_x}], a_2 = \frac{-K_{fax}}{I_x}, a_3 = \frac{-J_r}{I_x}$$
$$a_4 = [\frac{I_z - I_x}{I_y}], a_5 = \frac{-K_{fay}}{I_y}, a_6 = \frac{J_r}{I_y}$$
$$a_7 = [\frac{I_x - I_y}{I_z}], a_8 = \frac{-K_{faz}}{I_z}, a_9 = \frac{-K_{ftx}}{m}$$
$$a_{10} = \frac{-K_{fty}}{m}, a_{11} = \frac{-K_{ftz}}{m}$$
$$b_1 = \frac{d}{I_x}, b_2 = \frac{d}{I_y}, b_3 = \frac{1}{I_z}$$
$$U_x = Cx_1Sx_3Cx_5 + Sx_1Sx_5, U_y = Cx_1Sx_3Sx_5 - Sx_1Cx_5$$

III. Sliding Mode Control

Applying the recursive backstepping approach as an algorithm for synthesizing control laws simplifies all stages of computation related to tracking errors and Lyapunov functions in the following manner:

$$\text{For } i \in \{1,3,5,7,9,11\} : \begin{cases} z_i = x_i d - x_i \\ V_i = \frac{z_i^2}{2} \end{cases} \tag{52}$$

$$\text{For } i \in \{2,4,6,8,10,12\} : \begin{cases} z_i = x_i - \dot{x}_{(i-1)d} - \alpha_{i-1} z_{i-1} \\ V_i = \frac{V_{i-1} + z_i^2}{2} \end{cases} \tag{53}$$

where, $\alpha_i > 0, \forall i \in [1, 12]$
The sliding surfaces are as given below:

$$S_\phi = z_2 = x_2 - \dot{x}_{1d} - \alpha_1 z_1$$
$$S_\theta = z_4 = x_4 - \dot{x}_{3d} - \alpha_3 z_3$$
$$S_\psi = z_6 = x_6 - \dot{x}_{5d} - \alpha_5 z_5$$
$$S_x = z_8 = x_8 - \dot{x}_{7d} - \alpha_7 z_7$$
$$S_y = z_{10} = x_{10} - \dot{x}_{9d} - \alpha_9 z_9$$
$$S_z = z_{12} = x_{12} - \dot{x}_{11d} - \alpha_{11} z_{11}$$

S_ϕ, S_θ, S_ψ, S_x, S_y and S_z are the dynamic sliding surfaces. To formulate a stabilizing control law through sliding mode control, it is imperative to validate the requisite sliding condition (i.e.S$\dot{S}$ ¡ 0). Consequently, the resultant stabilizing control laws can be expressed as follows:

U$_2 = \frac{1}{b_1}\{-q_1 sign(S_\phi) - k_1 S_\phi - a_1 x_4 x_6 - a_2 x_2^2 - a_3 \bar{\Omega} x_4 + \ddot{\phi}_d + \alpha_1(\dot{\phi}_d - x_2)\}$
U$_3 = \frac{1}{b_2}\{-q_2 sign(S_\theta) - k_2 S_\phi - a_4 x_2 x_6 - a_5 x_4^2 - a_6 \bar{\Omega} x_2 + \ddot{\theta}_d + \alpha_3(\dot{\theta}_d - x_4)\}$
U$_4 = \frac{1}{b_3}\{-q_3 sign(S_\psi) - k_3 S_\psi - a_7 x_2 x_4 - a_8 x_6^2 + \ddot{\psi}_d + \alpha_5(\dot{\psi}_d - x_6)\}$
U$_x = \frac{m}{U_1}\{-q_4 sign(S_x) - k_4 S_x - a_9 x_8 + \ddot{x}_d + \alpha_7(\dot{} x_d - x_8)\} U_1 \neq 0$
U$_y = \frac{m}{U_1}\{-q_5 sign(S_y) - k_5 S_y - a_1 0 x_1 0 + \ddot{y}_d + \alpha_9(\dot{} y_d - y_{10})\} U_1 \neq 0$
U$_1 = \frac{m}{C_\phi C_\theta}\{-q_6 sign(S_z) - k_6 S_z - a_{11} x_{12} + \ddot{z}_d + \alpha_{11}(\dot{} z_d - x_1 2) + g\}$(54)

Proof for obtaining equation for U_2 From the backstepping approach, we have found the values for z_i and V_i from Eqs 52 and 53 We get,

$$V_2 = \frac{1}{2} z_1^2 + \frac{1}{2} Z_2^2 \tag{55}$$

$$z_2 = x_2 - \dot{x}_{1d} - \alpha_1 z_1 \tag{56}$$

From Eq:

$$S_\phi = z_2 = x_2 - \dot{x}_{1d} - \alpha_1 z_1 \tag{57}$$

$$V_2 = \frac{1}{2} z_1^2 + \frac{1}{2} S_\phi^2 \tag{58}$$

With the Equations

$$\dot{V}_2 = z_1 \dot{z}_1 + S_\phi \dot{S}_\phi \tag{59}$$

$$\dot{V}_2 = z_1 \dot{z}_1 + S_\phi\{a_1 x_4 x_6 + a_2 x_2^2 + a_3 \bar{\Omega} x_4 + b_1 U_2 + \ddot{\phi}_d - \alpha_1(\dot{\phi}_d - x_2)\} \tag{60}$$

The chosen law for the attractive surface is the time derivative satisfying ($S_\phi \dot{S}_\phi < 0$):

$$\begin{aligned} \dot{S}_\phi &= -q sign(S_\phi) - k_1 S_\phi \\ &= \dot{x}_2 - \ddot{x}_{1d} - \alpha_1 \dot{z}_1 \\ &= a_1 x_4 x_6 + a_2 x_2^2 + a_3 \bar{\Omega} x_4 + b_1 U_2 - \ddot{\phi}_d + \alpha_1(\dot{\phi}_d - x_2) \end{aligned}$$

Finally, the control input U_2 is extracted from Backstepping approach:

$$U_2 = \frac{1}{b_1}\{-q_1 sign(S_\phi) - k_1 S_\phi - a_1 x_4 x_6 - a_2 x_2^2 - a_3 \bar{\Omega} x_4 + \ddot{\phi}_d + \alpha_1(\dot{\phi}_d - x_2)\} \tag{61}$$

Same steps are carried to extract U_3,U_4,U_x,U_y and U_1.

IV. Optimization of Parameters

The control equations described earlier include parameters α_i, k_i, and q_i that directly influence the output. These parameters constant and real. The objective is to select values for these parameters that minimize the error between the desired and obtained trajectories. To achieve this optimization, we employ the optimization capabilities of the SciPy library in Python. Specifically, we utilize the ‘minimize‘ function, which employs the **BFGS (Broyden-Fletcher-Goldfarb-Shanno)** algorithm. This iterative optimization algorithm is well-suited for addressing unconstrained nonlinear optimization problems. The algorithm is executed to discover the most suitable values that yield minimal error within the range of 0 to 1000. The outcomes obtained for these parameter values post-optimization are as follows: α_i **= 0.2285737,** k_i **= 0.1, and** q_i **= 0.1.**

V. Simulation Results

The sliding mode control for a quadrotor is implemented in Matlab. The value of parameters considered for simulation are:

- m = 486 gm
- d = 25 cm
- β_0 = 189.63
- β_1 = 6.0612
- β_2 = 0.0122
- b = 280.19
- g = 9.8 m/s^2
- I_x = 3.8278 x 10^-3 kg/m^2
- I_y = 3.8288 x 10^-3 kg/m^2
- I_z = 7.6566 x 10^-3 kg/m^2
- K_{fax} = 5.567 x 10^-4 N/rad/s
- K_{fay} = 5.567 x 10^-4 N/rad/s
- K_{faz} = 6.354 x 10^-4 N/rad/s
- K_{ftx} = 5.567 x 10^-4 N/m/s
- K_{fty}= 5.567 x 10^-4 N/m/s
- K_{ftz} = 6.354 x 10^-4 N/m/s
- J_r = 2.8385 x 10^-5 Nm/rad/s^2

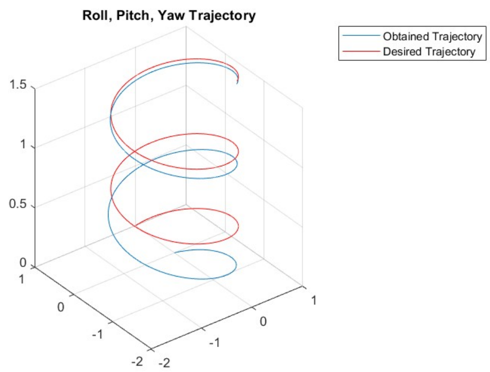

Fig. 3: Rotational Trajectory

As the trajectory has not been mentioned in the paper, the desired trajectory is assumed to be ϕ = sin(t), θ = cos(t), ψ = 0.1*t. From Figure 3, it can be noted that the achieved trajectory closely aligns with the intended one, and the error reduces over time.

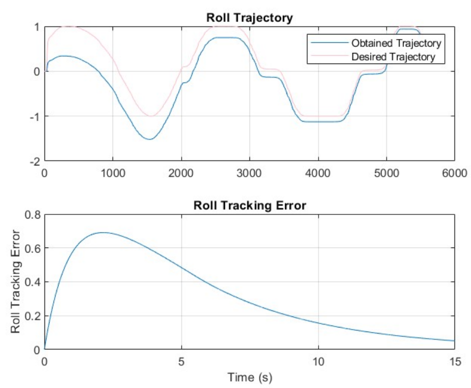

Fig. 4: Roll

Figure 4 consists the plot of desired and obtained trajectory in ϕ direction and the error between the two. The desired trajectory is considered to be sin(t). The distortions in the sin graph are observed due to the control gains, alpha values and ODE Solver configuration.

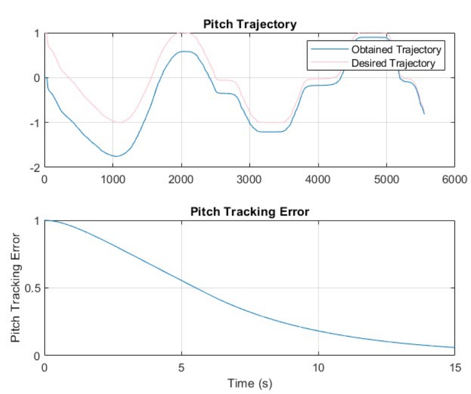

Fig. 5: Pitch

Figure 5 consists the plot of desired and obtained trajectory in θ direction and the error between the two. The desired trajectory is considered to be cos(t).

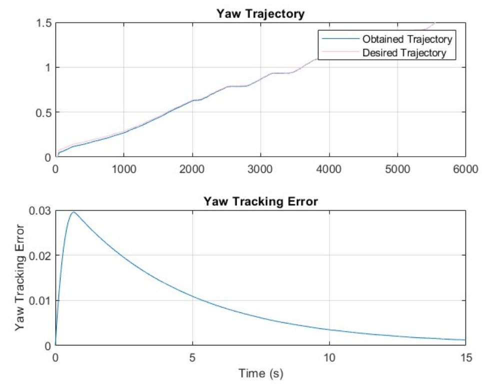

Fig. 6: Yaw

Figure 6 consists the plot of desired and obtained trajectory in ψ direction and the error between the two. The desired trajectory is considered to be 0.1*t.

Similarly, we assume the trajectory in translational directions to be x=sin(t), y = 2*t , z = 3*t.

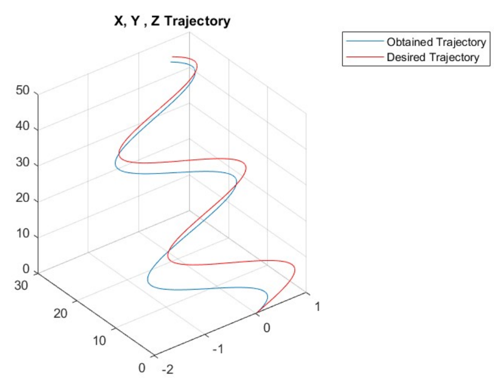

Fig. 7: Translational Trajectory

Figure 7 is a 3-D plot of the trajectory in x,y and z direction. It can be observed that the desired and obtained trajectories converge after the sliding mode control algorithm has been implemented.

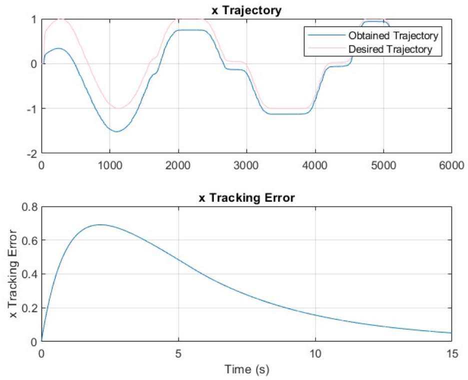

Fig. 8: x trajectory

Figure 8 consists the plot of desired and obtained trajectory in x direction and the error between the two. The desired trajectory is considered to be sin(t).

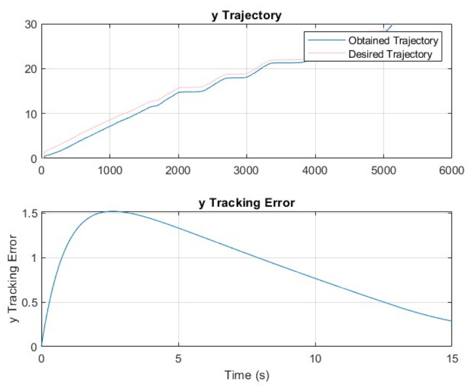

Fig. 9: y trajectory

Figure 9 consists the plot of desired and obtained trajectory in y direction and the error between the two. The desired trajectory is considered to be 2*t.

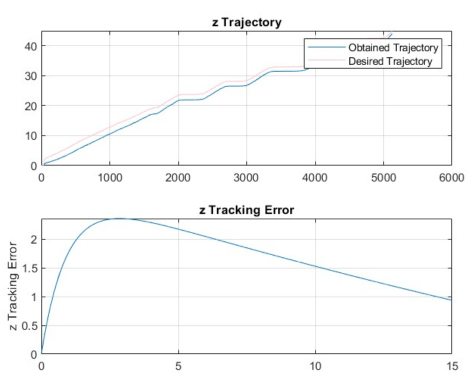

Fig. 10: z trajectory

Figure 10 consists the plot of desired and obtained trajectory in y direction and the error between the two. The desired trajectory is considered to be 3*t.

VI. Conclusion

In this report, we presented stabilizing control laws synthesis by sliding mode based on the backstepping approach that were published in the paper. Firstly, we start with the development of the dynamic model of the quadrotor taking into account the different physics phenomena that can influence the evolution of the system in space; The control laws presented herein facilitate the precise tracking of diverse trajectories specified in terms of the center of mass coordinates of the quadrotor. This capability remains robust in the face of the inherent complexity encapsulated by the proposed model. Notably, our approach successfully navigates the challenges posed by the intricate dynamics of the quadrotor, allowing for effective trajectory tracking. The complexity of the model necessitates the formulation of several assumptions to derive the trajectories, aligning with the intricacies of the physical system under consideration.The presented control laws serve as a testament to the efficacy of the sliding mode technique and the thoughtful incorporation of the backstepping approach, providing a robust foundation for trajectory tracking in complex, real-world scenarios.

References

[1] A.Tayebi, S.Mcgilvray, 2004 *"Attitude stabilization of a four-rotor aerial robot"*, IEEE conference on decision and control, December 14-17, Atlantis Paradise Island,Bahamas 1216-1217.

[2] Hamel T. Mahoney r. Lozano r. Et Ostrowski j,2002.*"Dynamic modelling and configuration stabilization for an X4-flyer."* In the 15éme IFAC world congress', Barcelona, Spain.

[3] P. Pounds, R.Mahony, 2002 *"Design of a four rotor"*ICRA, Auckland.

[4] R.Lozano, P.Castillo, and A.Dzul, 2004 *"global stabilization of the PVTOL: real time application to a mini aircraft"* International Journal of Control, Vol 77, Number 8, pp 735-740, may.

[5] Derafa L. Madani t. and Benallegue A, 2006 *"dynamic modelling and experimental identification of four rotor helicopter parameters"* ICIT Mumbai, India.

[6] S.Bouabdellah, A.Noth and R.Siegwart, 2004 *"modelling of the (OS4) quadrotor"* modelling course, EPFL, may.

[7] A.Benchaib, F.Boudjema and A.Rachid *"sliding mode flux observer based on backstepping approach for induction motor"* World Automation congress WAC'98, Alaska, USA, may 1998.

[8] M. Fliess, C. Join, and H. Sira-Ramirez, "Non-linear estimation is easy," *Int. J. Model., Ident. Control*, vol. 4, no. 1, pp. 12–27, 2008.

[9] A.Mokhtari, N.K.M'sirdi, K.Meghriche and A.Belaidi *"feedback linearization and linear observer for a quadrotor unmanned aerial vehicle"* Advanced Robotics, Vol.20, No.1, pp. 71-91, 2006.

[10] W.Khalil, Dombre, 2002 *"modelling, identification and control of robots"* HPS edition.