ROBOTICS ENGINEERING DEPARTMENT, WORCESTER POLYTECHNIC INSTITUTE

Robust Trajectory Tracking for Quadrotor UAVs using Sliding Mode Control

Fenil Desai, Samarth Shah

May 2, 2022

1 TRAJECTORY GENERATION

To generate a quintic polynomial trajectory for the quadrotor (Crazyflie 2.0), six points in space were considered. The points are $p_0(0,0,0)$, $p_1(0,0,1)$, $p_2(1,0,1)$, $p_3(1,1,1)$, $p_4(0,1,1)$ and $p_5(0,0,1)$. The time steps used for reaching these points are $t_0 = 0s, t_1 = 5s$, $t_2 = 20s$, $t_3 = 35s$, $t_4 = 50s$ and $t_5 = 65s$ respectively. The desired velocity and acceleration at the **start** and **end** for each section (between two time steps mentioned above) of the trajectory is considered to be **zero**.

The position $p(t)$ in any one direction (i.e, x,y or z) at any time instance t can be calculated using a quintic polynomial trajectory as shown in the following formulation presented in Equation (1.1).

$$p(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \tag{1.1}$$

The coefficients of the quintic polynomial trajectory, i.e, a_0, a_1, a_2, a_3, a_4 and a_5 can be calculated by the formulation seen in Equation (1.2).

$$\begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 & t_0^4 & t_0^5 \\ 0 & 1 & 2t_0 & 3t_0^2 & 4t_0^3 & 5t_0^4 \\ 0 & 0 & 2 & 6t_0 & 12t_0^2 & 20t_0^3 \\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 & t_f^5 \\ 0 & 1 & 2t_f & 3t_f^2 & 4t_f^3 & 5t_f^4 \\ 0 & 0 & 2 & 6t_f & 12t_f^2 & 20t_f^3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} p_0 \\ \dot{p}_0 \\ \ddot{p}_0 \\ p_f \\ \dot{p}_f \\ \ddot{p}_f \end{bmatrix} \tag{1.2}$$

Where, t_0 and t_f represents the start and end (finish) time for a particular section of the trajectory. p_0 and p_f represents the start and end position, whereas $\dot{p}_0$, $\dot{p}_f$, $\ddot{p}_0$ and $\ddot{p}_f$ represents the initial and final velocities and acceleration of this section of the trajectory.

Here, for the sake of this project, since the trajectories are piece-wise continuous we will get different coefficients of the trajectory for different sections of the desired trajectory. This trajectory is evaluated piece-wise (section-to-section) in only one direction (i.e either x, y or z). For generating the coefficients for each section of the trajectory, following values were used in the formulation (1.2).

- Trajectory in X coordinate
  - For $t_0 = 0s$ to $t_f = 5s$
    - $p_0 = 0$ m
    - $p_f = 0$ m
  - For $t_0 = 5s$ to $t_f = 20s$
    - $p_0 = 0$ m
    - $p_f = 1$ m
  - For $t_0 = 20s$ to $t_f = 35s$
    - $p_0 = 1$ m
    - $p_f = 1$ m
  - For $t_0 = 35s$ to $t_f = 50s$
    - $p_0 = 1$ m
    - $p_f = 0$ m
  - For $t_0 = 50s$ to $t_f = 65s$
    - $p_0 = 0$ m
    - $p_f = 0$ m
- Trajectory in Y coordinate
  - For $t_0 = 0s$ to $t_f = 5s$
    - $p_0 = 0$ m
    - $p_f = 0$ m
  - For $t_0 = 5s$ to $t_f = 20s$
    - $p_0 = 0$ m
    - $p_f = 0$ m
  - For $t_0 = 20s$ to $t_f = 35s$
    - $p_0 = 0$ m

        * $p_f = 1$ m
    - For $t_0 = 35s$ to $t_f = 50s$
        * $p_0 = 1$ m
        * $p_f = 1$ m
    - For $t_0 = 50s$ to $t_f = 65s$
        * $p_0 = 1$ m
        * $p_f = 0$ m

- Trajectory in Z coordinate
    - For $t_0 = 0s$ to $t_f = 5s$
        * $p_0 = 0$ m
        * $p_f = 1$ m
    - For $t_0 = 5s$ to $t_f = 20s$
        * $p_0 = 1$ m
        * $p_f = 1$ m
    - For $t_0 = 20s$ to $t_f = 35s$
        * $p_0 = 1$ m
        * $p_f = 1$ m
    - For $t_0 = 35s$ to $t_f = 50s$
        * $p_0 = 1$ m
        * $p_f = 1$ m
    - For $t_0 = 50s$ to $t_f = 65s$
        * $p_0 = 1$ m
        * $p_f = 1$ m

Now the piece wise trajectory generated in all the three directions are combined together to obtain the overall position (x,y and z) of the drone at any time instance t. Note that the velocity ($\dot{p}$) and acceleration ($\ddot{p}$) at all the intermediate points is considered to be zero for the trajectories in all three directions. The obtained trajectory is plotted as seen in Figure 1. using a python script.

Visualization of the obtained trajectory in the 3D space is seen in Figure 1. The desired velocity profile in X, Y and Z directions are as seen in Figure 2., 3. and 4. respectively. Also, the desired acceleration profile in X, Y and Z directions is as seen in Figure 5., 6 and 7. respectively.

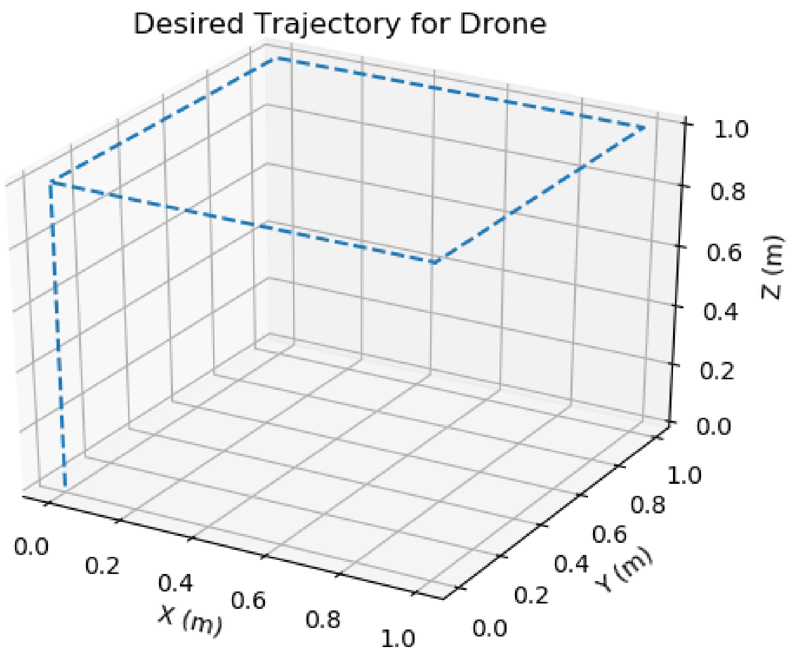

Figure 1.1: Desired Trajectory of Drone in 3D space

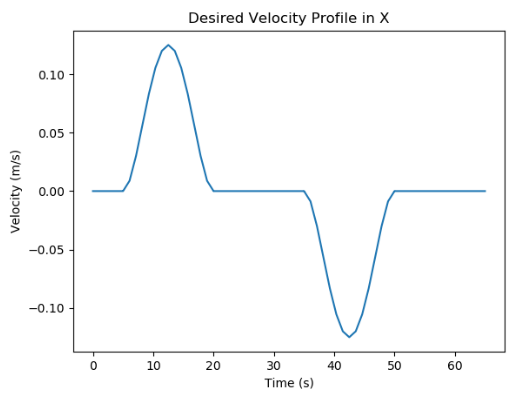

Figure 1.2: Desired Velocity Profile in X direction

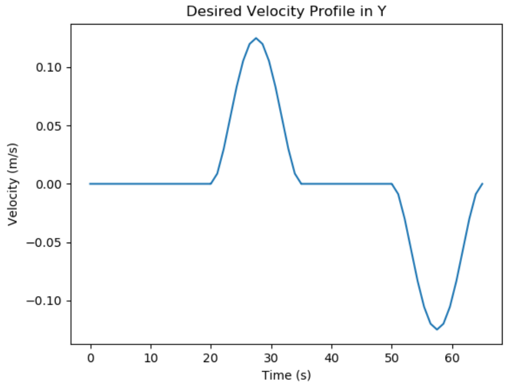

Figure 1.3: Desired Velocity Profile in Y direction

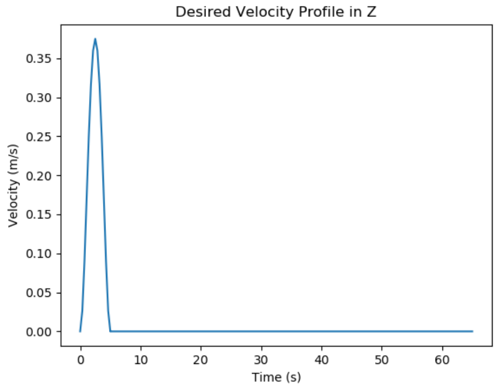

Figure 1.4: Desired Velocity Profile in Z direction

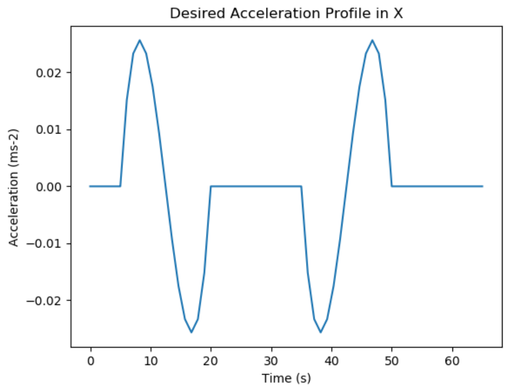

Figure 1.5: Desired Acceleration Profile in X direction

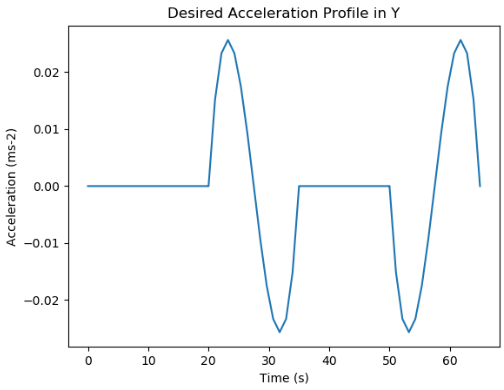

Figure 1.6: Desired Acceleration Profile in Y direction

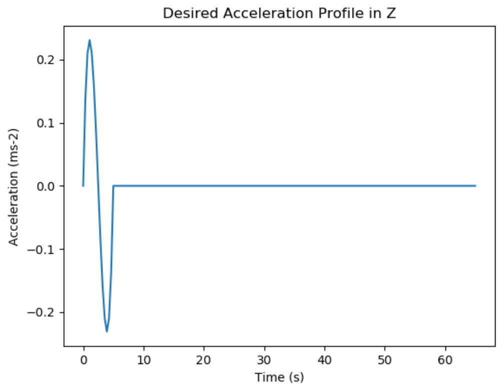

Figure 1.7: Desired Acceleration Profile in Z direction

2 Sliding Mode Controller Design

Four different and independent **Sliding Mode Controllers** are designed to control the height z of the drone and the roll ϕ, pitch θ and yaw ψ angles of the drone. The control inputs u_1, u_2, u_3 and u_3 corresponds to the thrust force from all the propellers and moments applied about the body frame axes by the propellers respectively.

2.1 Sliding Model Control Design for Height (Z)

Equation of Motion in Z direction is give as

$$\ddot{z} = \frac{(\cos\phi\cos\theta)\,u_1}{m} - g \tag{2.1}$$

This equation of motion is a SISO system and is expressed in the control affine form, therefore to design a sliding mode control let us define error for this system.

$$e = z - z_d \tag{2.2}$$

$$\dot{e} = \dot{z} - \dot{z_d} \tag{2.3}$$

$$\ddot{e} = \ddot{z} - \ddot{z_d} \tag{2.4}$$

Where, z is the current height of the drone, z_d is the desired height of the drone obtained from the desired trajectory, $\dot{z}$ is the current velocity of the drone in the z-direction, $\dot{z_d}$ is the desired velocity of the drone in the z-direction, similarly $\ddot{z}$ is the current acceleration of the drone in z-direction and $\ddot{z_d}$ is the desired acceleration of the drone in z-direction.

Now let us express this second order equation of motion as a first order problem in s. Such that $\dot{s}$ contains u_1 (control input) and if s is zero that implies e, $\dot{e}$ and $\ddot{e}$ is zero.

$$s = \dot{e} + \lambda e \tag{2.5}$$

$$\dot{s} = \ddot{e} + \lambda\dot{e} \tag{2.6}$$

where, λ is a positive tunable parameter, which determines how quick is the rate of sliding.

To design u_1 the sliding condition is to be satisfied, which is given by

$$s\dot{s} \leq -k|s| \tag{2.7}$$

Therefore,

$$s(\ddot{e} + \lambda\dot{e}) \leq -k|s| \tag{2.8}$$

$$s(\ddot{z} - \ddot{z_d} + \lambda\dot{e}) \leq -k|s| \tag{2.9}$$

From the equations of motion we can write

$$s\left(\frac{(\cos\phi\cos\theta)\,u_1}{m} - g - \ddot{z_d} + \lambda\dot{e}\right) \leq -k|s| \tag{2.10}$$

Now the task is to select u_1 such that the above condition is always satisfied. Which can be done by selecting the following formulation for u_1

$$u_1 = \frac{-k * sgn(s) + \ddot{z}_d - \lambda \dot{e} + g}{\frac{\cos\phi\cos\theta}{m}} \tag{2.11}$$

Here, $sgn(s)$ is the function that return sign (negative or positive) of s. However, the project demands implementation of sliding mode control with a boundary layer. Therefore, instead of $sgn(s)$ we can use $sat(\frac{s}{\alpha})$, where $sat()$ is a saturation function and α is boundary layer. Therefore the new u_1 becomes

$$u_1 = \frac{-k * sat(\frac{s}{\alpha}) + \ddot{z}_d - \lambda \dot{e} + g}{\frac{\cos\phi\cos\theta}{m}} \tag{2.12}$$

2.2 Sliding Mode Control For the Roll (ϕ)

Equation of Motion in ϕ is give as

$$\ddot{\phi} = \dot{\theta}\dot{\psi}(\frac{I_y - I_z}{I_x}) - (\frac{I_p}{I_x})\Omega\dot{\theta} + \frac{u_2}{I_x} \tag{2.13}$$

Wehere, $\Omega = \omega_1 - \omega_2 + \omega_3 - \omega_4$ and ω_i is the rotor speeds for all four rotors of the drone. This equation of motion is a SISO system and is expressed in the control affine form, therefore to design a sliding mode control let us define error for this system.

$$e = \phi - \phi_d \tag{2.14}$$

$$\dot{e} = \dot{\phi} - \dot{\phi_d} \tag{2.15}$$

$$\ddot{e} = \ddot{\phi} - \ddot{\phi_d} \tag{2.16}$$

Where, ϕ is the current roll angle of the drone, ϕ_d is the desired roll angle of the drone obtained from the desired trajectory, $\dot{\phi}$ is the current angular velocity of the drone around body axis x, $\dot{\phi_d}$ is the desired angular velocity of the drone around body axis x, similarly $\ddot{\phi}$ is the current angular acceleration of the drone around body axis x and $\ddot{\phi_d}$ is the desired angular acceleration of the drone around body axis x.

To estimate ϕ_d, we have used the following formulations:

$$F_y = m\Big(- K_p(y - y_d) - K_d(\dot{y} - \dot{y_d}) + \ddot{y_d}\Big) \tag{2.17}$$

$$\phi_d = \arcsin(\frac{-F_y}{u_1}) \tag{2.18}$$

Where, y is the position of the drone in y direction, and $\dot{y}$ is the velocity of the drone in the y direction. Whereas, the y_d and $\dot{y_d}$ is desired position and velocity in y direction which can be obtained from the desired trajectory designed above and K_p and K_d are tunable parameters.

Now let us express this second order equation of motion as a first order problem in s. Such that $\dot{s}$ contains u_2 (control input) and if s is zero that implies e, $\dot{e}$ and $\ddot{e}$ is zero.

$$s = \dot{e} + \lambda e \tag{2.19}$$

$$\dot{s} = \ddot{e} + \lambda \dot{e} \tag{2.20}$$

where, λ is a positive tunable parameter, which determines how quick is the rate of sliding.

To design u_2 the sliding condition is to be satisfied, which is given by

$$s\dot{s} \leq -k|s| \tag{2.21}$$

Therefore,

$$s(\ddot{e} + \lambda \dot{e}) \leq -k|s| \tag{2.22}$$

$$s(\ddot{\phi} - \ddot{\phi}_d + \lambda \dot{e}) \leq -k|s| \tag{2.23}$$

From the equations of motion we can write

$$s\Big(\dot{\theta}\dot{\psi}(\frac{I_y - I_z}{I_x}) - (\frac{I_p}{I_x})\Omega\dot{\theta} + \frac{u_2}{I_x} - \ddot{\phi}_d + \lambda \dot{e}\Big) \leq -k|s| \tag{2.24}$$

Now the task is to select u_2 such that the above condition is always satisfied. Which can be done by selecting the following formulation for u_2

$$u_2 = -I_x K * sgn(s) - \dot{\theta}\dot{\psi}(I_y - I_z) + I_p\Omega\dot{\theta} + I_x\ddot{\phi}_d - I_x\lambda\dot{e} \tag{2.25}$$

Here, $sgn(s)$ is the function that return sign (negative or positive) of s. However, the project demands implementation of sliding mode control with a boundary layer. Therefore, instead of $sgn(s)$ we can use $sat(\frac{s}{\alpha})$, where $sat()$ is a saturation function and α is boundary layer. Therefore the new u_2 becomes

$$u_2 = -I_x K * sat(\frac{s}{\alpha}) - \dot{\theta}\dot{\psi}(I_y - I_z) + I_p\Omega\dot{\theta} + I_x\ddot{\phi}_d - I_x\lambda\dot{e} \tag{2.26}$$

2.3 Sliding Mode Control For the Pitch (θ)

Equation of Motion in θ is give as

$$\ddot{\theta} = \dot{\phi}\dot{\psi}(\frac{I_z - I_x}{I_y}) + (\frac{I_p}{I_y})\Omega\dot{\phi} + \frac{u_3}{I_y} \tag{2.27}$$

Where, $\Omega = \omega_1 - \omega_2 + \omega_3 - \omega_4$ and ω_i is the rotor speeds for all four rotors of the drone. This equation of motion is a SISO system and is expressed in the control affine form, therefore to design a sliding mode control let us define error for this system.

$$e = \theta - \theta_d \tag{2.28}$$
$$\dot{e} = \dot{\theta} - \dot{\theta_d} \tag{2.29}$$
$$\ddot{e} = \ddot{\theta} - \ddot{\theta_d} \tag{2.30}$$

Where, θ is the current pitch angle of the drone, θ_d is the desired pitch angle of the drone obtained from the desired trajectory, $\dot{\theta}$ is the current angular velocity of the drone around body axis y, $\dot{\theta_d}$ is the desired angular velocity of the drone around body axis y, similarly $\ddot{\theta}$ is the current angular acceleration of the drone around body axis y and $\ddot{\theta_d}$ is the desired angular acceleration of the drone around body axis y.

To estimate θ_d, we have used the following formulations:

$$F_x = m\Big(-K_p(x - x_d) - K_d(\dot{x} - \dot{x_d}) + \ddot{x_d}\Big) \tag{2.31}$$

$$\theta_d = \arcsin(\frac{F_x}{u_1}) \tag{2.32}$$

Where, x is the position of the drone in x direction, and $\dot{x}$ is the velocity of the drone in the x direction. Whereas, the x_d and $\dot{x_d}$ is desired position and velocity in x direction which can be obtained from the desired trajectory designed above and K_p and K_d are tunable parameters.

Now let us express this second order equation of motion as a first order problem in s. Such that $\dot{s}$ contains u_3 (control input) and if s is zero that implies e, $\dot{e}$ and $\ddot{e}$ is zero.

$$s = \dot{e} + \lambda e \tag{2.33}$$
$$\dot{s} = \ddot{e} + \lambda \dot{e} \tag{2.34}$$

where, λ is a positive tunable parameter, which determines how quick is the rate of sliding.

To design u_2 the sliding condition is to be satisfied, which is given by

$$s\dot{s} \leq -k|s| \tag{2.35}$$

Therefore,

$$s(\ddot{e} + \lambda \dot{e}) \leq -k|s| \tag{2.36}$$
$$s(\ddot{\theta} - \ddot{\theta_d} + \lambda \dot{e}) \leq -k|s| \tag{2.37}$$

From the equations of motion we can write

$$s\Big(\dot{\phi}\dot{\psi}(\frac{I_z - I_x}{I_y}) + (\frac{I_p}{I_y})\Omega\dot{\phi} + \frac{u_3}{I_y} - \ddot{\theta_d} + \lambda \dot{e}\Big) \leq -k|s| \tag{2.38}$$

Now the task is to select u_3 such that the above condition is always satisfied. Which can be done by selecting the following formulation for u_3

$$u_3 = -I_y K * sgn(s) - \dot{\phi}\dot{\psi}(I_z - I_x) - I_p \Omega \dot{\phi} + I_y \ddot{\theta_d} - I_y \lambda \dot{e} \tag{2.39}$$

Here, $sgn(s)$ is the function that return sign (negative or positive) of s. However, the project demands implementation of sliding mode control with a boundary layer. Therefore, instead of $sgn(s)$ we can use $sat(\frac{s}{\alpha})$, where $sat()$ is a saturation function and α is boundary layer. Therefore the new u_3 becomes

$$u_3 = -I_y K * sat(\frac{s}{\alpha}) - \dot{\phi}\dot{\psi}(I_z - I_x) - I_p \Omega \dot{\phi} + I_y \ddot{\theta_d} - I_y \lambda \dot{e} \tag{2.40}$$

2.4 Sliding Mode Control For the Yaw (ψ)

Equation of Motion in ψ is give as

$$\ddot{\psi} = \dot{\phi}\dot{\theta}(\frac{I_x - I_y}{I_z}) + \frac{u_4}{I_z} \tag{2.41}$$

This equation of motion is a SISO system and is expressed in the control affine form, therefore to design a sliding mode control let us define error for this system.

$$e = \psi - \psi_d \tag{2.42}$$

$$\dot{e} = \dot{\psi} - \dot{\psi_d} \tag{2.43}$$

$$\ddot{e} = \ddot{\psi} - \ddot{\psi_d} \tag{2.44}$$

Where, ψ is the current yaw angle of the drone, ψ_d is the desired yaw angle of the drone obtained from the desired trajectory, $\dot{\psi}$ is the current angular velocity of the drone around body axis z, $\dot{\psi_d}$ is the desired angular velocity of the drone around body axis z, similarly $\ddot{\psi}$ is the current angular acceleration of the drone around body axis z and $\ddot{\psi_d}$ is the desired angular acceleration of the drone around body axis z.

Now let us express this second order equation of motion as a first order problem in s. Such that $\dot{s}$ contains u_4 (control input) and if s is zero that implies e, $\dot{e}$ and $\ddot{e}$ is zero.

$$s = \dot{e} + \lambda e \tag{2.45}$$

$$\dot{s} = \ddot{e} + \lambda \dot{e} \tag{2.46}$$

where, λ is a positive tunable parameter, which determines how quick is the rate of sliding.

To design u_2 the sliding condition is to be satisfied, which is given by

$$s\dot{s} \leq -k|s| \tag{2.47}$$

Therefore,

$$s(\ddot{e}+\lambda\dot{e}) \leq -k|s| \tag{2.48}$$

$$s(\ddot{\psi}-\ddot{\psi}_d+\lambda\dot{e}) \leq -k|s| \tag{2.49}$$

From the equations of motion we can write

$$s\Big(\dot{\phi}\dot{\theta}(\frac{I_x-I_y}{I_z})+\frac{u_4}{I_z}-\ddot{\psi}_d+\lambda\dot{e}\Big) \leq -k|s| \tag{2.50}$$

Now the task is to select u_4 such that the above condition is always satisfied. Which can be done by selecting the following formulation for u_4

$$u_4 = -I_z k sgn(s) - \dot{\phi}\dot{\theta}(I_x-I_y) - I_z\lambda\dot{e} + I_z\ddot{\psi}_d \tag{2.51}$$

Here, $sgn(s)$ is the function that return sign (negative or positive) of s. However, the project demands implementation of sliding mode control with a boundary layer. Therefore, instead of $sgn(s)$ we can use $sat(\frac{s}{\alpha})$, where $sat()$ is a saturation function and α is boundary layer. Therefore the new u_4 becomes

$$u_4 = -I_z k sat(\frac{s}{\alpha}) - \dot{\phi}\dot{\theta}(I_x-I_y) - I_z\lambda\dot{e} + I_z\ddot{\psi}_d \tag{2.52}$$

It is mentioned in the project that for all four sliding mode controllers, we have to assume that $\psi_d = 0$, $\dot{\phi}_d = \dot{\theta}_d = \dot{\psi}_d = 0$ and $\ddot{\phi}_d = \ddot{\theta}_d = \ddot{\psi}_d = 0$. These entities can be substituted in the above derived control laws for u_1, u_2, u_3 and u_4 to obtain the final control laws.

The tunable parameters for all four controllers (i.e, k, λ and α) can be tuned separately for each controller as per the desired performance of the controllers and will be different for different control variable. Moreover, the estimation parameters K_p and K_d can also be tuned according to the estimation requirements and performance of the controllers.