RBE502 Group Project

Dushyant Patil and Abizer Patanwala

1 Part 1

Desired trajectory for generalized coordinates X, Y, Z)
The quintic polynomial for desired trajectory of the quadrotor are defined in the following form:
$Q_{des} = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5$
This equation is differentiated to find the desired velocity and desired acceleration terms as follows:
$\dot{Q}_{des} = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4$
$\ddot{Q}_{des} = 2a_2 + 6a_3 t + 12a_4 t^2 + 20a_5 t^3$

Solving these equation with the help of initial and final position, velocity and acceleration data for the 5 trajectory equations for 5 given waypoints, we find the desired trajectory of the quadrotor as follows:

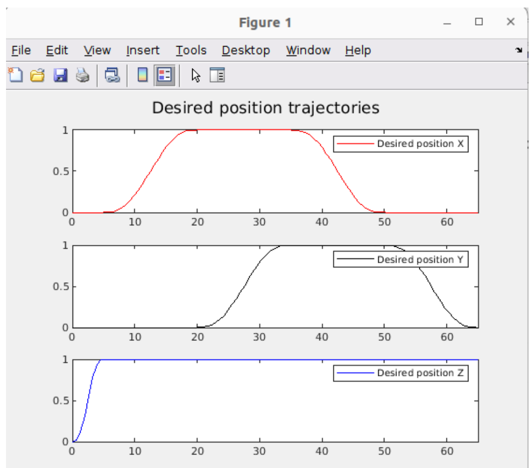

Figure 1: Desired trajectory X, Y, Z for the time span of 65 sec

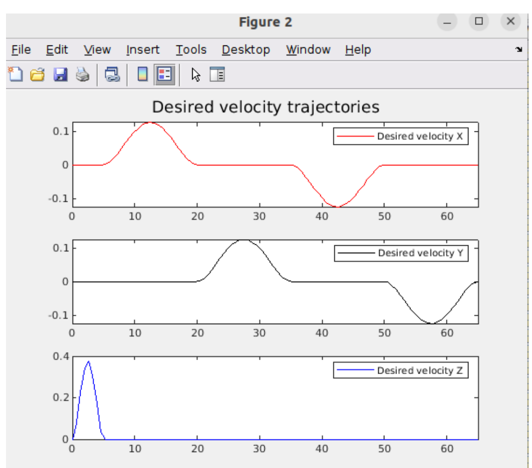

Figure 2: Desired velocity trajectory X, Y, Z for the time span of 65 sec

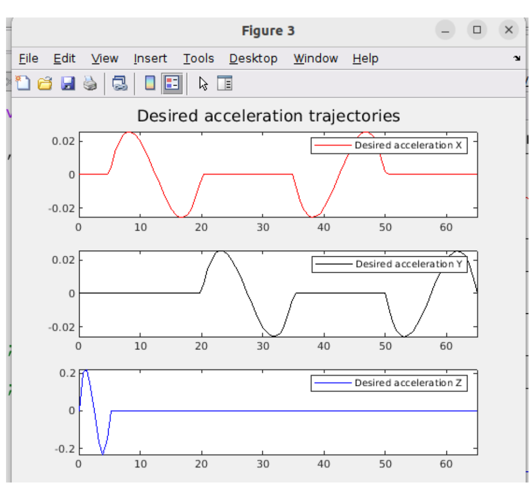

Figure 3: Desired acceleration trajectory X, Y, Z for the time span of 65 sec

2 Part 2

Boundary layer-based sliding mode control Boundary layer control was used to derive all 4 controls - u1, u2, u3, u4. The derivation for all control inputs is given below:
u1:
Select a sliding surface as:

$$\mathrm{s} = \dot{e} + \lambda e$$

Here

$$\mathrm{e} = z - z_{des}$$
$$\dot{e} = \dot{z} - \dot{z}_{des}$$
$$\ddot{e} = \ddot{z} - \ddot{z}_{des}$$
$$\dot{s} = \ddot{e} + \lambda \dot{e}$$

Multiplying s by $\dot{s}$

$$\mathrm{s}\dot{s} = \mathrm{s}\ (\ddot{e} + \lambda \dot{e})$$
$$\mathrm{s}\dot{s} = \mathrm{s}\ (\ddot{z} - \ddot{z}_{des} + \lambda \dot{e})$$

As per given data and quadrotor equation of motion

$$\ddot{z} = cos(\theta)cos(\phi) * u_1/m - g$$
$$\mathrm{s}\dot{s} = \mathrm{s}\ (cos(\theta)cos(\phi) * u_1/m - g - \ddot{z}_{des} + \lambda \dot{e})$$

Let c = $cos(\theta)cos(\phi)/m$. Substituting c in above equation

$$\mathrm{s}\dot{s} = \mathrm{s}^*\mathrm{c}^*\ (u_1 - g/c - \ddot{z}_{des}/c + \lambda \dot{e}/c)$$
But
$$\mathrm{s}\dot{s} \leq .\ \text{-K}\ |s|$$

Thus, we select u_1 such that we cancel out remaining terms in the equation above and add a u_r term to satisfy the identity for sliding mode control.

$$u_1 = g/c + \ddot{z}/c - (\lambda \dot{e})/c + u_r$$
$$u_r = -K sat(s, boundary_1)$$

Where
if ($|s| < boundary_1$)
    $sat(s, boundary_1) = s/boundary_1$
else
    $sat(s, boundary_1) = sign(s)$

Similarly deriving the control inputs for other 3 controls:

u2:
Select a sliding surface as:

$$\mathrm{s} = \dot{e} + \lambda e$$

Here

$$\mathrm{e} = \phi - \phi_{des}$$
$$\phi_{des} = sin^{-1}(-F_y/u_1)$$
Here
$$F_y = m(-k_p(y - y_d) - k_d(\dot{y} - \dot{y_d}) + \ddot{y_d})$$
$$\dot{e} = \dot{\phi} - \dot{\phi}_{des}$$
$$\ddot{e} = \ddot{\phi} - \ddot{\phi}_{des}$$
$$\dot{s} = \ddot{e} + \lambda \dot{e}$$

Multiplying s by $\dot{s}$

$$\mathrm{s}\dot{s} = \mathrm{s}\ (\ddot{e} + \lambda \dot{e})$$
$$\mathrm{s}\dot{s} = \mathrm{s}\ (\ddot{\phi} - \ddot{\phi}_{des} + \lambda \dot{e})$$

As per given data and quadrotor equation of motion

$$\ddot{\phi} = \dot{\theta}\dot{\psi}(I_y - I_z)/I_x - \dot{\theta}\Omega I_p/I_x + u_2/I_x$$
$$\mathrm{s}\dot{s} = s/I_x(\dot{\theta}\dot{\psi}(I_y - I_z) - \dot{\theta}\Omega I_p - \ddot{\phi}_{des} + I_x \lambda \dot{e} + u_2)$$
But
$$\mathrm{s}\dot{s} \leq .\ \text{-K}\ |s|$$

Thus, we select u_2 such that we cancel out remaining terms in the equation above and add a u_r term to satisfy the identity for sliding mode control.

$$u_2 = (-\dot{\theta}\dot{\psi}(I_y - I_z) + \dot{\theta}\Omega I_p - \lambda \dot{e} I_x) + u_r$$
As $\ddot{\phi}_{des} = 0$
$$u_r = -K * I_x * sat(s, boundary_2)$$

Where
if ($|s| < boundary_2$)
    $sat(s, boundary_2) = s/boundary_2$
else
    $sat(s, boundary_2) = sign(s)$

u3:
Select a sliding surface as:

$$\text{s} = \dot{e} + \lambda e$$

Here

$$\text{e} = \theta - \theta_{des}$$
$$\theta_{des} = sin^{-1}(F_x/u_1)$$
$$\text{Here}$$
$$F_y = m(-k_p(x - x_d) - k_d(\dot{x} - \dot{x_d}) + \ddot{x_d})$$
$$\dot{e} = \dot{\theta} - \dot{\theta}_{des}$$
$$\ddot{e} = \ddot{\theta} - \ddot{\theta}_{des}$$
$$\dot{s} = \ddot{e} + \lambda\dot{e}$$

Multiplying s by $\dot{s}$

$$\text{s}\dot{s} = \text{s } (\ddot{e} + \lambda\dot{e})$$
$$\text{s}\dot{s} = \text{s } (\ddot{\theta} - \ddot{\theta}_{des} + \lambda\dot{e})$$

As per given data and quadrotor equation of motion

$$\ddot{\theta} = \dot{\phi}\dot{\psi}(I_z - I_x)/I_y + \dot{\phi}\Omega I_p/I_y + u_3/I_y$$
$$\text{s}\dot{s} = s/I_y(\dot{\phi}\dot{\psi}(I_z - I_x) + \dot{\phi}\Omega I_p - \ddot{\theta}_{des} + I_y\lambda\dot{e} + u_3)$$
$$\text{But}$$
$$\text{s}\dot{s} \leq . \text{ -K } |s|$$

Thus, we select u_3 such that we cancel out remaining terms in the equation above and add a u_r term to satisfy the identity for sliding mode control.

$$u_3 = (-\dot{\phi}\dot{\psi}(I_z - I_x) - \dot{\theta}\Omega I_p - \lambda\dot{e}I_y) + u_r$$
$$\text{As } \ddot{\theta}_{des} = 0$$
$$u_r = -K * I_y * sat(s, boundary_3)$$

Where
if ($|s| < boundary_3$)
    $sat(s, boundary_3) = s/boundary_3$
else
    $sat(s, boundary_3) = sign(s)$

u4:
Select a sliding surface as:

$$\text{s} = \dot{e} + \lambda e$$

Here

$$\text{e} = \psi - \psi_{des}$$
$$\text{But } \psi_{des} = 0 \text{ as per given data}$$
$$\dot{e} = \dot{\psi}$$
$$\ddot{e} = \ddot{\psi}$$
$$\dot{s} = \ddot{e} + \lambda\dot{e}$$

Multiplying s by $\dot{s}$

$$\text{s}\dot{s} = \text{s } (\ddot{e} + \lambda\dot{e})$$
$$\text{s}\dot{s} = \text{s } (\ddot{\psi} + \lambda\dot{e})$$

As per given data and quadrotor equation of motion

$$\ddot{\psi} = \dot{\phi}\dot{\theta}(I_x - I_y)/I_z + u_4/I_z$$
$$\text{s}\dot{s} = s/I_z(\dot{\phi}\dot{\theta}(I_x - I_y) + I_z\lambda\dot{e} + u_4)$$
$$\text{But}$$
$$\text{s}\dot{s} \leq . \text{ -K } |s|$$

Thus, we select u_3 such that we cancel out remaining terms in the equation above and add a u_r term to satisfy the identity for sliding mode control.

$$u_4 = (-\dot{\phi}\dot{\theta}(I_x - I_y) - \lambda\dot{e}I_y) + u_r$$
$$\text{As } \ddot{\theta}_{des} = 0$$
$$u_r = -K * I_z * sat(s, boundary_3)$$

Where
if ($|s| < boundary_3$)
    $sat(s, boundary_3) = s/boundary_3$
else
    $sat(s, boundary_3) = sign(s)$

3 Part 3

ROS node implementation
The ROS node implementation was done using python script provided in the assignment. The following script were added to the initial python script:
1. Parameter initialization
2. Desired trajectory calculation
3. Sliding Mode control implementation
4. Trajectory Visualization
We explain below the details of functions mentioned above:
1.Parameter initialization The parameters initialization was done in the init method for the Quadrotor class. We added quintic trajectory coeffficient, system inertia properties and global u_temp variables to store the control input values.

2. Desired trajectory calculation The desired trajectory was calculated using the methodology explained in the part 'a' of the report.

3. sliding mode control The sliding mode control with boudary layer was implemented using the control laws derived in part b of this report. The K and λ gains were tuned so that the generalized coordinates tracked and converged with desired trajectories as optimally as possible. The control inputs were used to derive the propeller velocities using the allocation matrix. Contraints were applied to maintain the angular velocities in the given bounds of $0 < \omega < 2618$

4. Trajectory Visualization The controlled trajectory was plotted in 3D and was compared against the desired trajectory to qualitatively evaluate the performance of the designed controller.
Parameter Tuning
To achieve the sliding mode control and track the desired trajectory optimally, we need to tune the K and λ gains as follows:
K_1 -
As per the equations in section b of the report, k_1 is the gain used in control of the altitude 'z'. The increase in k_1 causes faster convergence to sliding surface and also causes increase in magnitude of acceleration in X, Y, and Z direction. The final k_1 values is set to 15

λ_1 -
As per the equations in section b of the report, increase in λ_1 causes faster reduction in error terms and causes faster conversion to the desired trajectories. The final λ_1 value is set to 10

K_p, K_d for F_x and F_y -
As per the equations of motion given in the assignment, the K_p and K_d gains are used to find F_x and F_y which are further used to find the desired θ and ϕ angles of the quadrotor. The gain values are selected such that the $F_x and F_y$ values are less than u1 so that the $sin^{-1}(F_x/u1)$ and $sin^{-1}(-F_y/u1)$ do not give any error due to invalid values. Also if $(F_x/u1)$ and $-F_y/u1)$ values go beyond 1 or -1 we cap it to 1 or -1 so that $sin^{-1}(F_x/u1)$ and $sin^{-1}(-F_y/u1)$ do not give any error. The final gains for F_x and F_y are set as follows:
$k_{px} = 120; k_{dx} = 5;$
$k_{py} = 80; k_{dy} = 5;$

k_2, k_3 and λ_2, λ_2 -
The gains k_2 and k_3 are tuned for a faster approach to the sliding surface s and λ_2 and λ_3 are tuned for faster convergence to the desired values of θ and ϕ. The final values for k_2 and k_3 are set to 100. The quadrotor is very sensitive to these gain values and if k_2 and k_3 are set low, then the quadrotor becomes unstable and falls down/ crashes mid-flight. Similarly λ_2 and λ_3 are set to 15 and 1 respectively.

k_4 and λ_4 -
The control input u_4 is used to control the rotation of the robot around the z axis (ψ). For the simulation performed during this assignment, the ψ does not change much and thus the gain values for u_4 are as follows:
$k_4 = 20, \lambda_4 = 10.$
The values of K2 and K3 are high because in control law of ϕ and θ we have a term which is the sum of angular velocities of the quadrotor and hence we require higher K2 and K3 values to cancel that.
The simulation is performed using these gains values and a screen recording video file is attached along with the report for reference. The traced trajectory is stored in a .pkl file using pickle module. This saved data is later used to visualize the trajectory which will be explained in the next section.

4 Part 4

The part 4 is used to visualize the traced trajectory and qualitatively compare the traced vs desired trajectory. The plotting is done using the saved data stored in a .pkl file created in main script using *save_data* function. The 3D plot is as shown below:

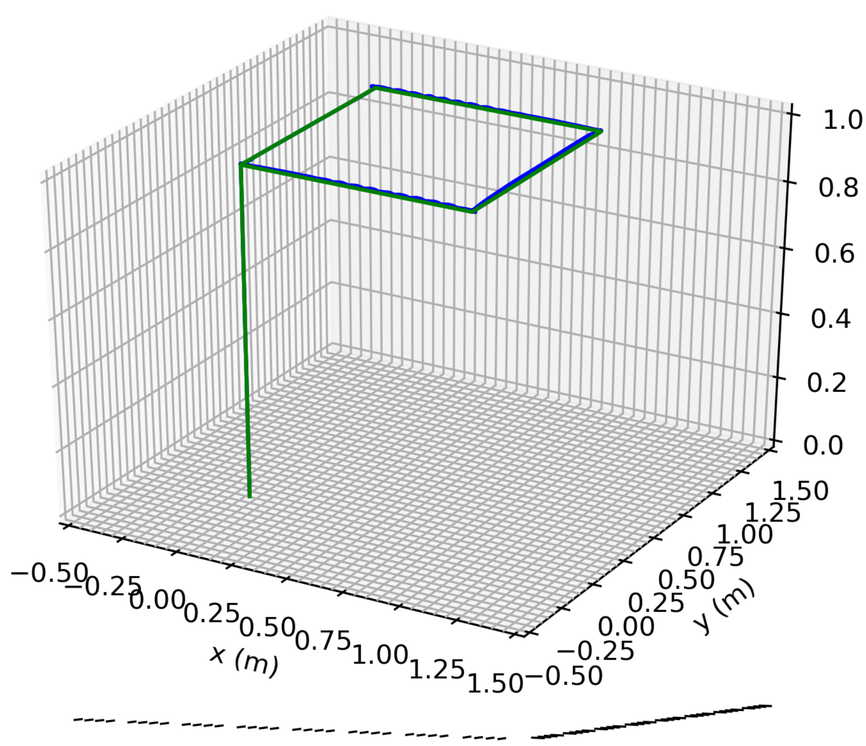

Figure 4: Desired (Green) vs Traced (Blue) trajectory

As can be seen in the figure above, the quadrotor travels along the desired trajectory very accurately and the robust control using boundary layer sliding mode control is implemented successfully.