Aggressive Maneuvers for Quadrotors - Minimum Snap Trajectory + Nonlinear controller - Gazebo-ROS Simulation

1st Tofigh HOSSEINI
Department of Mechatronics
Erciyes University

***Abstract*—This project focuses on the development of a quadrotor simulation capable of executing aggressive maneuvers. A simulation environment was created using Gazebo integrated with ROS (Robot Operating System) nodes, providing a realistic platform for testing and evaluation. The key components of this project include trajectory planning and tracking, as well as the development of a nonlinear controller for the quadrotor. For trajectory planning, a "Minimum Snap Trajectory" planner was developed. This planner generates aggressive trajectories by optimizing the 4th derivative of position, known as snap. These trajectories are designed to enable the quadrotor to perform high-acceleration maneuvers effectively. To track these aggressive trajectories, a nonlinear controller was developed. Leveraging the concept of differential flatness, the planned trajectory is mapped to quadrotor states, allowing for precise tracking of the desired trajectory. The controller ensures that the quadrotor follows the planned trajectory while considering its dynamic constraints and environmental conditions. This report provides a detailed description of the design, implementation, and evaluation of the quadrotor simulation and control system. Simulation results demonstrate the effectiveness of the developed system in enabling the quadrotor to execute aggressive maneuvers accurately and efficiently.**

***Index Terms*—Aggressive Maneuvers, Minimum Snap Trajectory, Quadrotor Control, Nonlinear Control, ROS**

I. Introduction

In the past decade, there has been a significant increase in the development and utilization of multirotor systems. These aerial vehicles have found applications in various fields such as surveillance, inspection, and aerial photography, among others. With this rise in interest, there has been a surge in research focusing on trajectory generation and control strategies for these vehicles. Among multirotor configurations, the quadrotor, equipped with four rotors, has emerged as one of the most popular choices due to its simplicity, agility, and versatility.

Numerous control strategies have been proposed and implemented for quadrotor systems in the existing literature. These include Casaded PID controllers, Linear Quadratic Regulators (LQR), and Model Predictive Controllers (MPC). However, many of these controllers rely on small angle approximations, limiting their applicability to scenarios involving aggressive maneuvers and large attitude deviations. In this project, we aim to address this limitation by developing a nonlinear controller that does not make these small angle approximations, allowing for more accurate control over a wider range of maneuvers.

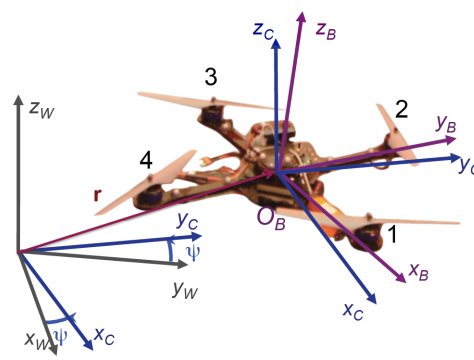

Fig. 1: World frame $\mathcal{W}$, Flat frame $\mathcal{C}$ and Body frame $\mathcal{B}$ [1]

Furthermore, trajectory planning plays a crucial role in enabling quadrotors to perform complex maneuvers. To achieve aggressive flights, we will first derive the equations of flatness for quadrotors. Leveraging these equations, we will develop a Minimum Snap Trajectory planner capable of generating aggressive trajectories that optimize the 4th derivative of position, known as snap. These trajectories will enable the quadrotor to perform high-acceleration maneuvers efficiently and accurately.

This project aims to contribute to the advancement of quadrotor control and trajectory planning techniques, allowing for more agile and precise maneuvering in various applications. While this work is not unique, it serves as an implementation of the methods proposed in [1], providing a practical demonstration of the effectiveness of the developed control and trajectory planning strategies. Through this implementation, we seek to validate the proposed approaches and explore their performance in realistic simulation environments.

II. Quadrotor Model

In this section, we will present the quadrotor model used in this project. The quadrotor is represented using two coordinate systems: the world frame or inertial frame $\mathcal{W} = [\mathbf{x}_{\mathcal{W}}, \mathbf{y}_{\mathcal{W}}, \mathbf{z}_{\mathcal{W}}]$, which is attached to the origin and remains stationary, and the body frame $\mathcal{B} = [\mathbf{x}_{\mathcal{B}}, \mathbf{y}_{\mathcal{B}}, \mathbf{z}_{\mathcal{B}}]$, which is attached to the quadrotor's center of mass.

We will use the Z-Y-X Euler combination to represent the roll ϕ, pitch θ, and yaw ψ. We will also interchangeably use quaternions to represent the orientation of the quadrotor where $\mathbf{q} = [q_w, q_x, q_y, q_z]^T$. The rotation from the world frame $\mathcal{W}$ to the body frame $\mathcal{B}$ can be written as:

$$^{\mathcal{W}}R_{\mathcal{B}} = {}^{\mathcal{W}}R_{\mathcal{C}}\,{}^{\mathcal{C}}R_{\mathcal{B}}$$

where $\mathcal{C}$ is the flat frame.

The angular velocity of frame $\mathcal{B}$ with respect to $\mathcal{W}$, represented in $\mathcal{B}$ with three components p, q, and r, is given by:

$$\omega_{\mathcal{BW}} = p\mathbf{x}_{\mathcal{B}} + q\mathbf{y}_{\mathcal{B}} + r\mathbf{z}_{\mathcal{B}}$$

The state of the quadrotor can be represented by:

- Position of the center of mass: $\mathbf{r} = [x, y, z]^T$
- Orientation of the body frame $\mathcal{B}$, parameterized by quaternions: $\mathbf{q} = [q_w, q_x, q_y, q_z]^T$
- Velocity of the center of mass: $\mathbf{v} = [\dot{x}, \dot{y}, \dot{z}]^T$
- Angular velocity of frame $\mathcal{B}$ with respect to frame $\mathcal{W}$: $\omega_{\mathcal{BW}} = [p, q, r]^T$

So the overall state vector is:

$$\mathbf{X} = [x, y, z, q_w, q_x, q_y, q_z, \dot{x}, \dot{y}, \dot{z}, p, q, r]^T \tag{1}$$

The quadrotor is an underactuated system with 6 degrees of freedom, of which only 4 are actuated. These actuated degrees of freedom are:

- Thrust T along the $\mathbf{z}_{\mathcal{B}}$ axis,
- Moment M_x around the $\mathbf{x}_{\mathcal{B}}$ axis,
- Moment M_y around the $\mathbf{y}_{\mathcal{B}}$ axis, and
- Moment M_z around the $\mathbf{z}_{\mathcal{B}}$ axis.

Therefore, the input vector of the quadrotor can be represented as:

$$\mathbf{u} = [T, M_x, M_y, M_z]^T = [u_1, u_2, u_3, u_4]^T \tag{2}$$

Assuming that the motor dynamics are much faster than the quadrotor dynamics, and any motor angular velocity can be instantly achieved on command, the input vector 2 equation can be mapped to motor velocities through what is known in the literature as the motor mixing equation:

$$\mathbf{u} = \begin{bmatrix} k_F & k_F & k_F & k_F \\ 0 & k_F L & 0 & -k_F L \\ -k_F L & 0 & -k_F L & 0 \\ k_M & -k_M & k_M & -k_M \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{3}$$

where k_F is the thrust coefficient, k_M is the torque coefficient, and L is the distance from the center of the quadrotor to each motor. The motor velocities ω_i^2, for $i = 1, 2, 3, 4$, represent the squared angular velocities of each motor.

The Newtonian's equation of acceleration of center of mass for the quadrotor can be written as:

$$m\ddot{r} = -mg\mathbf{z}_{\mathcal{W}} + u_1\mathbf{z}_{\mathcal{B}} \tag{4}$$

The angular acceleration of quadrotor can be determined by Euler's equation:

$$\dot{\omega}_{\mathcal{BW}} = \mathcal{I}^{-1}\left[-\omega_{\mathcal{BW}} \times \mathcal{I}\omega_{\mathcal{BW}} + \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix}\right] \tag{5}$$

where m and $\mathcal{I}$ are mass and inertia tensor of the quadrotor.

III. Differential Flatness

The quadrotor is known to be differentially flat, meaning the states and inputs of the system can be written as algebraic functions of four carefully selected flat outputs and their derivatives. For quadrotors, the choice of flat outputs is:

$$\sigma = [x, y, z, \psi]^T \tag{6}$$

where x, y, z, and ψ represent the position and yaw angle of the quadrotor, respectively. The position, velocity, and acceleration of the center of mass are simply the first three terms of σ, $\dot{\sigma}$, and $\ddot{\sigma}$, respectively.

Since a quadrotor can only linearly accelerate along its body $\mathbf{z}_{\mathcal{B}}$ axis, we have:

$$\mathbf{z}_{\mathcal{B}} = \frac{\mathbf{a}}{||\mathbf{a}||}, \quad \mathbf{a} = [\ddot{x}, \ddot{y}, \ddot{z} + g]^T \tag{7}$$

We can show that the rotation of the quadrotor is a function of flat outputs by defining:

$$\mathbf{x}_{\mathcal{C}} = [cos\psi, sin\psi, 0]^T$$

We can determine $\mathbf{y}_{\mathcal{B}}$ and $\mathbf{x}_{\mathcal{B}}$ as follows:

$$\mathbf{y}_{\mathcal{B}} = \frac{\mathbf{z}_{\mathcal{B}} \times \mathbf{x}_{\mathcal{C}}}{||\mathbf{z}_{\mathcal{B}} \times \mathbf{x}_{\mathcal{C}}||}, \quad \mathbf{x}_{\mathcal{B}} = \mathbf{y}_{\mathcal{B}} \times \mathbf{z}_{\mathcal{B}}$$

Since the rotation from frame $\mathcal{W}$ to frame $\mathcal{B}$ is represented by the matrix of unit vectors of $\mathcal{B}$, we have:

$$^{\mathcal{W}}R_{\mathcal{B}} = \begin{bmatrix} \mathbf{x}_{\mathcal{B}} & \mathbf{y}_{\mathcal{B}} & \mathbf{z}_{\mathcal{B}} \end{bmatrix} \tag{8}$$

To show that angular velocity is an algebraic function of the flat outputs, we take the derivatives of Equation 4:

$$m\dot{a} = \dot{u_1}\mathbf{z}_{\mathcal{B}} + \omega_{\mathcal{WB}} \times u_1\mathbf{z}_{\mathcal{B}}$$

The projection of this expression on $\mathbf{z}_{\mathcal{B}}$ defines the vector $\mathbf{h}_\omega$ as:

$$\mathbf{h}_\omega = \omega_{\mathcal{WB}} \times \mathbf{z}_{\mathcal{B}} = \frac{m}{u_1}(\dot{\mathbf{a}} - (\mathbf{z}_{\mathcal{B}} \cdot \dot{\mathbf{a}})\mathbf{z}_{\mathcal{B}})$$

$\mathbf{h}_\omega$ is the projection of $\frac{m}{u_1}\dot{\mathbf{a}}$ onto the $\mathbf{x}_{\mathcal{B}} - \mathbf{y}_{\mathcal{B}}$ plane. The body components of angular velocity are then found by:

$$p = -\mathbf{h}_\omega \cdot \mathbf{y}_{\mathcal{B}}, \quad q = \mathbf{h}_\omega \cdot \mathbf{x}_{\mathcal{B}}$$

The third component is given by:

$$r = \dot{\psi}\mathbf{z}_{\mathcal{W}} \cdot \mathbf{z}_{\mathcal{B}}$$

Now that the entire state can be found using the flat outputs, it is time to develop a controller to track the trajectory.

IV. Nonlinear Controller

I now present a nonlinear controller to track a trajectory parameterized with a set of flat outputs $\sigma_T(t) = [\mathbf{r}_T(t)^T, \psi_T(t)]^T$. We can define the position and velocity errors as:

$$\mathbf{e}_p = \mathbf{r} - \mathbf{r}_T, \quad \mathbf{e}_v = \dot{\mathbf{r}} - \dot{\mathbf{r}}_T$$

Defining K_p and K_v as positive definite diagonal gain matrices, we can compute the desired force vector and the desired body frame $\mathbf{z}_{\mathcal{B}}$ axis:

$$\mathbf{F}_{des} = -K_p\mathbf{e}_p - K_v\mathbf{e}_v + mg\mathbf{z}_{\mathcal{W}} + m\ddot{\mathbf{r}}_T \tag{9}$$

By projecting the desired force vector $\mathbf{F}_{des}$ onto the quadrotor body frame axis $\mathbf{z}_{\mathcal{B}}$, the desired thrust can be calculated:

$$u_1 = \mathbf{F}_{des} \cdot \mathbf{z}_{\mathcal{B}} \tag{10}$$

The other three inputs to the system must be calculated by considering the rotation. The desired direction of $\mathbf{z}_{\mathcal{B}}$ is calculated by normalizing the desired force vector $\mathbf{F}_{des}$:

$$\mathbf{z}_{\mathcal{B},des} = \frac{\mathbf{F}_{des}}{||\mathbf{F}_{des}||}$$

The other two unit vectors of the desired body frame can be calculated using the equations derived from differential flatness:

$$\mathbf{x}_{\mathcal{C},des} = \begin{bmatrix}\cos\psi_T & \sin\psi_T & 0\end{bmatrix}^T$$

Therefore, the desired $\mathbf{y}_{\mathcal{B},des}$ and $\mathbf{x}_{\mathcal{B},des}$ unit vectors are:

$$\mathbf{y}_{\mathcal{B},des} = \frac{\mathbf{z}_{\mathcal{B},des} \times \mathbf{x}_{\mathcal{C},des}}{||\mathbf{z}_{\mathcal{B},des} \times \mathbf{x}_{\mathcal{C},des}||}, \quad \mathbf{x}_{\mathcal{B},des} = \mathbf{y}_{\mathcal{B},des} \times \mathbf{z}_{\mathcal{B},des}$$

Therefore, the desired rotation matrix is:

$$R_{des} = \begin{bmatrix}\mathbf{x}_{\mathcal{B},des} & \mathbf{y}_{\mathcal{B},des} & \mathbf{z}_{\mathcal{B},des}\end{bmatrix} \tag{11}$$

The rotation error can be defined as:

$$R_{\text{error}} = {}^{\mathcal{W}}R_{\mathcal{B}}^T R_{des}$$

Parameterizing the R_{error} with a vector of unit quaternions $\mathbf{q}_e$, the rotation error can be written as:

$$\mathbf{e}_r = \text{sign}(\mathbf{q}_{w,e}) \begin{bmatrix}\mathbf{q}_{x,e} \\ \mathbf{q}_{y,e} \\ \mathbf{q}_{z,e}\end{bmatrix} \tag{12}$$

To calculate the desired angular velocities, we can use the equations of differential flatness. The body components of the desired angular velocity are then found by:

$$p_{\text{des}} = -\mathbf{h}_\omega \cdot \mathbf{y}_{\mathcal{B},\text{des}}, \quad q_{\text{des}} = \mathbf{h}_\omega \cdot \mathbf{x}_{\mathcal{B},\text{des}}$$

The third component is given by:

$$r_{\text{des}} = \dot{\psi}\mathbf{z}_{\mathcal{W}} \cdot \mathbf{z}_{\mathcal{B},\text{des}}$$

Thus, the desired angular velocity vector and angular velocity error are:

$$\omega_{\mathcal{BW},\text{des}} = \begin{bmatrix}p_{\text{des}} & q_{\text{des}} & r_{\text{des}}\end{bmatrix}^T \quad \mathbf{e}_\omega = \omega_{\mathcal{BW}} - \omega_{\mathcal{BW},\text{des}} \tag{13}$$

Now, the desired moments and the remaining three inputs of the quadrotors with the inclusion of feedback terms canceling $\omega \times \mathcal{I}\omega$ are computed as follows:

$$\begin{bmatrix}u_2 & u_3 & u_4\end{bmatrix}^T = \omega_{\mathcal{BW}} \times \mathcal{I}\omega_{\mathcal{BW}} - \mathcal{I}(-K_r\mathbf{e}_r - K_\omega\mathbf{e}_\omega) \tag{14}$$

where K_r and K_ω are again diagonal weight matrices.

V. Minimum Snap Trajectory

We define a *keyframe* as a position in space along with a yaw angle. To have a smooth trajectory passing through multiple *keyframes*, we define piecewise polynomial functions with a degree of at least 5 (remember that snap is the 4th derivative of position) for each of the two keyframes which the quadrotor must fly through. Therefore, the overall trajectory with $m+1$ keyframes can be represented by m polynomials. Take P as function with m polynomials, so the minimum snap trajectory formulation is:

$$P_T(t) = \begin{cases} \sum_{i=0}^n {}^1a_i t^i & t_0 \le t < t_1 \\ \sum_{i=0}^n {}^2a_i t^i & t_1 \le t < t_2 \\ \vdots \\ \sum_{i=0}^n {}^m a_i t^i & t_{m-1} \le t < t_m \end{cases} \tag{15}$$

To minimize the snap, the following optimization problem should be solved:

$$\min \int_{t_0}^{t_m} \left\| \frac{d^4 P_T}{dt^4} \right\|^2 dt \tag{16}$$

subject to the following constraints:

$$\begin{aligned} &P_T(t_i) = P_i, \quad i = 0, \ldots, m \\ &\left.\frac{d^k P_T}{dt^k}\right|_{t=0} = 0, \quad k = 1, \ldots, 4 \\ &\left.\frac{d^k P_T}{dt^k}\right|_{t=t_m} = 0, \quad k = 1, \ldots, 4 \end{aligned}$$

By writing all the polynomial coefficients in a single vector form, we can represent them as follows:

$$\mathbf{p} = [{}^1a_0, {}^1a_1, \ldots, {}^2a_0, {}^2a_1, \ldots, {}^m a_0, \ldots, {}^m a_n]^T \tag{17}$$

This optimization problem can be converted to a Quadratic Program (QP) form, which can be written as follows with two equality constraints:

$$\min \mathbf{p}^T Q \mathbf{p} \tag{18}$$

subject to:

$$\begin{aligned} A_0\mathbf{p} &= \mathbf{b}_0 \\ A_\tau\mathbf{p} &= \mathbf{b}_\tau \end{aligned}$$

where Q is the Hessian matrix and should be computed by [2]:

$$Q_r^{il} = \begin{cases} 2\left(\prod_{m=0}^{r-1}(i-m)(l-m)\right)\frac{\tau^{i+1-2r+1}}{i+l-2r+1} & \text{if } i \ge r \wedge l \ge r \\ 0 & \text{if } i < r \vee l < r \end{cases}$$

where $\tau = t_{i+1} - t_i$ is the duration between the keyframes.

References

[1] D. Mellinger and V. Kumar, ”Minimum snap trajectory generation and control for quadrotors,” 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 2011, pp. 2520-2525, doi: 10.1109/ICRA.2011.5980409.

[2] Bry, Adam, Charles Richter, Abraham Bachrach, and Nicholas Roy. “Aggressive Flight of Fixed-Wing and Quadrotor Aircraft in Dense Indoor Environments.” The International Journal of Robotics Research, 34, no. 7 (June 2015): 969–1002.