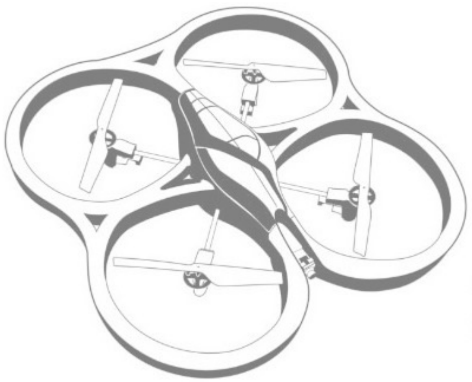
UAV Workshop

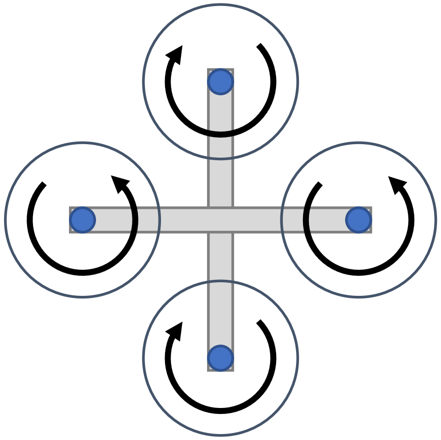
Basics of Quadrotor

Prof. Venki Muthukumar, Ph.D.

Quadrotor: Flying Principle



http://blog.parrot.com/2010/02/10/macworld-2010-fly-the-parrot-ardrone/



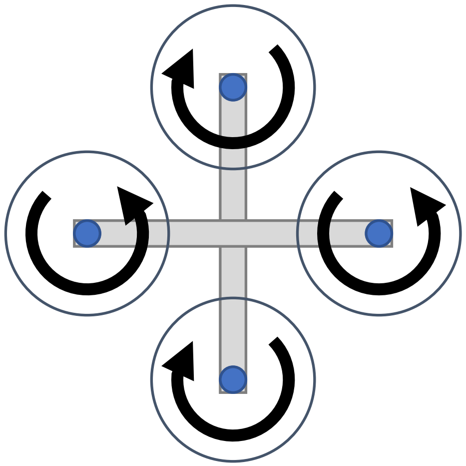
UAV Workshop

Keep position:

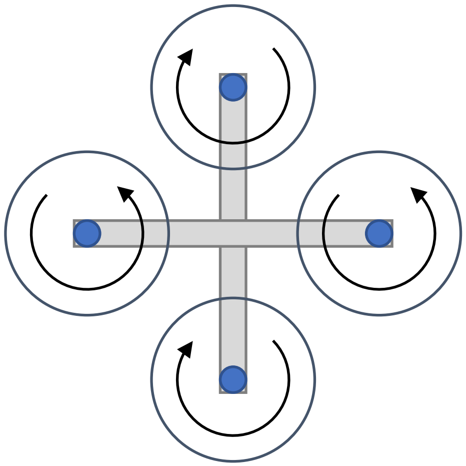
- Thrust compensates for earth gravity
- Torques of all four rotors sum to zero

Quadrotor: Basic Motions

UAV Workshop

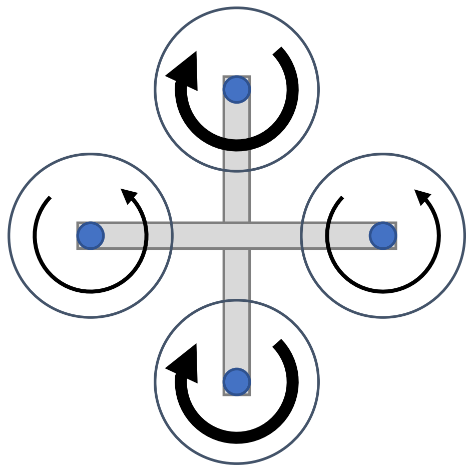


Ascend

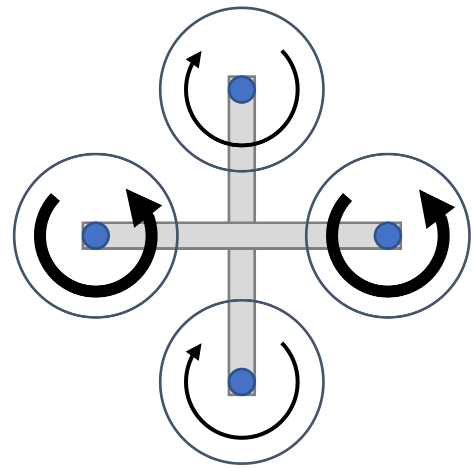


Descend

Quadrotor: Basic Motions

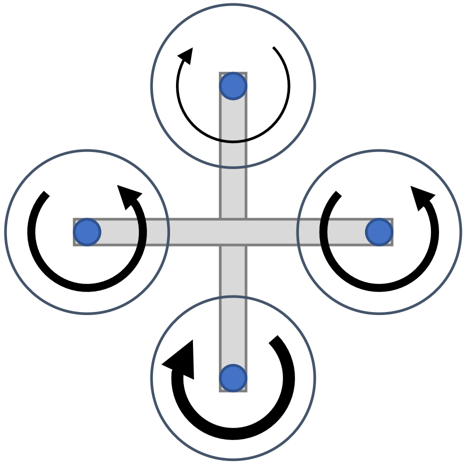


Turn Left

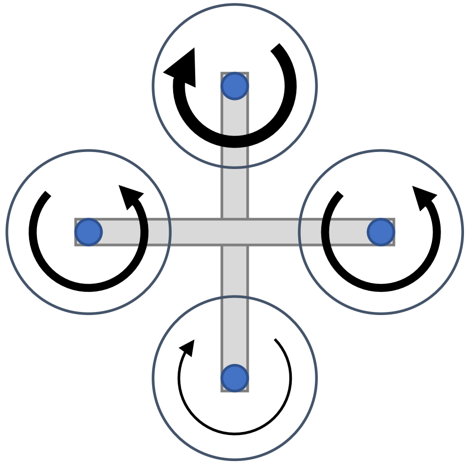


Turn Right

Quadrotor: Basic Motions



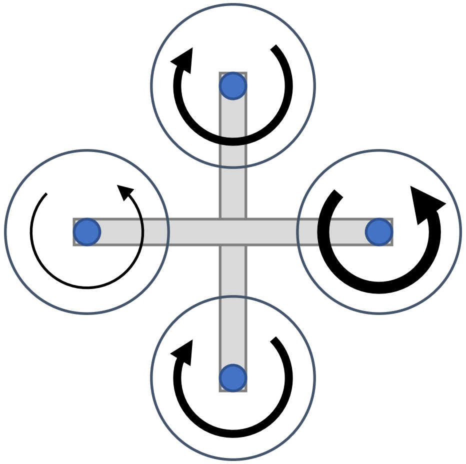
Move forward



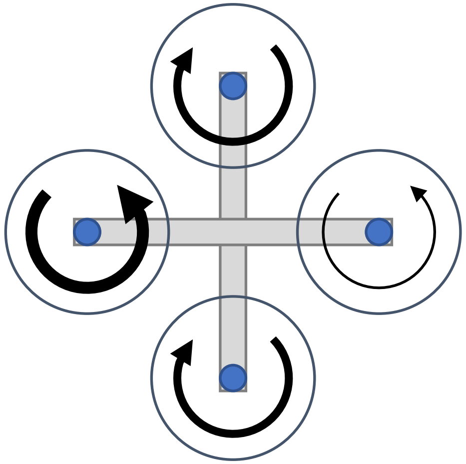
Move backwards

Quadrotor: Basic Motions

UAV Workshop



Move left



Move right

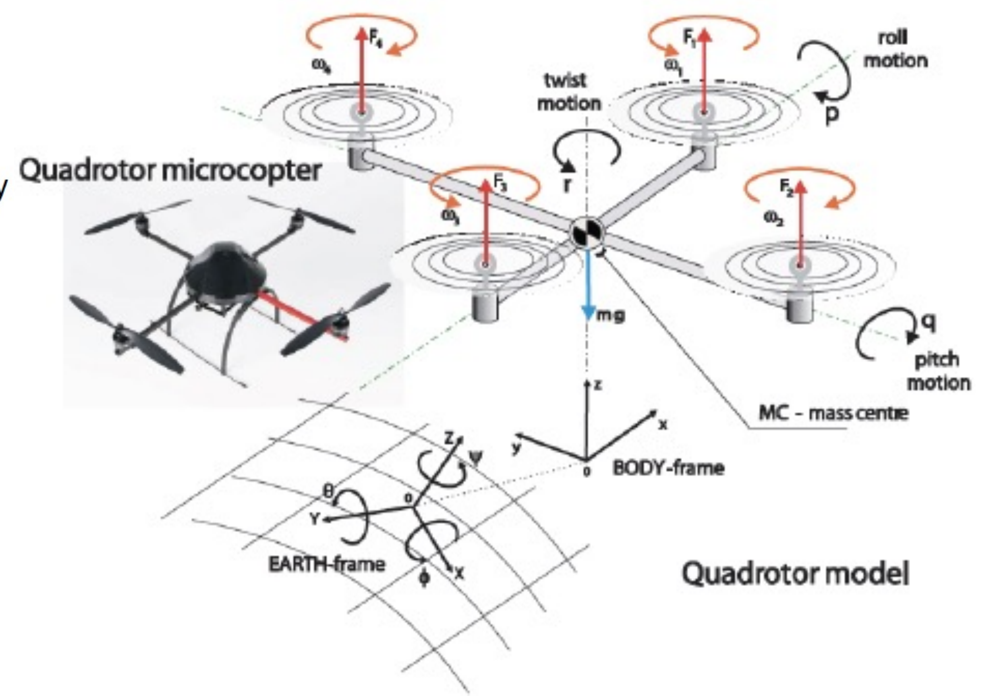
Quadrotor Model

Newton-Euler equations:

total force, mass, linear acceleration, linear velocity

$$\begin{bmatrix}\mathbf{F}\\ \tau\end{bmatrix} = \begin{bmatrix} m\mathbb{1}_3 & \mathbb{0}_3 \\ \mathbb{0}_3 & \mathbf{I}_3\end{bmatrix}\begin{bmatrix}\mathbf{a}\\ \alpha\end{bmatrix} + \begin{bmatrix}\omega\times m\mathbf{v}\\ \omega\times \mathbf{I}_3\omega\end{bmatrix}$$

total torque, moment of inertia, angular acceleration, angular velocity



Quadrotor model

UAV Workshop

Roll ϕ **, Pitch** θ **, Yaw** ψ

Earth/inertia frame - starting position of the quadrotor is the origin of the global frame or the inertia frame (Fi)

Fv is the vehicle frame - inertia frame, Fi, linear shifted to the centre of gravity (COG) for the quadrotor.

Fb is the body frame when Fv is rotated by yaw, pitch and roll angles along Zv, Yv and Xv axis.

$$R_{vb}{}^{v} = \begin{bmatrix} C_\theta C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ C_\theta S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi \\ -S_\theta & S_\phi C_\theta & C_\phi C_\theta \end{bmatrix}$$

Rotation Matrix

Forces

Positions in Fi : p_n, p_e, h

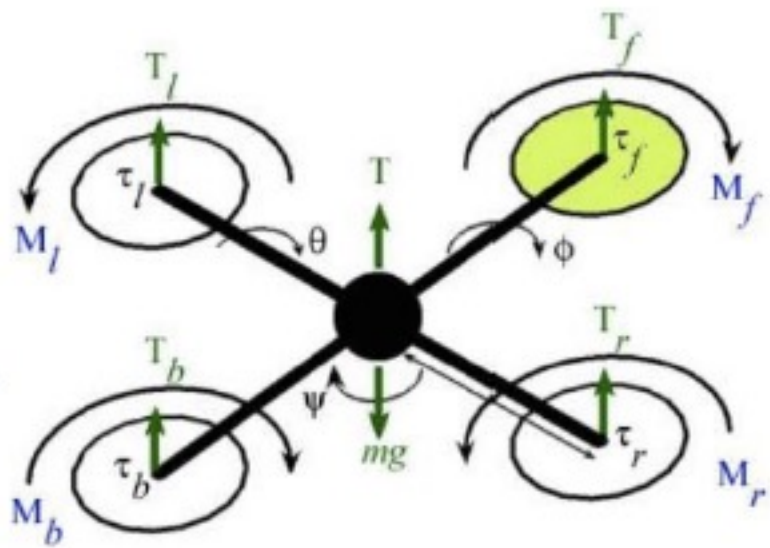
Velocities in Fb: u, v, w

Angular velocities in Fb: p, q, r

Euler angles:

- Yaw angle in Fv: ψ
- Pitch angle in Fv1: ϑ
- Roll angle in Fv2: ϕ

Total thrust in F_b: $T = T_f + T_b + T_l + T_r$



f_x , f_y , f_z are total forces acting on the body frame, F_b

Kinematic Equations of Motion

Six of the 12 state equations for the MAV come from the kinematic equations relating positions and velocities:

$$\begin{pmatrix} \dot{p}_n \\ \dot{p}_e \\ \dot{p}_d \end{pmatrix} = \begin{pmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix}$$

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix}$$

The remaining six equations will come from applying Newton's 2nd law to the translational and rotational motion of the aircraft

Dynamics

UAV Workshop

Translational Dynamics

$$\begin{pmatrix}\dot{u}\\ \dot{v}\\ \dot{w}\end{pmatrix} = -\begin{pmatrix}p\\ q\\ r\end{pmatrix} \times \begin{pmatrix}u\\ v\\ w\end{pmatrix} + \frac{1}{\mathsf{m}}\begin{pmatrix}f_x\\ f_y\\ f_z\end{pmatrix} = \begin{pmatrix}rv-qw\\ pw-ru\\ qu-pv\end{pmatrix} + \frac{1}{\mathsf{m}}\begin{pmatrix}f_x\\ f_y\\ f_z\end{pmatrix}$$

Rotational Dynamics

$$\begin{aligned}\begin{pmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{pmatrix} &= \begin{pmatrix}\frac{J_z}{\Gamma} & 0 & \frac{J_{xz}}{\Gamma}\\ 0 & \frac{1}{J_y} & 0\\ \frac{J_{xz}}{\Gamma} & 0 & \frac{J_x}{\Gamma}\end{pmatrix}\left[\begin{pmatrix}0 & r & -q\\ -r & 0 & p\\ q & -p & 0\end{pmatrix}\begin{pmatrix}J_x & 0 & -J_{xz}\\ 0 & J_y & 0\\ -J_{xz} & 0 & J_z\end{pmatrix}\begin{pmatrix}p\\ q\\ r\end{pmatrix} + \begin{pmatrix}l\\ m\\ n\end{pmatrix}\right]\\
&= \begin{pmatrix}\frac{J_z}{\Gamma} & 0 & \frac{J_{xz}}{\Gamma}\\ 0 & \frac{1}{J_y} & 0\\ \frac{J_{xz}}{\Gamma} & 0 & \frac{J_x}{\Gamma}\end{pmatrix}\left[\begin{pmatrix}J_{xz}pq + (J_y - J_z)qr\\ J_{xz}(r^2 - p^2) + (J_z - J_x)pr\\ (J_x - J_y)pq - J_{xz}qr\end{pmatrix} + \begin{pmatrix}l\\ m\\ n\end{pmatrix}\right]\\
&= \begin{pmatrix}\Gamma_1 pq - \Gamma_2 qr + \Gamma_3 l + \Gamma_4 n\\ \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + \frac{1}{J_y} m\\ \Gamma_7 pq - \Gamma_1 qr + \Gamma_4 l + \Gamma_8 n\end{pmatrix}\end{aligned}$$

where Γ's are functions of moments and products of inertia

Summary of Equation

$$\begin{bmatrix} \dot{p}_n \\ \dot{p}_e \\ -\dot{h} \end{bmatrix} = \begin{bmatrix} C_\theta C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ C_\theta S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi \\ -S_\theta & S_\phi C_\theta & C_\phi C_\theta \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} + \frac{1}{m} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}$$

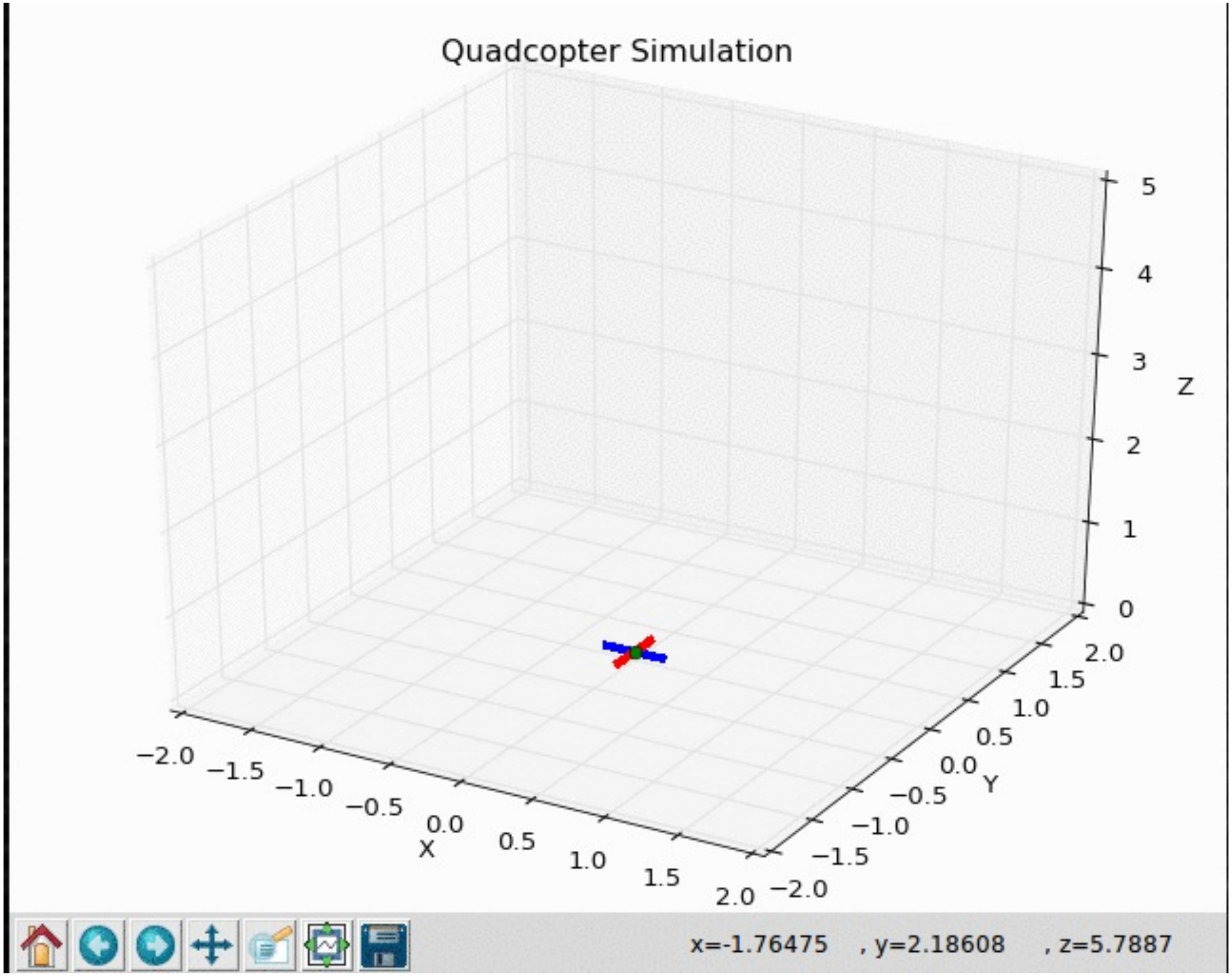
$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi / C_\theta & C_\phi / C_\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \{(I_y - I_z) qr + \tau_\phi\} / I_x \\ \{(I_z - I_x) pr + \tau_\theta\} / I_y \\ \{(I_x - I_y) pq + \tau_\psi\} / I_z \end{bmatrix}$$

UAV Workshop

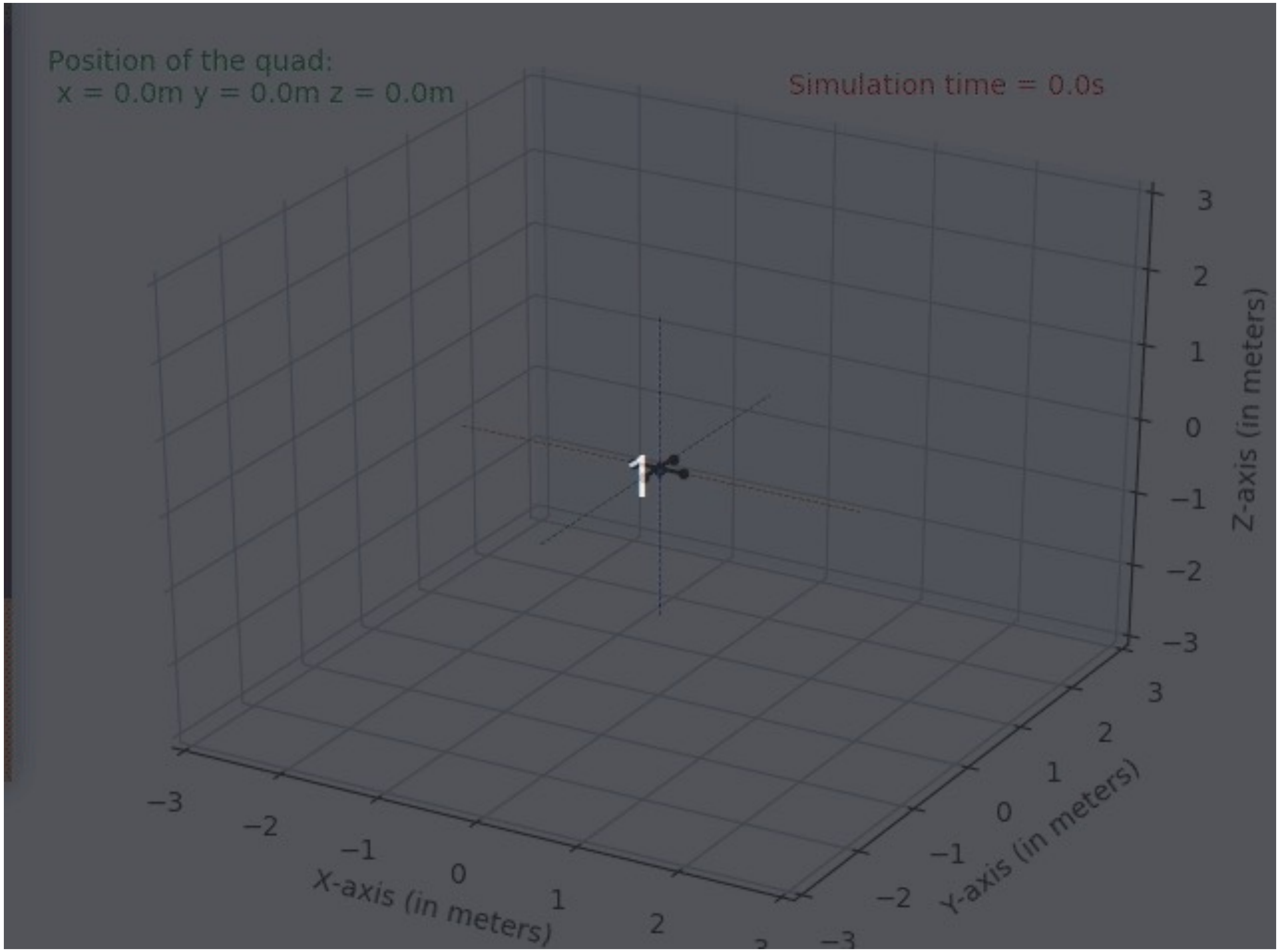
Quadcopter Simulator

https://github.com/abhijitmajumdar/Quadcopter_simulator

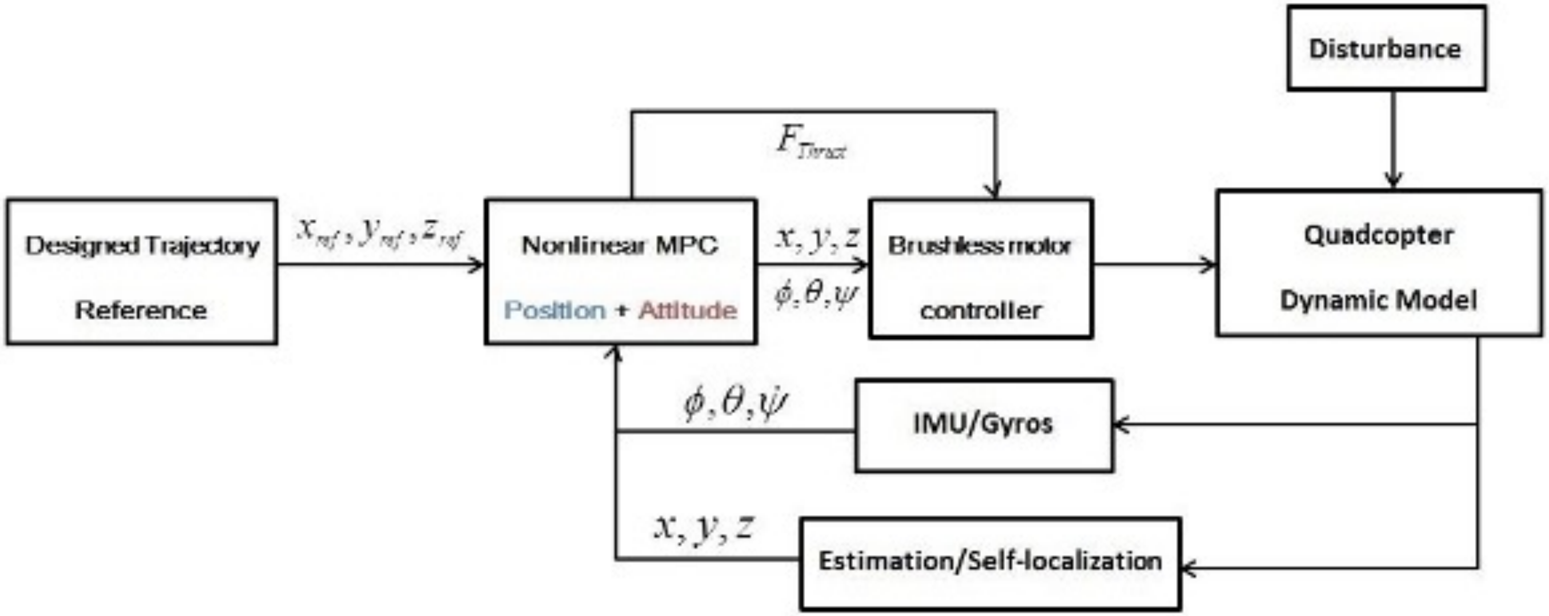


https://github.com/NishanthARao/Python-Quadrotor-Simulation

UAV Workshop



Controller


Disturbance
F_{Thrust}
Designed Trajectory
Reference
$x_{ref}, y_{ref}, z_{ref}$
Nonlinear MPC
Position + Attitude
x, y, z
ϕ, θ, ψ
Brushless motor
controller
Quadcopter
Dynamic Model
$\phi, \theta, \dot{\psi}$
IMU/Gyros
x, y, z
Estimation/Self-localization