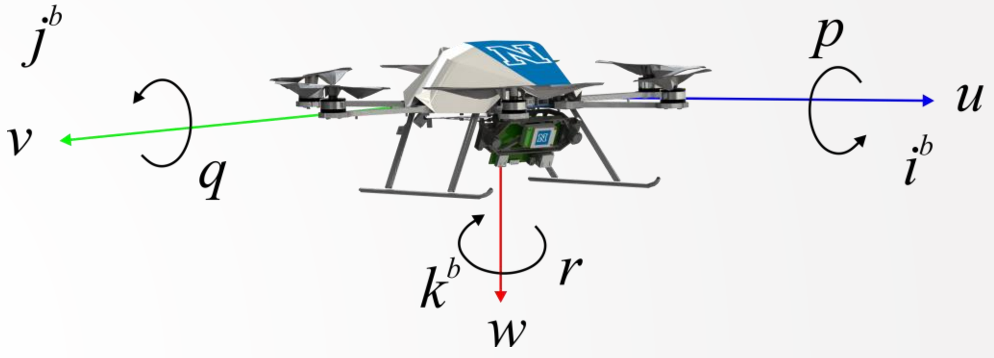
# Autonomous Mobile Robot Design

## Topic: Micro Aerial Vehicle Dynamics

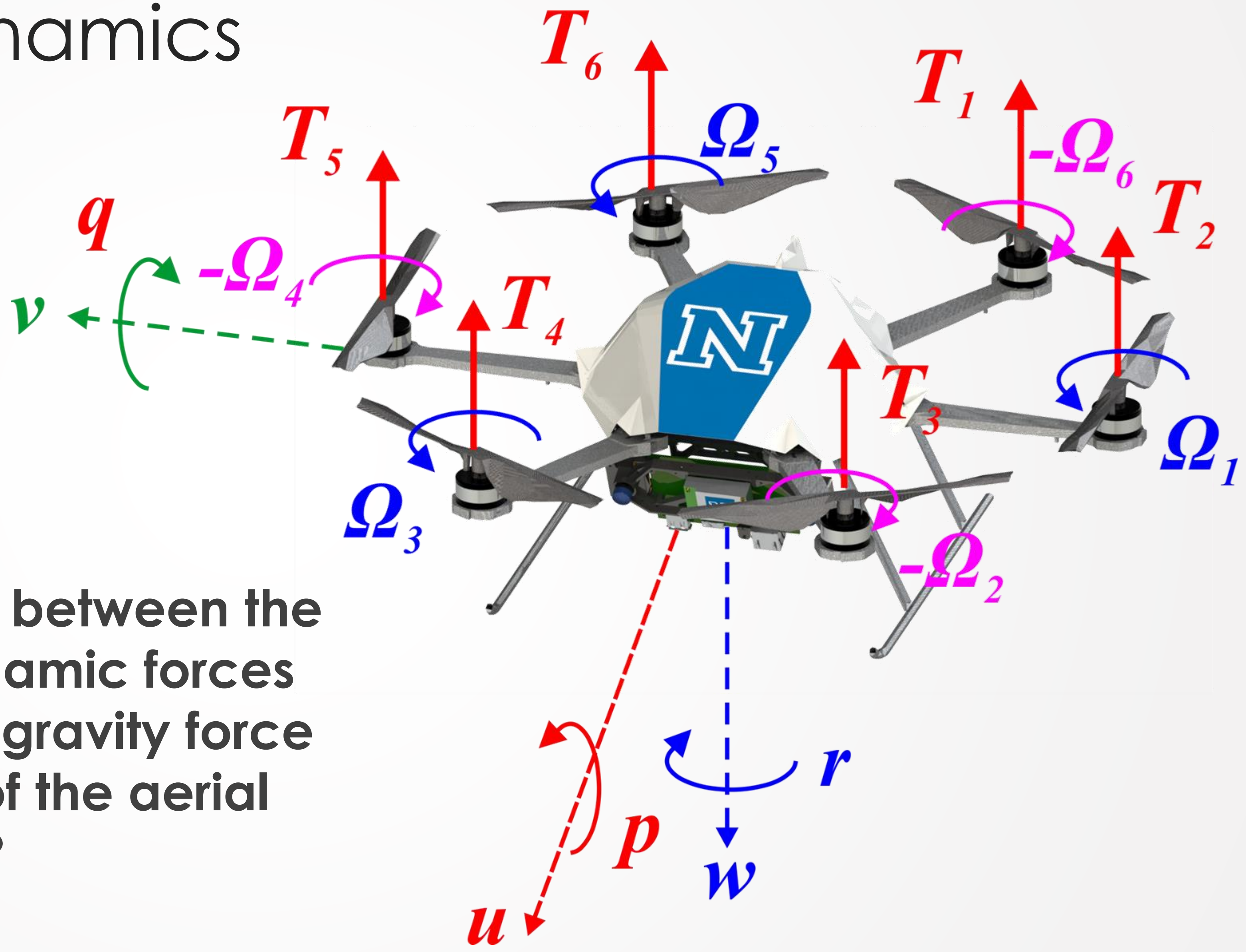
Dr. Kostas Alexis (CSE)

# Goal of this lecture

- The goal of this lecture is to derive the equations of motion that describe the motion of a multirotor Micro Aerial Vehicle.
- This Micro Aerial Vehicle (MAV) has 6 Degrees of Freedom but only 4 distinct inputs.
  - It is an underactuated system.
- To achieve this goal, we rely on:
  - A model of the Aerodynamic Forces & Moments
  - A model of the motion of the vehicle body as actuated by the forces and moments acting on it.



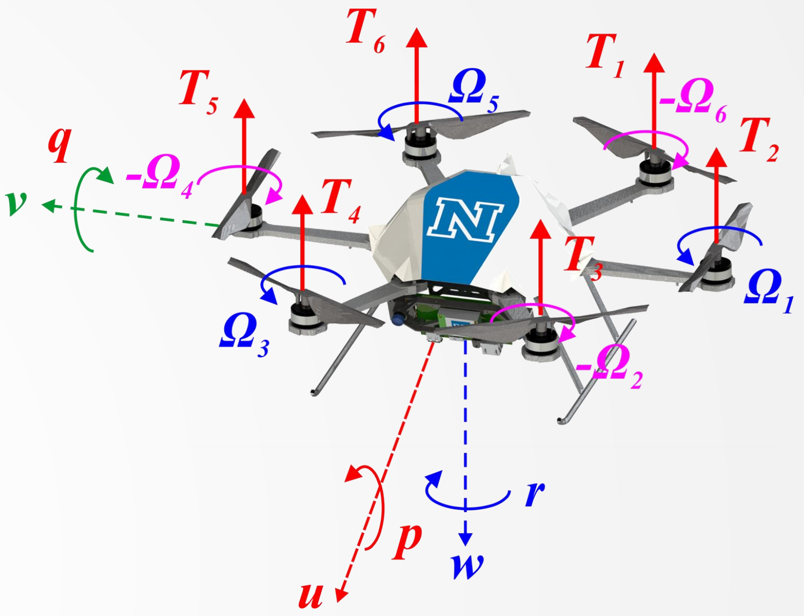
# MAV Dynamics




**What is the relation between the propeller aerodynamic forces and moments, the gravity force and the motion of the aerial robot?**

# MAV Dynamics

- Assumption 1: the Micro Aerial Vehicle is flying as a rigid body with negligible aerodynamic effects on it – for the employed airspeeds.
- The propeller is considered as a simple propeller disc that generates thrust and a moment around its shaft.



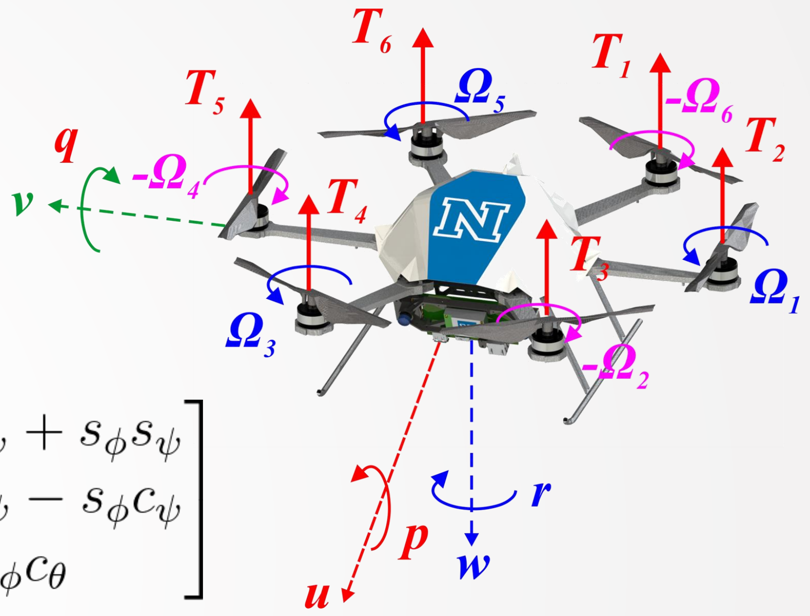

- Recall:

$$F_T = T = C_T \rho A (\Omega R)^2$$
$$M_Q = Q = C_Q \rho A (\Omega R)^2 R$$

- And let us write:

$$T_i = k_n \Omega_i^2$$
$$M_i = (-1)^{i-1} k_m T_i$$

# MAV Dynamics



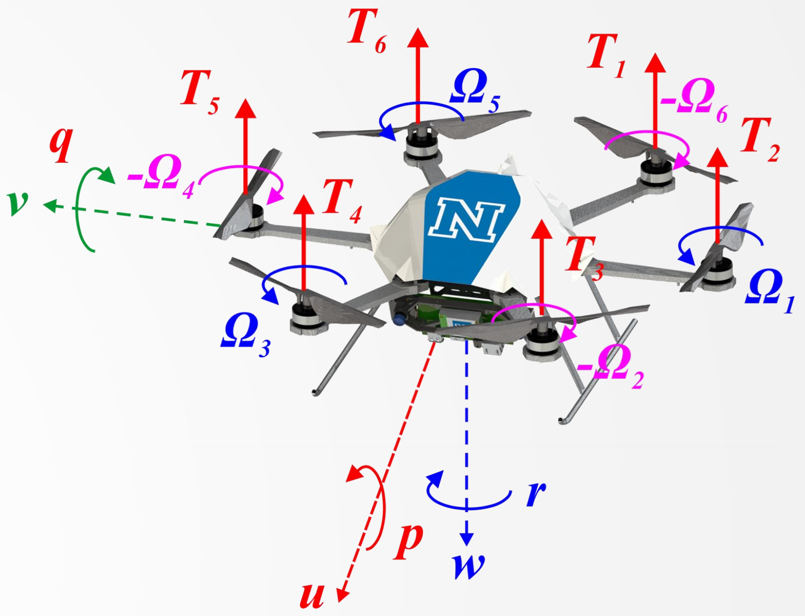

- Recall the kinematic equations:
  - Translational Kinematic Expression:

$$\begin{bmatrix} \dot{p}_n \\ \dot{p}_e \\ \dot{p}_d \end{bmatrix} = \mathcal{R}_b^v \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \quad \mathcal{R}_b^v = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix}$$

  - Rotational Kinematic Expression

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

# MAV Dynamics




- Recall Newton's 2nd Law:

$$m\frac{d\mathbf{V}_g}{dt_i} = \mathbf{f}$$

  - $f$ is the summary of all external forces
  - $m$ is the mass of the robot
  - Time derivative is taken wrt the interial frame

- Using the expression:

$$\frac{d\mathbf{V}_g}{dti} = \frac{d\mathbf{V}_g}{dt_b} + \omega_{b/i} \times \mathbf{V}_g \quad \Rightarrow \quad m(\frac{d\mathbf{V}_g}{dt_b} + \omega_{b/i} \times \mathbf{V}_g) = \mathbf{f}$$

- Which expressed in the body frame:

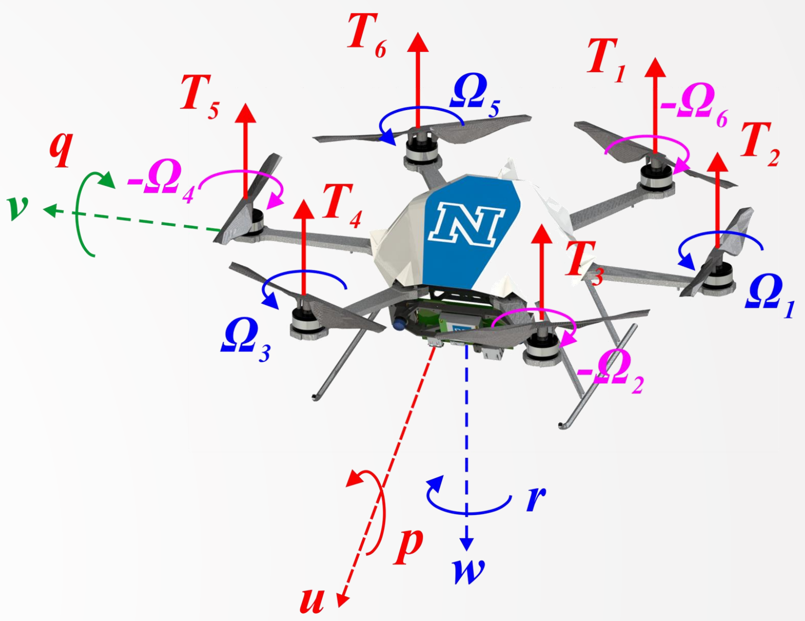
$$m(\frac{d\mathbf{V}_g^b}{dt_b} + \omega_{b/i}^b \times \mathbf{V}_g^b) = \mathbf{f}^b$$

# MAV Dynamics

- Recall Newton's 2nd Law:

$$m\frac{d\mathbf{V}_g}{dt_i} = \mathbf{f}$$

  - $f$ is the summary of all external forces
  - $m$ is the mass of the robot
  - **Time derivative is taken wrt the interial frame**




- Time Derivatives in a Rotating Frame:
  - Introduce the unit vectors ***i,j,k*** representing standard unit basis vectors in the rotating frame. As they rotate they will remain normalized. If we let them rotate at a speed $\Omega$ about an axis $\boldsymbol{\Omega}$ then each unit vector ***u*** of the rotating coordinate system abides by the rule:

$$\frac{d}{dt}\mathbf{u} = \boldsymbol{\Omega} \times \mathbf{u}$$

# MAV Dynamics

- Time Derivatives in a Rotating Frame:
  - Introduce the unit vectors ***i,j,k*** representing standard unit basis vectors in the rotating frame. As they rotate they will remain normalized. If we let them rotate at a speed $\Omega$ about an axis $\boldsymbol{\Omega}$ then each unit vector $\boldsymbol{u}$ of the rotating coordinate system abides by the rule:

$$\frac{d}{dt}\mathbf{u} = \boldsymbol{\Omega} \times \mathbf{u}$$

  - Then if we have a unit vector:

$$\mathbf{f}(t) = \mathbf{f}_x(t)\mathbf{i} + \mathbf{f}_y(t)\mathbf{j} + \mathbf{f}_z(t)\mathbf{k}$$

  - To examine its first derivative – we have to use the product rule of differentiation:

$$\frac{d}{dt}\mathbf{f} = \frac{df_x}{dt}\mathbf{i} + \frac{d\mathbf{i}}{dt}f_x + \frac{df_y}{dt}\mathbf{j} + \frac{d\mathbf{j}}{dt}f_y + \frac{df_z}{dt}\mathbf{k} + \frac{d\mathbf{k}}{dt}f_z \Rightarrow$$

$$\frac{d}{dt}\mathbf{f} = \left[\left(\tfrac{d}{dt}\right)_r + \boldsymbol{\Omega}\times\right]\mathbf{f}$$

# MAV Dynamics

- Time Derivatives in a Rotating Frame:
  - Introduce the unit vectors ***i,j,k*** representing standard unit basis vectors in the rotating frame. As they rotate they will remain normalized. If we let them rotate at a speed $\Omega$ about an axis $\boldsymbol{\Omega}$ then each unit vector ***u*** of the rotating coordinate system abides by the rule:

$$\frac{d}{dt}\mathbf{u} = \boldsymbol{\Omega} \times \mathbf{u}$$

  - Then if we have a unit vector:

$$\mathbf{f}(t) = \mathbf{f}_x(t)\mathbf{i} + \mathbf{f}_y(t)\mathbf{j} + \mathbf{f}_z(t)\mathbf{k}$$

  - To examine its first derivative – we have to use the product rule of differentiation:

$$\frac{d}{dt}\mathbf{f} = \frac{df_x}{dt}\mathbf{i} + \frac{d\mathbf{i}}{dt}f_x + \frac{df_y}{dt}\mathbf{j} + \frac{d\mathbf{j}}{dt}f_y + \frac{df_z}{dt}\mathbf{k} + \frac{d\mathbf{k}}{dt}f_z \Rightarrow$$

$$\frac{d}{dt}\mathbf{f} = \left[\left(\frac{d}{dt}\right)_r + \boldsymbol{\Omega}\times\right]\mathbf{f}$$

# MAV Dynamics

- As a result: Relation Between Velocities in the Inertial & Rotating Frame
  - Let $\boldsymbol{v}$ be the position of an object's position:

$$\mathbf{v} = \frac{d}{dt}\mathbf{p}$$

  - Then the relation of the velocity as expressed in the inertial frame and as expressed in the rotating frame becomes:

$$\mathbf{v}_i = \mathbf{v}_r + \boldsymbol{\Omega} \times \mathbf{p}$$

- Similarly: Relation Between Accelerations in the Inertial & Rotating Frame
  - Let $\boldsymbol{a}$ be the acceleration of an object's position. Then:

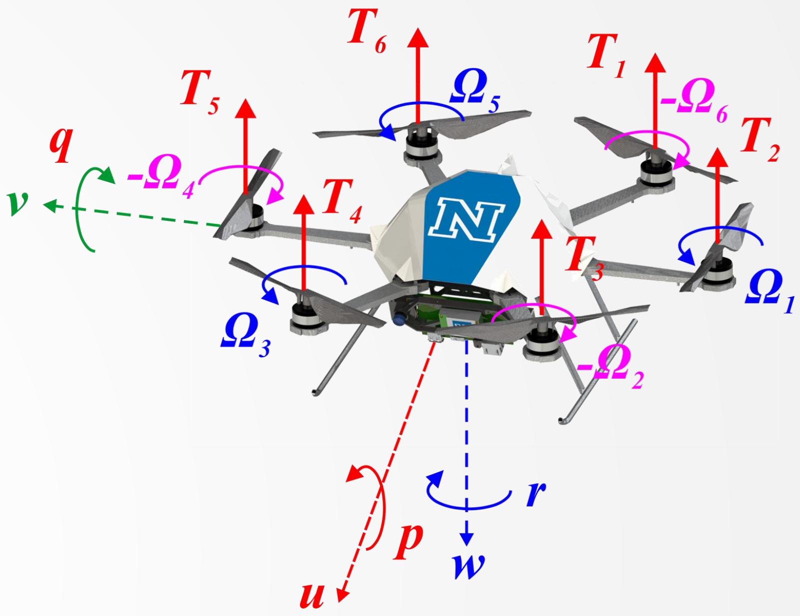
$$\mathbf{a}_i = \left(\frac{d\mathbf{p}}{dt}\right)_i = \left(\frac{d\mathbf{v}}{dt}\right)_i = \left[\left(\frac{d}{dt}_r + \boldsymbol{\Omega}\times\right)\right]\left[\left(\frac{d\mathbf{p}}{dt}\right)_r + \boldsymbol{\Omega} \times \mathbf{p}\right]$$

  - Carrying out the differentiations:

$$\mathbf{a}_r = \mathbf{a}_i - 2\boldsymbol{\Omega} \times \mathbf{v}_r - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{p}) - \frac{d\boldsymbol{\Omega}}{dt} \times \mathbf{p}$$

- *Subscripts $i$, $r$ represent the inertial frame and the rotating frame respectively.*

# MAV Dynamics




- Recall Newton's 2nd Law:

$$m\frac{d\mathbf{V}_g}{dt_i} = \mathbf{f}$$

  - $f$ is the summary of all external forces
  - $m$ is the mass of the robot
  - Time derivative is taken wrt the interial frame
- Using the expression:

$$\frac{d\mathbf{V}_g}{dti} = \frac{d\mathbf{V}_g}{dt_b} + \omega_{b/i} \times \mathbf{V}_g \quad \Rightarrow \quad m(\frac{d\mathbf{V}_g}{dt_b} + \omega_{b/i} \times \mathbf{V}_g) = \mathbf{f}$$

- Which expressed in the body frame:

$$m(\frac{d\mathbf{V}_g^b}{dt_b} + \omega_{b/i}^b \times \mathbf{V}_g^b) = \mathbf{f}^b$$

# MAV Dynamics

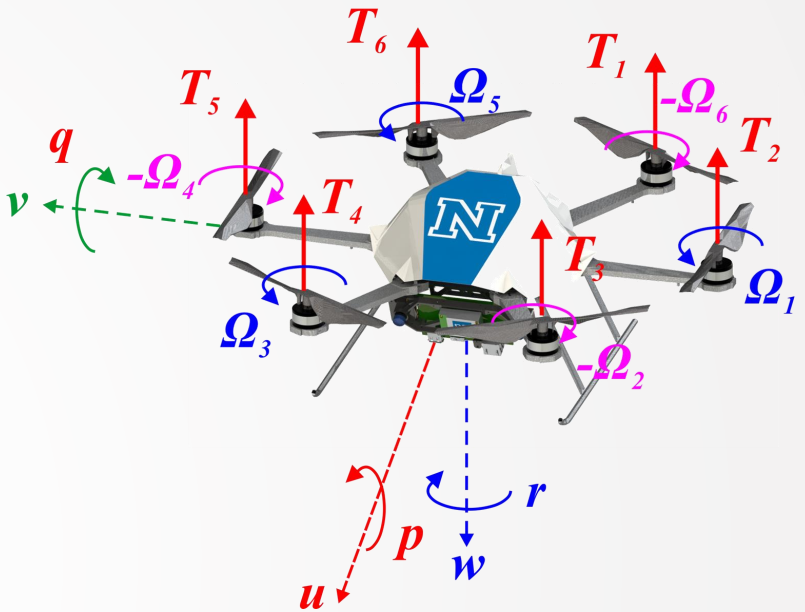
$$m\left(\frac{d\mathbf{V}_g^b}{dt_b} + \omega_{b/i}^b \times \mathbf{V}_g^b\right) = \mathbf{f}^b$$

- Where

$$\mathbf{V}_g^b = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \quad \omega_{b/i}^b = \begin{bmatrix} p \\ q \\ r \end{bmatrix}, \mathbf{f}^b = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}$$

- Therefore:

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



# MAV Dynamics

- Recall Newton's 2nd Law:

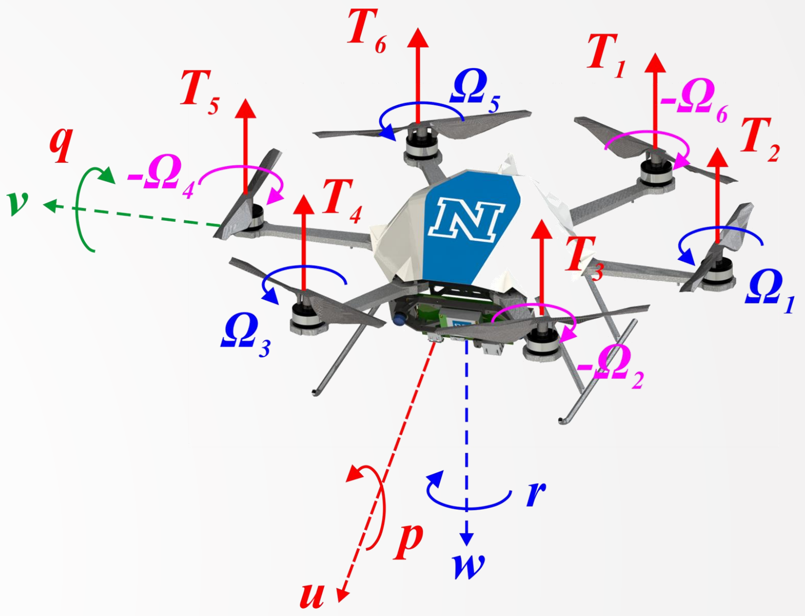
$$\frac{d\mathbf{h}}{dt_i} = \mathbf{m}$$

  - ***h*** is the angular momentum vector
  - ***m*** is the summary of all external moments
  - Time derivative is taken wrt the interial frame
- Therefore:

$$\frac{d\mathbf{h}}{dt_i} = \frac{d\mathbf{h}}{dt_b} + \omega_{b/i} \times \mathbf{h} = \mathbf{m}$$

- Which expressed in the body frame:

$$\frac{d\mathbf{h}^b}{dt_b} + \omega_{b/i}^b \times \mathbf{h}^b = \mathbf{m}^b$$



# MAV Dynamics

- Recall Newton's 2nd Law:

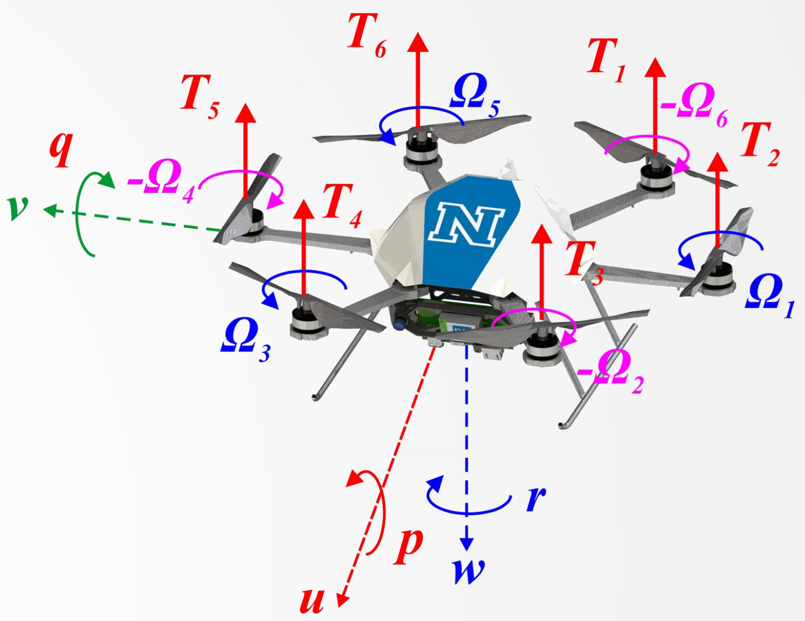
$$\frac{d\mathbf{h}}{dt_i} = \mathbf{m}$$

  - ***h*** is the angular momentum vector
  - ***m*** is the summary of all external moments
  - Time derivative is taken wrt the interial frame
- Therefore:

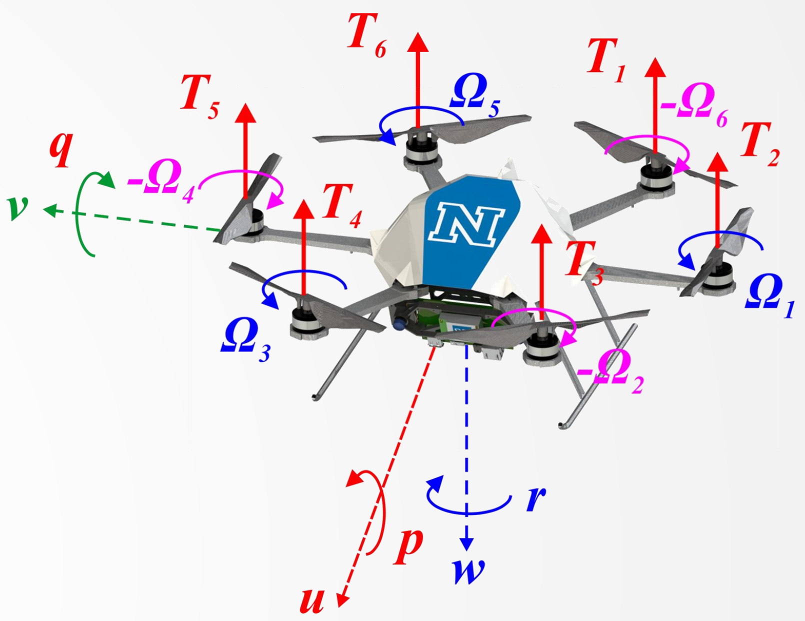
$$\frac{d\mathbf{h}}{dt_i} = \frac{d\mathbf{h}}{dt_b} + \omega_{b/i} \times \mathbf{h} = \mathbf{m}$$

- Which expressed in the body frame:

$$\frac{d\mathbf{h}^b}{dt_b} + \omega_{b/i}^b \times \mathbf{h}^b = \mathbf{m}^b$$



# MAV Dynamics




- For a rigid body, the angular momentum is defined as the product of the inertia matrix and the angular velocity vector:

$$\mathbf{h}^b = \mathbf{J}\omega^b_{b/i}$$

- where

$$\mathbf{J} = \begin{bmatrix} J_x & -J_{xy} & -J_{xz} \\ -J_{xy} & J_y & -J_{yz} \\ -J_{xz} & -J_{yz} & J_z \end{bmatrix}$$

- But as the multirotor MAV is symmetric:

$$\mathbf{J} = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix}$$

# MAV Dynamics

- Replacing in:

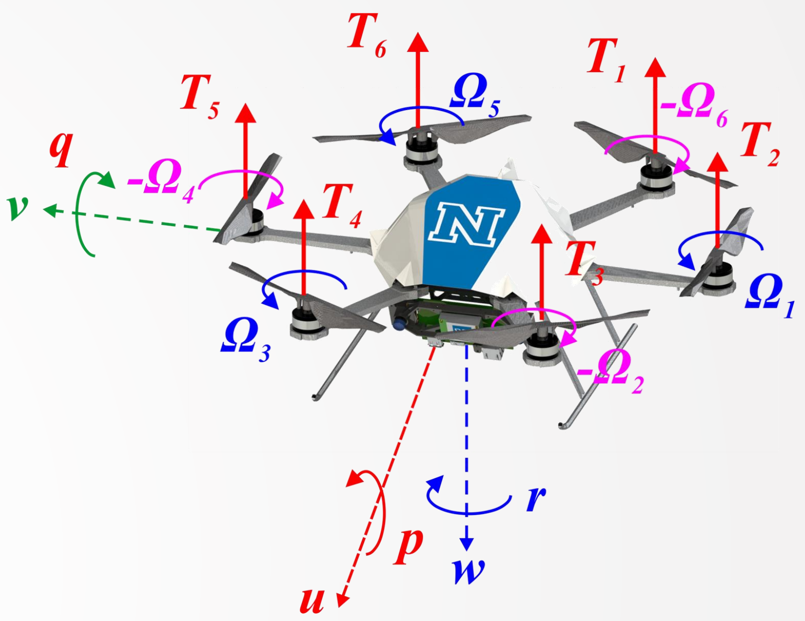
$$\frac{d\mathbf{h}^b}{dt_b} + \omega_{b/i}^b \times \mathbf{h}^b = \mathbf{m}^b$$

- Gives:

$$\mathbf{J}\frac{d\omega_{b/i}^b}{dt_b} + \omega_{b/i}^b \times (\mathbf{J}\omega_{b/i}^b) = \mathbf{m}^b \Rightarrow$$
$$\dot{\omega}_{b/i}^b = \mathbf{J}^{-1}[-\omega_{b/i}^b \times (\mathbf{J}\omega_{b/i}^b) + \mathbf{m}^b]$$

- where

$$\dot{\omega}_{b/i}^b = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}$$



# MAV Dynamics

- By setting the moments vector:

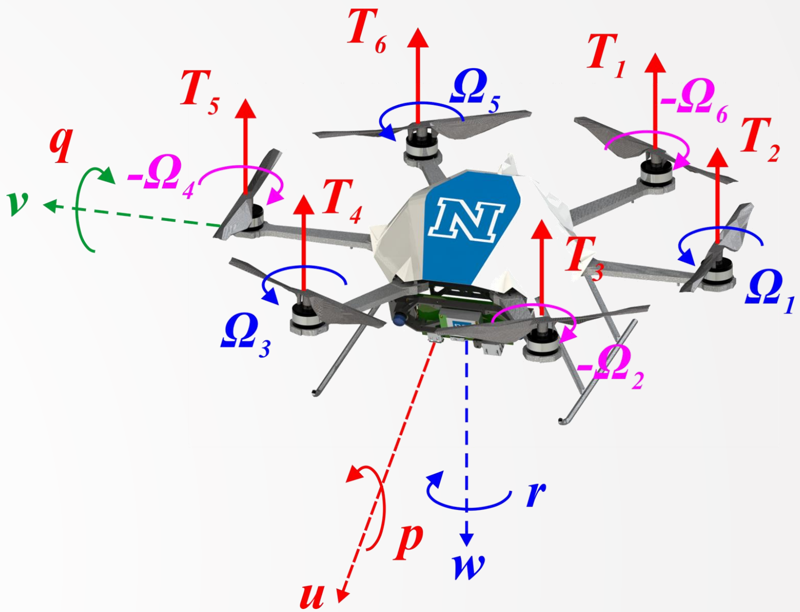
$$\mathbf{m}^b = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix}$$

- Then for the symmetric MAV, equation:

$$\dot{\omega}^b_{b/i} \overset{\vee}{=} \mathbf{J}^{-1}[-\omega^b_{b/i} \times (\mathbf{J}\omega^b_{b/i}) + \mathbf{m}^b]$$

- Becomes:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{J_y - J_z}{J_x} qr \\ \frac{J_z - J_x}{J_y} pr \\ \frac{J_x - J_y}{J_z} qr \end{bmatrix} + \begin{bmatrix} \frac{1}{m} M_x \\ \frac{1}{m} M_y \\ \frac{1}{z} M_z \end{bmatrix}$$



# MAV Dynamics

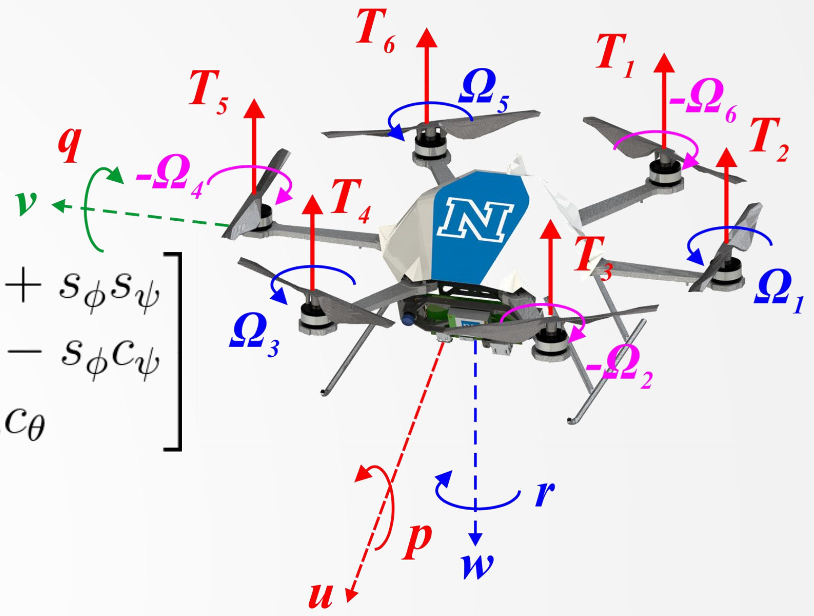
- To append the forces and moments we need to combine their formulation with

$$\begin{bmatrix}\dot{p}_n\\ \dot{p}_e\\ \dot{p}_d\end{bmatrix} = \mathcal{R}_b^v \begin{bmatrix}u\\ v\\ w\end{bmatrix}, \quad \mathcal{R}_b^v = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix}$$

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

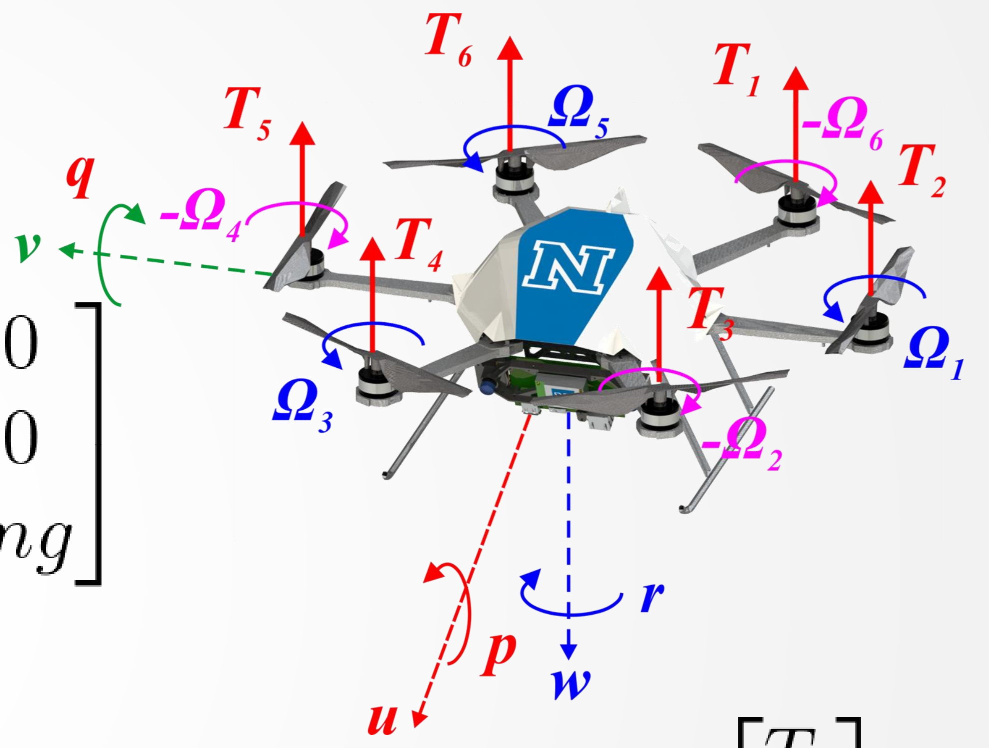
$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}\begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix} \frac{J_y - J_z}{J_x} qr \\ \frac{J_z - J_x}{J_y} pr \\ \frac{J_x - J_y}{J_z} qr \end{bmatrix} + \begin{bmatrix} \frac{1}{m} M_x \\ \frac{1}{m} M_y \\ \frac{1}{z} M_z \end{bmatrix}$$




- ***Next step: append the MAV forces and moments***

# MAV Dynamics




- MAV forces in the body frame:

$$\mathbf{f}_b = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^{6} T_i \end{bmatrix} - \mathcal{R}_v^b \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}$$

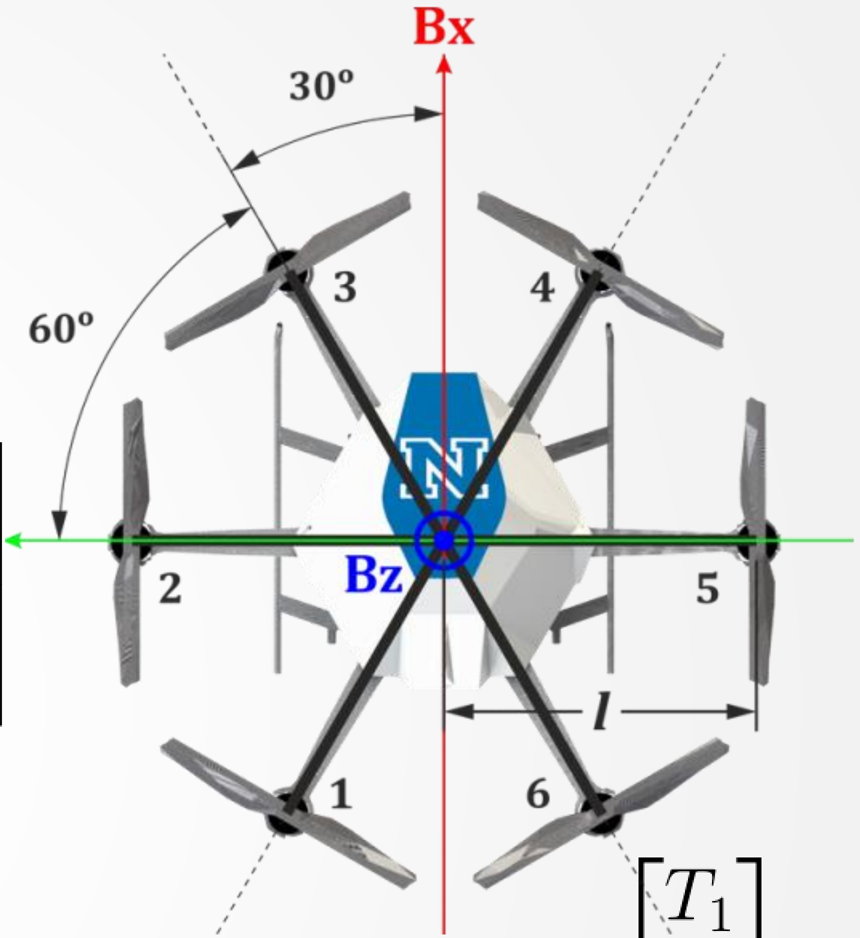
- Moments in the body frame:

$$\mathbf{m}_b = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} ls_{30} & l & ls_{30} & -ls_{30} & -l & ls_{30} \\ -lc_{60} & 0 & lc_{60} & lc_{60} & 0 & -lc_{60} \\ -k_m & k_m & -k_m & k_m & -k_m & k_m \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \end{bmatrix}$$

# MAV Dynamics

- MAV forces in the body frame:

$$\mathbf{f}_b = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^{6} T_i \end{bmatrix} - \mathcal{R}_v^b \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}$$




- Moments in the body frame:

$$\mathbf{m}_b = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} ls_{30} & l & ls_{30} & -ls_{30} & -l & ls_{30} \\ -lc_{60} & 0 & lc_{60} & lc_{60} & 0 & -lc_{60} \\ -k_m & k_m & -k_m & k_m & -k_m & k_m \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \end{bmatrix}$$

# Code Example

- MATLAB Quadrotor Simulator:
  - https://github.com/unr-arl/autonomous_mobile_robot_design_course/tree/master/matlab/vehicle-dynamics
  - Accurate dynamics simulator with further realistic features on sensing data and planning algorithms.
  - Create by: Ke Sun, University of Pennsylvania
  - run "quad_sim.m"

- ROS/Gazebo Multirotor Simulator:
  - http://www.kostasalexis.com/rotors-simulator3.html
  - Advanced aerial robots simulator, recreating real-life autonomous operation in terms of actuation, dynamics, control systems, sensor systems, localization algorithms, as well as path planning.
  - Very realistic – relying on Gazebo and physics engine.
  - roslaunch rotors_gazebo mav_hovering_example.launch mav_name:=firefly world_name:=basic

# Code Example

- Indicative in-class run

# Code Example

- Want to learn more? Want to get a small bonus?
  - Run different trajectories using the MATLAB simulator.
  - Run an example of RotorS
    - Send me a video of the results

  - Each of the above gives +2% in your overall grade (absolute scale)

# Find out more

- S. Leutenegger,C. Huerzeler, A.K. Stowers, K. Alexis, M. Achtelik, D. Lentink, P. Oh, and R. Siegwart. **"Flying Robots"**, *Handbook of Robotics* (upcoming new version – available upon request).
- Python? http://www.kostasalexis.com/simulations-with-simpy.html
- Official MATLAB Demo: http://www.mathworks.com/help/aeroblks/examples/quadcopter-project.html?refresh=true
- **Help with Differential Equations?** https://www.khanacademy.org/math/differential-equations
- **Always check:** http://www.kostasalexis.com/literature-and-links1.html

# Thank you!

Please ask your question!