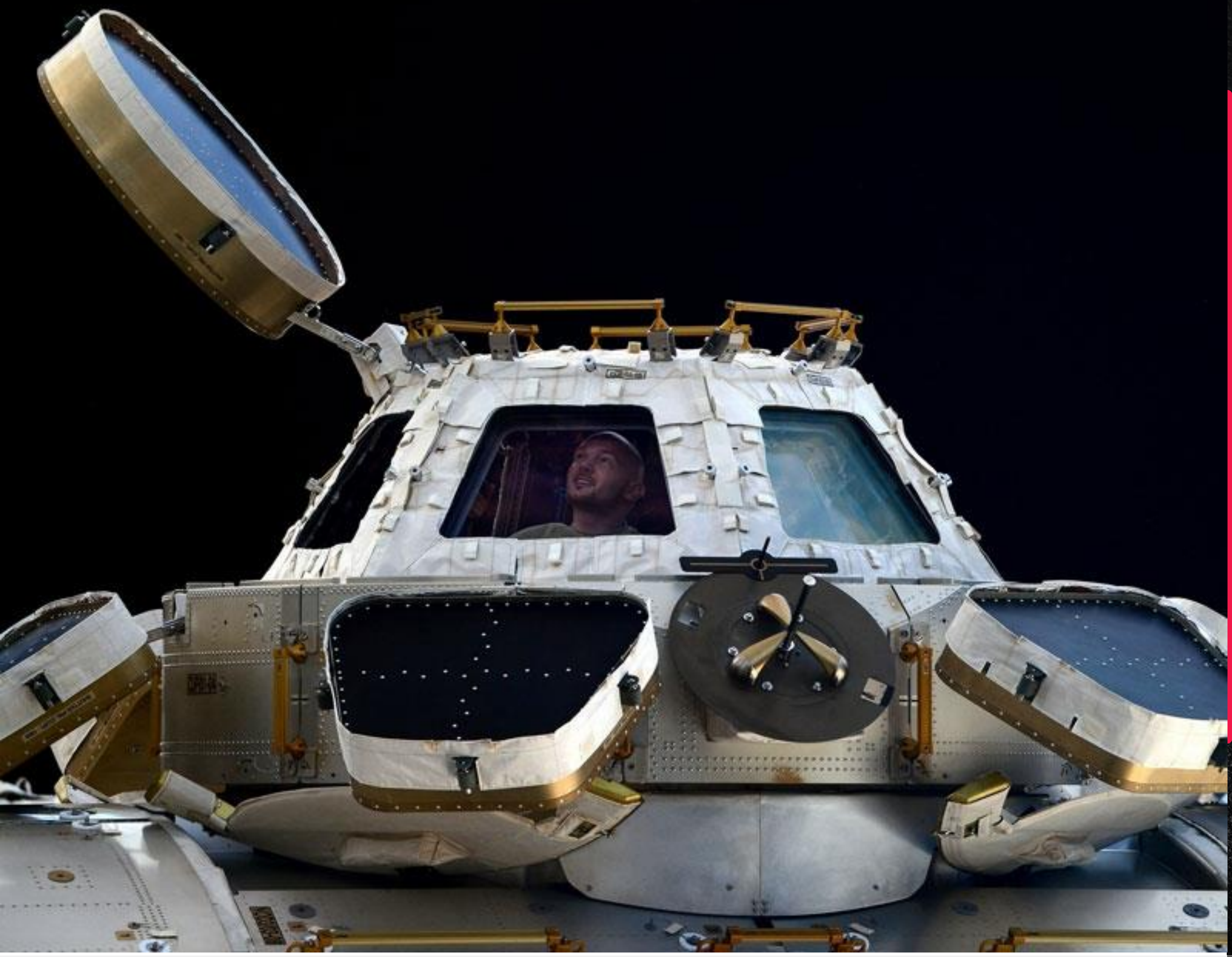


MOTION UAV

Open Source Autopilots

Presented by:

Iman Shirdareh

Outline

- Why, How, What?
- Basics of Pixhawk project
- Pixhawk FMU hardware overview
- Basic concepts
- Control theory
- Estimation theory
- Pixhawk Software overview
- Operational Software Guide
- Operational Assembly Guide
- Flying Guide
- Flight Log Analysis Guide
- Advanced Guides

Basic concepts

Navigation and Guidance

Main purposes

- Specifying of state variables (Position, Velocity, Acceleration, Angle,…) at each moment
- Defining of last desired position
- What should it do to follow the specified path
- Specifying desired Angle, Heading, Side acceleration,…
- Running defined desired variable

Basic concepts

Navigation and Guidance

Guidance

Path planning & Trajectory planning/ Find last position or velocity

Basic concepts

Navigation and Guidance

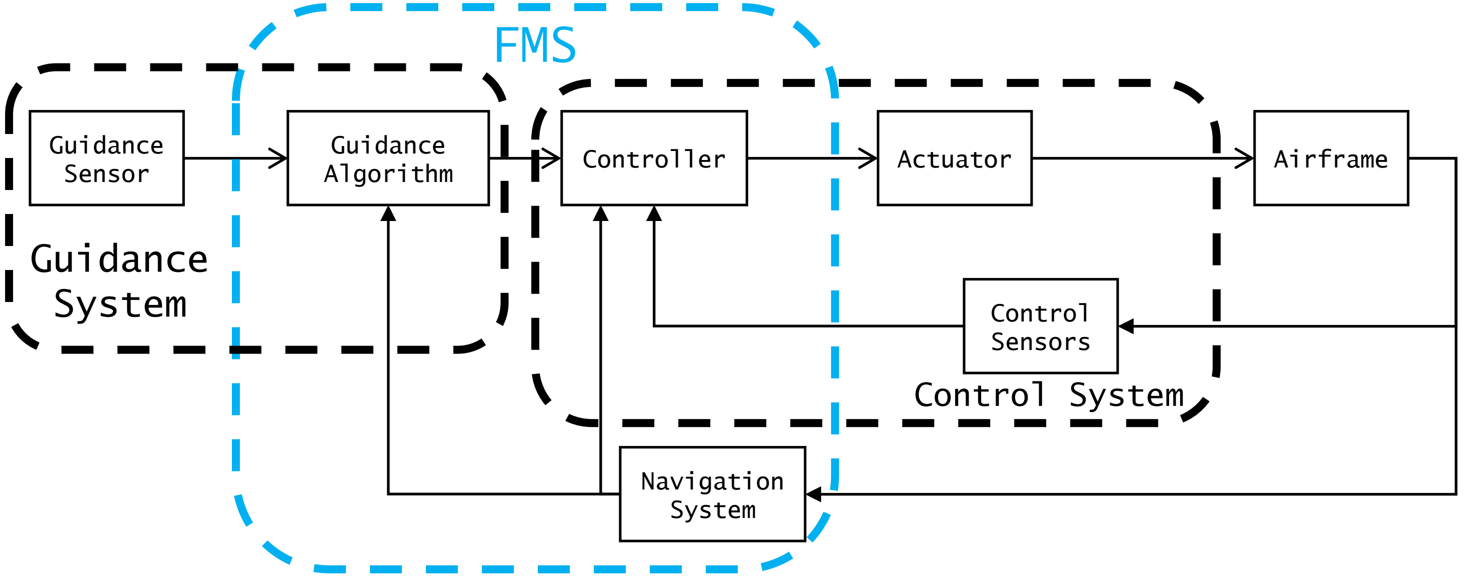
Navigation

Calculating position& velocity at each moment

Basic concepts

Navigation and Guidance

Block Diagram of a flying system



Basic concepts

Navigation and Guidance

Guidance Supplement

- Guidance Sensor
- Guidance Processor
- Guidance Algorithm

Basic concepts

Navigation and Guidance

Guidance Sensor

Basic concepts

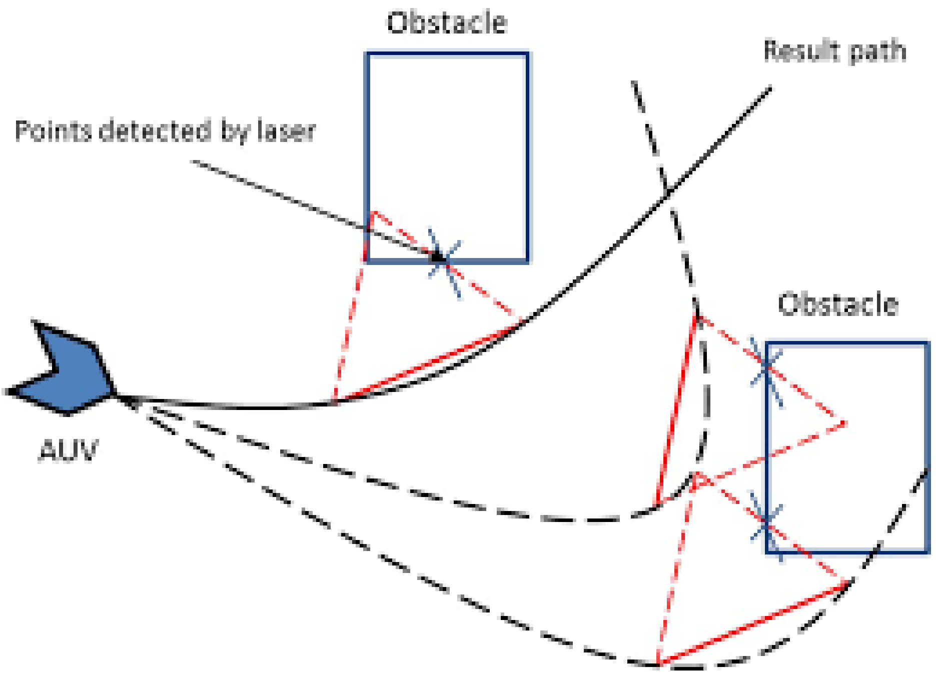
Navigation and Guidance

Guidance Computer

Basic concepts

Navigation and Guidance

Guidance Algorithm



Basic concepts

Navigation and Guidance

Guidance Phases

- ❑ Launch/Take-off
- ❑ Crusoe
- ❑ Land

Basic concepts

Navigation and Guidance

Guidance Phases

- ❑ Launch/Take-off
- ❑ Crusoe
- ❑ Land

Basic concepts

Navigation and Guidance

Guidance paths

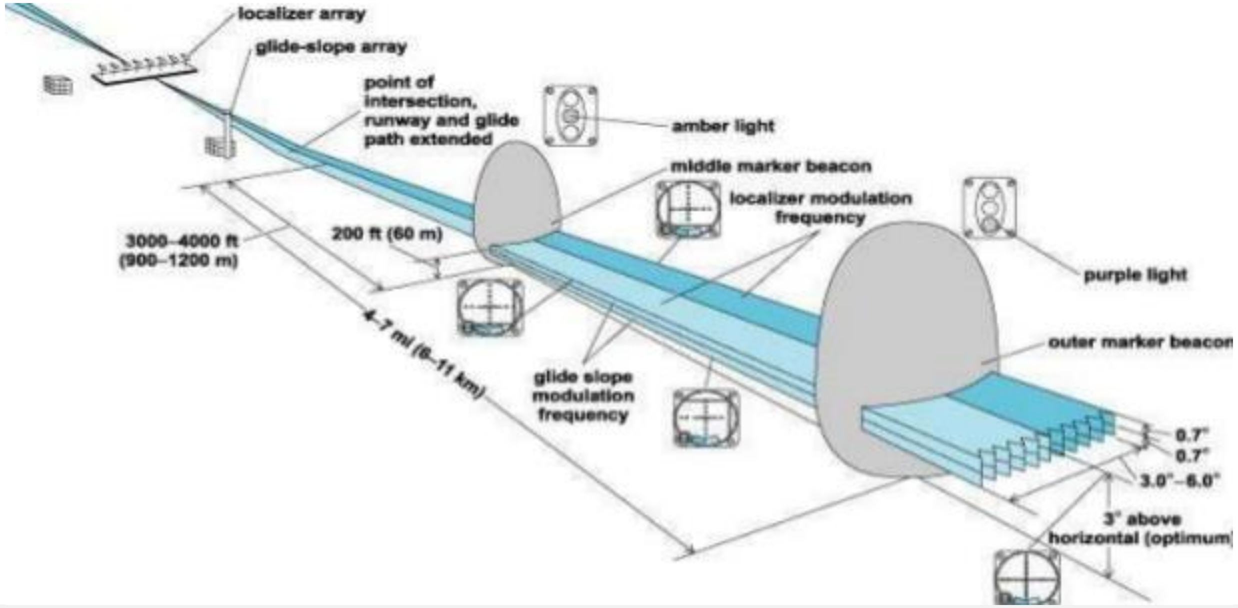
- ❑ Direct Path
- ❑ Optimum Path
- ❑ Crouse path
- ❑ Topographic path

Basic concepts

Navigation and Guidance

Guidance paths

- ❑ Direct/Collision Path
- ❑ Optimum Path
- ❑ Crouse path
- ❑ Topographic path



Basic concepts

Navigation and Guidance

Navigation Sensors

- ❑ Absolute Sensors
- ❑ Relative Sensors

Basic concepts

Navigation and Guidance

Absolute Inertial Navigation Sensors

Inertial Navigation System (INS)

- Position
- Velocity
- Angular Velocity
- Euler Angle

Basic concepts

Navigation and Guidance

Absolute Inertial Navigation Sensors

Global Positioning System(GPS)

- Position
- Velocity

Basic concepts

Navigation and Guidance

Absolute Inertial Navigation Sensors

Attitude & Heading Reference System (AHRS)

- Angular velocity
- Euler Angle
- Heading
- Altitude

Basic concepts

Navigation and Guidance

Absolute Inertial Navigation Sensors

Inertial Measurement Unit (IMU)

- Angular velocity
- Euler Angle

Basic concepts

Navigation and Guidance

Absolute Inertial Navigation Sensors

Compass

- Magnetic vector

Basic concepts

Navigation and Guidance

Vision Navigation Sensors

Camera & Laser scanner

Simultaneous

Localization

And

Mapping

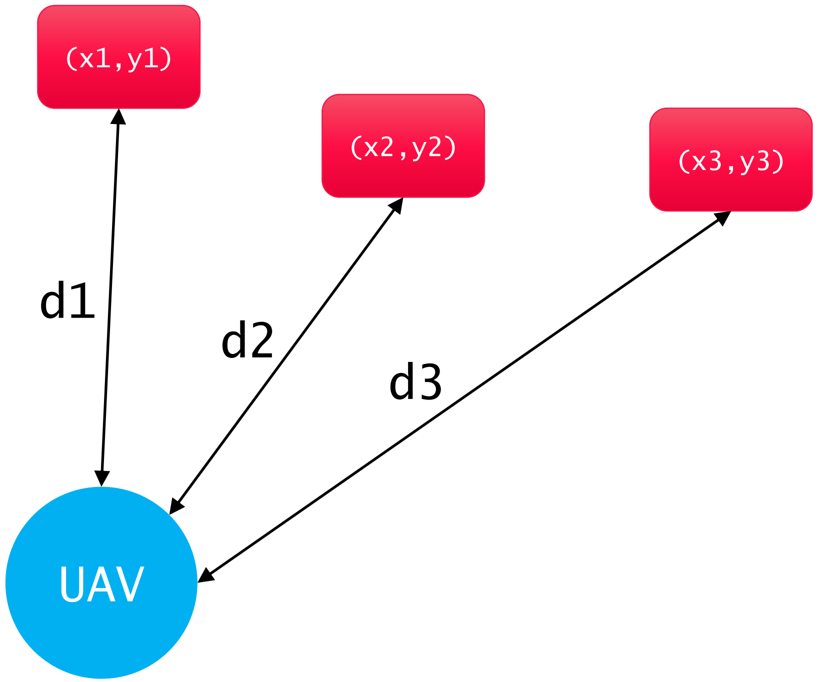
Basic concepts

Navigation and Guidance

Vision Navigation Sensors

Camera & Laser scanner

- SLAM



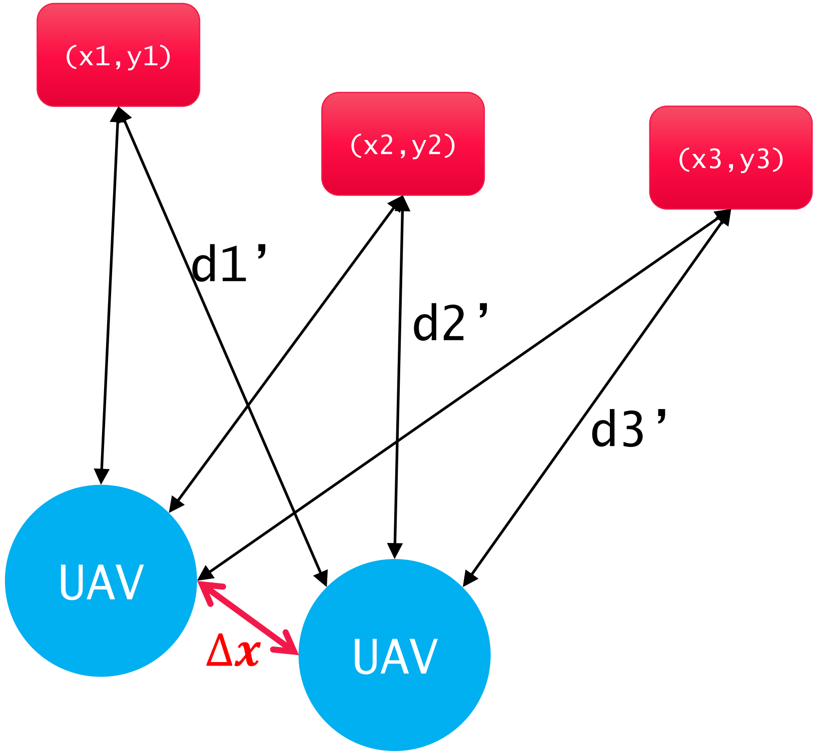
Basic concepts

Navigation and Guidance

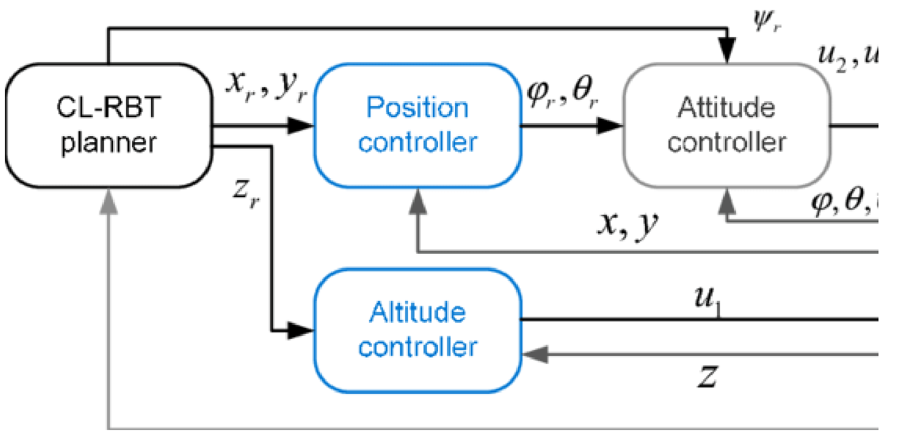
Vision Navigation Sensors

Camera & Laser scanner

- SLAM

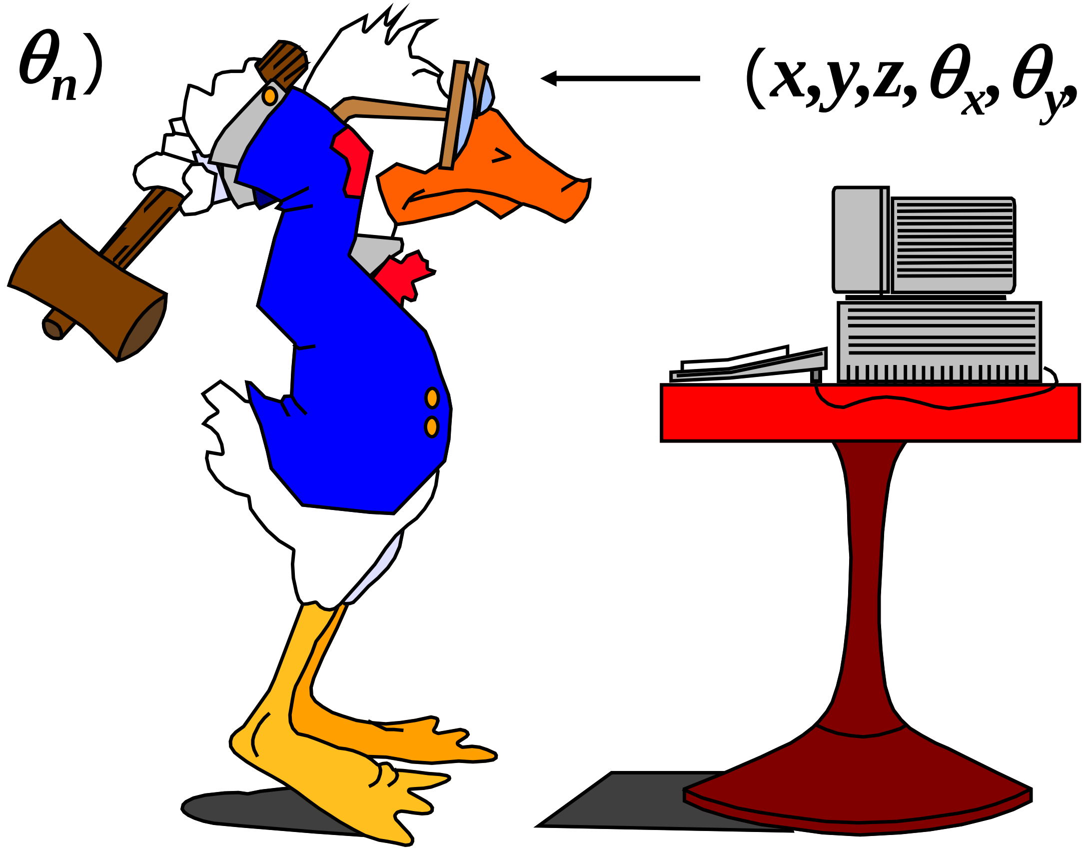


Control theory



Kinematic

Basic Mechanics



Kinematic

Basic Mechanics

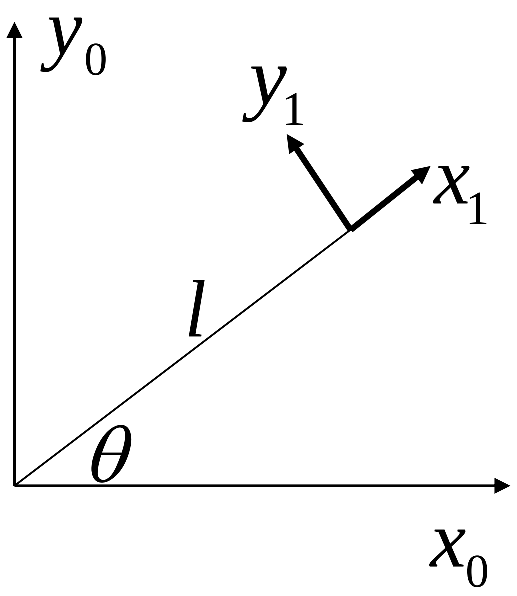
Forward kinematics

$$x_0 = l\cos\theta$$

$$y_0 = l\sin\theta$$

Inverse kinematics

$$\theta = \cos^{-1}(x_0 / l)$$



Kinematic

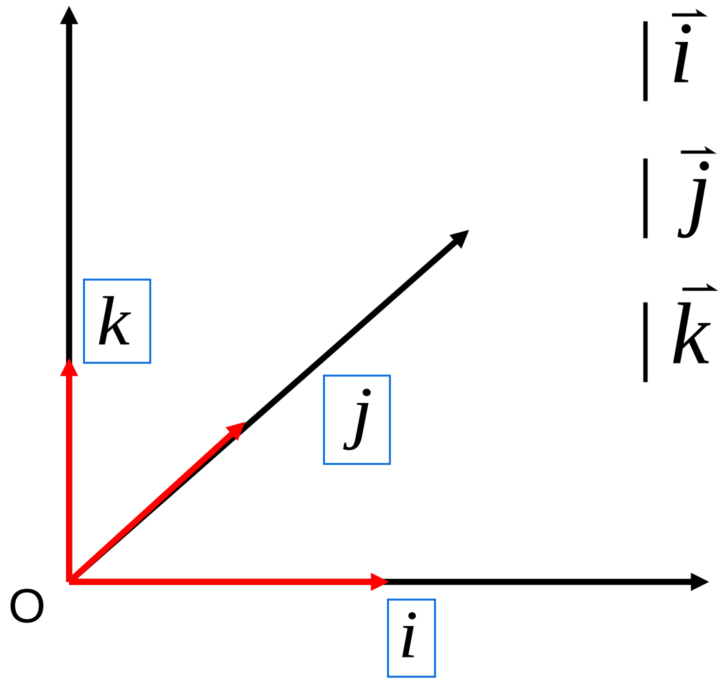
Orthogonal Matrix

Mutually perpendicular

$$\vec{i} \cdot \vec{j} = 0$$

$$\vec{i} \cdot \vec{k} = 0$$

$$\vec{k} \cdot \vec{j} = 0$$



Unit vectors

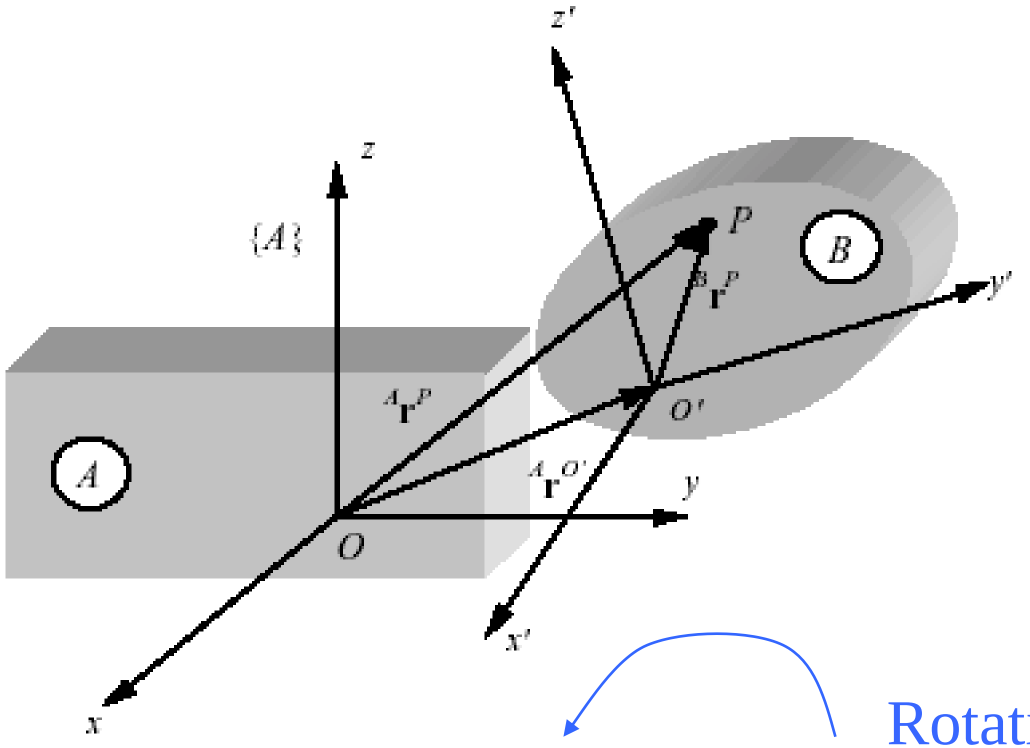
$$|\vec{i}| = 1$$

$$|\vec{j}| = 1$$

$$|\vec{k}| = 1$$

Kinematic

Rotation Matrix



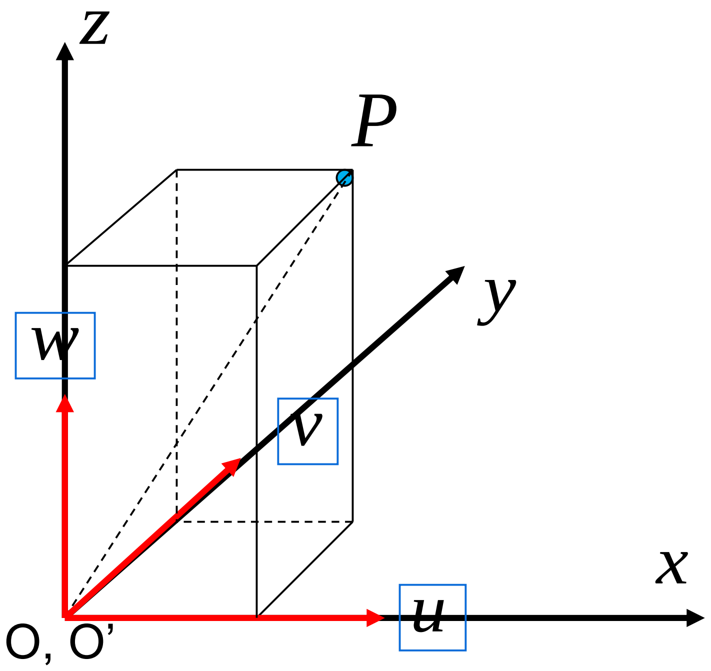
Rotation of $\{B\}$ with respect to $\{A\}$

$${}^{A}\mathbf{r}^{P} = \boxed{{}^{A}\mathbf{R}_{B}}\,{}^{B}\mathbf{r}^{P} + \boxed{{}^{A}\mathbf{r}^{O'}}$$

Translation of the origin of $\{B\}$ with respect to origin of $\{A\}$

Kinematic

Rotation Matrix



Point represented in O'uvw:

$$\vec{P}_{uvw} = p_u \mathrm{i}_\mathrm{u} + p_v \mathrm{j}_\mathrm{v} + p_w \mathrm{k}_\mathrm{w}$$

Two frames coincide ==> $p_u = p_x \quad p_v = p_y \quad p_w = p_z$

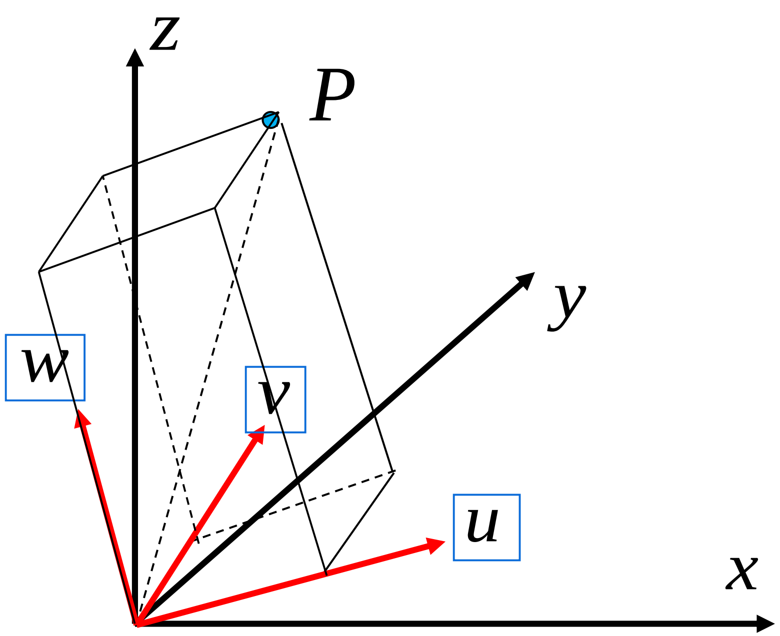
Kinematic

Rotation Matrix

$$\vec{P}_{xyz} = p_x \mathrm{i}_\mathrm{x} + p_y \mathrm{j}_\mathrm{y} + p_z \mathrm{k}_\mathrm{z}$$

$$\vec{P}_{uvw} = p_u \mathrm{i}_\mathrm{u} + p_v \mathrm{j}_\mathrm{v} + p_w \mathrm{k}_\mathrm{w}$$

$$P_{xyz} = RP_{uvw}$$



Kinematic

Rotation Matrix

P_x , P_y , and P_z represent the projections of $\boldsymbol{P}$ onto OX, OY, OZ axes, respectively

Since

$$P = p_u \mathrm{i_u} + p_v \mathrm{j_v} + p_w \mathrm{k_w}$$

$$p_x = \mathrm{i_x} \cdot P = \mathrm{i_x} \cdot \mathrm{i_u} p_u + \mathrm{i_x} \cdot \mathrm{j_v} p_v + \mathrm{i_x} \cdot \mathrm{k_w} p_w$$

$$p_y = \mathrm{j_y} \cdot P = \mathrm{j_y} \cdot \mathrm{i_u} p_u + \mathrm{j_y} \cdot \mathrm{j_v} p_v + \mathrm{j_y} \cdot \mathrm{k_w} p_w$$

$$p_z = \mathrm{k_z} \cdot P = \mathrm{k_z} \cdot \mathrm{i_u} p_u + \mathrm{k_z} \cdot \mathrm{j_v} p_v + \mathrm{k_z} \cdot \mathrm{k_w} p_w$$

Kinematic

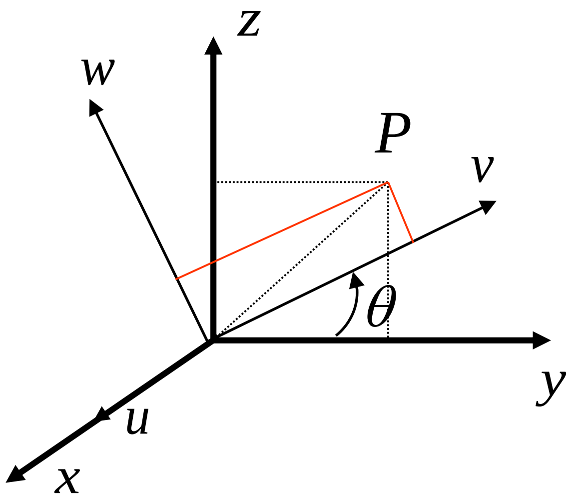
Rotation Matrix

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} \mathrm{i_x} \cdot \mathrm{i_u} & \mathrm{i_x} \cdot \mathrm{j_v} & \mathrm{i_x} \cdot \mathrm{k_w} \\ \mathrm{j_y} \cdot \mathrm{i_u} & \mathrm{j_y} \cdot \mathrm{j_v} & \mathrm{j_y} \cdot \mathrm{k_w} \\ \mathrm{k_z} \cdot \mathrm{i_u} & \mathrm{k_z} \cdot \mathrm{j_v} & \mathrm{k_z} \cdot \mathrm{k_w} \end{bmatrix} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix}$$

- Rotation about x-axis with

$$P_{xyz} = RP_{uvw}$$

$$Rot(x,\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta & -S\theta \\ 0 & S\theta & C\theta \end{bmatrix}$$



Kinematic

Rotation Matrix

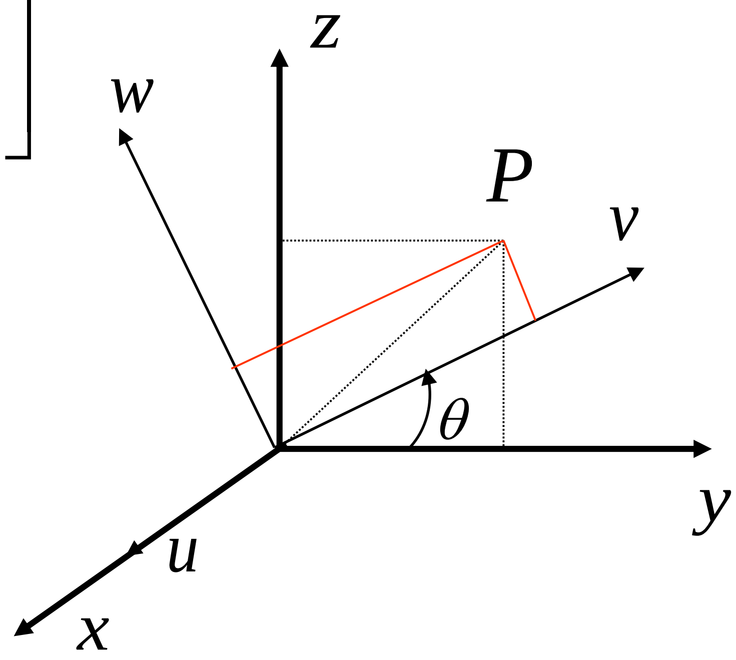
Rotation about x axis with θ

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix}$$

$$p_x = p_u$$

$$p_y = p_v \cos\theta - p_w \sin\theta$$

$$p_z = p_v \sin\theta + p_w \cos\theta$$



Kinematic

Rotation Matrix

Rotation about x-axis with θ

$$Rot(x,\theta)=\begin{bmatrix}1 & 0 & 0\\ 0 & C\theta & -S\theta\\ 0 & S\theta & C\theta\end{bmatrix}$$

Rotation about y-axis with θ

$$Rot(y,\theta)=\begin{bmatrix}C\theta & 0 & S\theta\\ 0 & 1 & 0\\ -S\theta & 0 & C\theta\end{bmatrix}$$

Rotation about z-axis with θ

$$Rot(z,\theta)=\begin{bmatrix}C\theta & -S\theta & 0\\ S\theta & C\theta & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$P_{xyz}=RP_{uvw}$$

Kinematic

Rotation Matrix

$$R = \begin{bmatrix} \mathrm{i_x} \cdot \mathrm{i_u} & \mathrm{i_x} \cdot \mathrm{j_v} & \mathrm{i_x} \cdot \mathrm{k_w} \\ \mathrm{j_y} \cdot \mathrm{i_u} & \mathrm{j_y} \cdot \mathrm{j_v} & \mathrm{j_y} \cdot \mathrm{k_w} \\ \mathrm{k_z} \cdot \mathrm{i_u} & \mathrm{k_z} \cdot \mathrm{j_v} & \mathrm{k_z} \cdot \mathrm{k_w} \end{bmatrix} \quad P_{xyz} = RP_{uvw}$$

Obtain the coordinate of P_{uvw} from the coordinate of P_{xyz}

Dot products are commutative!

$$\begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} = \begin{bmatrix} \mathrm{i_u} \cdot \mathrm{i_x} & \mathrm{i_u} \cdot \mathrm{j_y} & \mathrm{i_u} \cdot \mathrm{k_z} \\ \mathrm{j_v} \cdot \mathrm{i_x} & \mathrm{j_v} \cdot \mathrm{j_y} & \mathrm{j_v} \cdot \mathrm{k_z} \\ \mathrm{k_w} \cdot \mathrm{i_x} & \mathrm{k_w} \cdot \mathrm{j_y} & \mathrm{k_w} \cdot \mathrm{k_z} \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$$

$$\begin{aligned} P_{uvw} &= QP_{xyz} \\ P_{xyz} &= RP_{uvw} \end{aligned} \quad \Rightarrow \quad QR = R^T R = R^{-1} R = I_3$$ <== 3X3 identity matrix

$$Q = R^{-1} = R^T$$

Kinematic

Rotation Matrix

- A point $a_{uvw} = (4,3,2)$ is attached to a rotating frame, the frame rotates 60 degree about the OZ axis of the reference frame. Find the coordinates of the point relative to the reference frame after the rotation.

$$a_{xyz} = Rot(z,60)a_{uvw}$$

$$= \begin{bmatrix} 0.5 & -0.866 & 0 \\ 0.866 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} -0.598 \\ 4.964 \\ 2 \end{bmatrix}$$

Kinematic

Rotation Matrix

Rotation ϕ about OY axis

Rotation θ about OW axis

Rotation α about OU axis

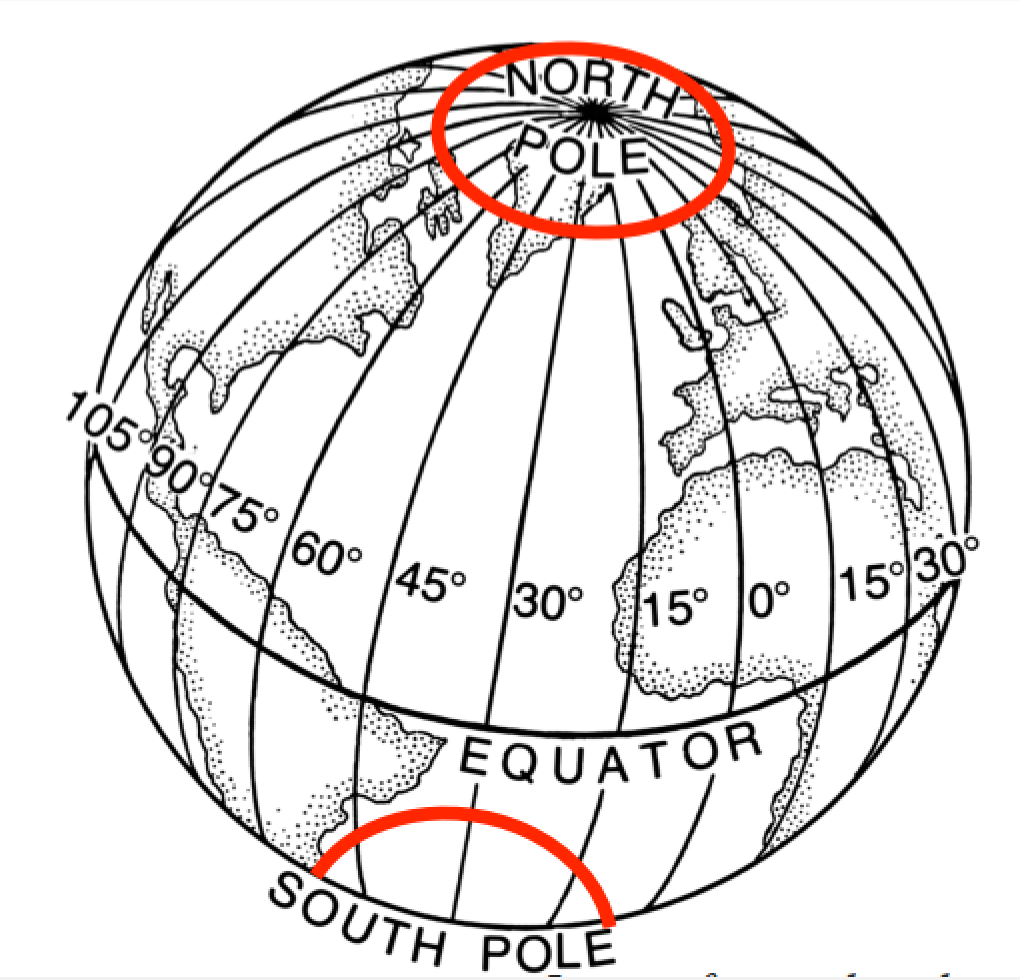
Answer…

$$R = Rot(y,\phi) I_3 Rot(w,\theta) Rot(u,\alpha)$$

$$= \begin{bmatrix} C\phi & 0 & S\phi \\ 0 & 1 & 0 \\ -S\phi & 0 & C\phi \end{bmatrix} \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{bmatrix}$$

$$= \begin{bmatrix} C\phi C\theta & S\phi S\alpha - C\phi S\theta C\alpha & C\phi S\theta S\alpha + S\phi C\alpha \\ S\theta & C\theta C\alpha & -C\theta S\alpha \\ -S\phi C\theta & S\phi S\theta C\alpha + C\phi S\alpha & C\phi C\alpha - S\phi S\theta S\alpha \end{bmatrix}$$

Pre-multiply if rotate about the OXYZ axes
Post-multiply if rotate about the OUVW axes



Ambiguity

Ambiguity

Atan Function

Recall:

$$\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)} = \frac{y}{x}$$

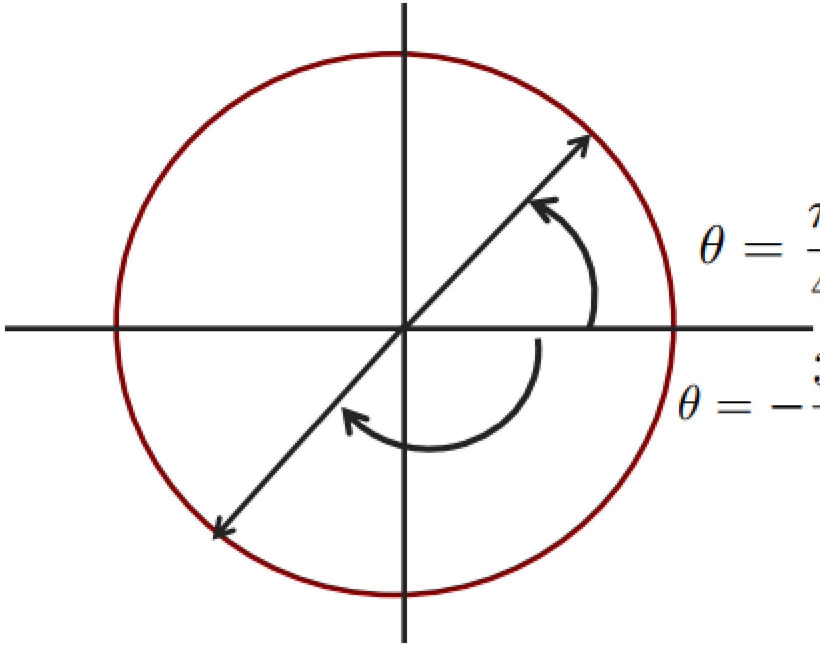
The function $\theta = \tan^{-1}(\frac{y}{x})$ returns the angle θ for which $\tan(\theta) = \frac{y}{x}$.

$$\tan(\frac{\pi}{6}) = \frac{1}{\sqrt{3}} \longrightarrow \tan^{-1}(\frac{1}{\sqrt{3}}) = \frac{\pi}{6}$$

$$\text{atan}(\frac{y}{x}) = \tan^{-1}(\frac{y}{x})$$

Ambiguity

Atan Function



$$\theta = \frac{\pi}{4} \rightarrow \quad \sin(\theta) = \frac{\sqrt{2}}{2} \qquad \cos(\theta) = \frac{\sqrt{2}}{2} \qquad \tan(\theta) = 1$$

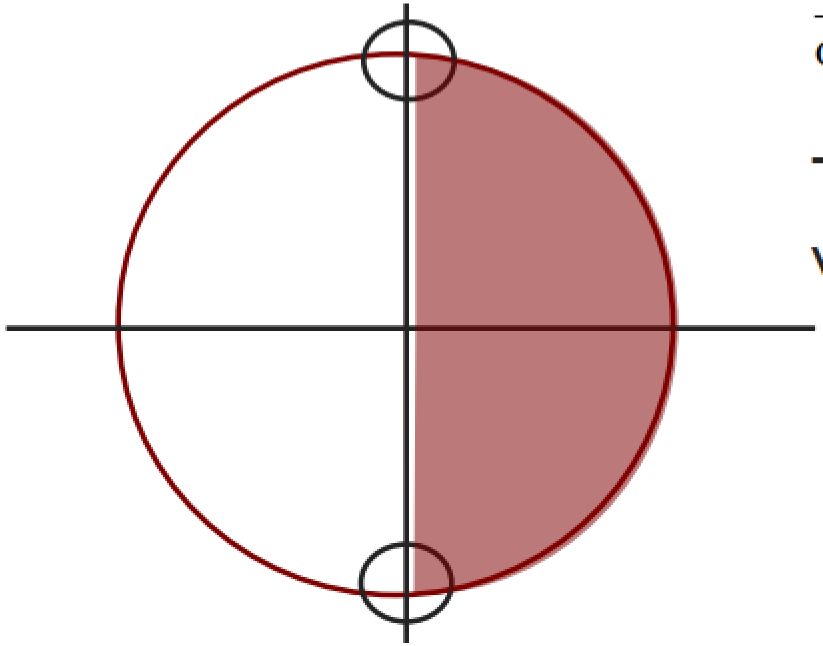
$$\theta = -\frac{3\pi}{4} \rightarrow \quad \sin(\theta) = -\frac{\sqrt{2}}{2} \qquad \cos(\theta) = -\frac{\sqrt{2}}{2} \qquad \tan(\theta) = 1$$

$$\tan^{-1}(\frac{1}{1}) = \tan^{-1}(\frac{-1}{-1})$$

The atan function cannot distinguish between opposite points on the unit circle.

Ambiguity

Atan Function



$$\frac{\sin(\theta)}{\cos(\theta)} = \frac{y}{x} = \frac{\pm 1}{0} = \text{undefined}$$

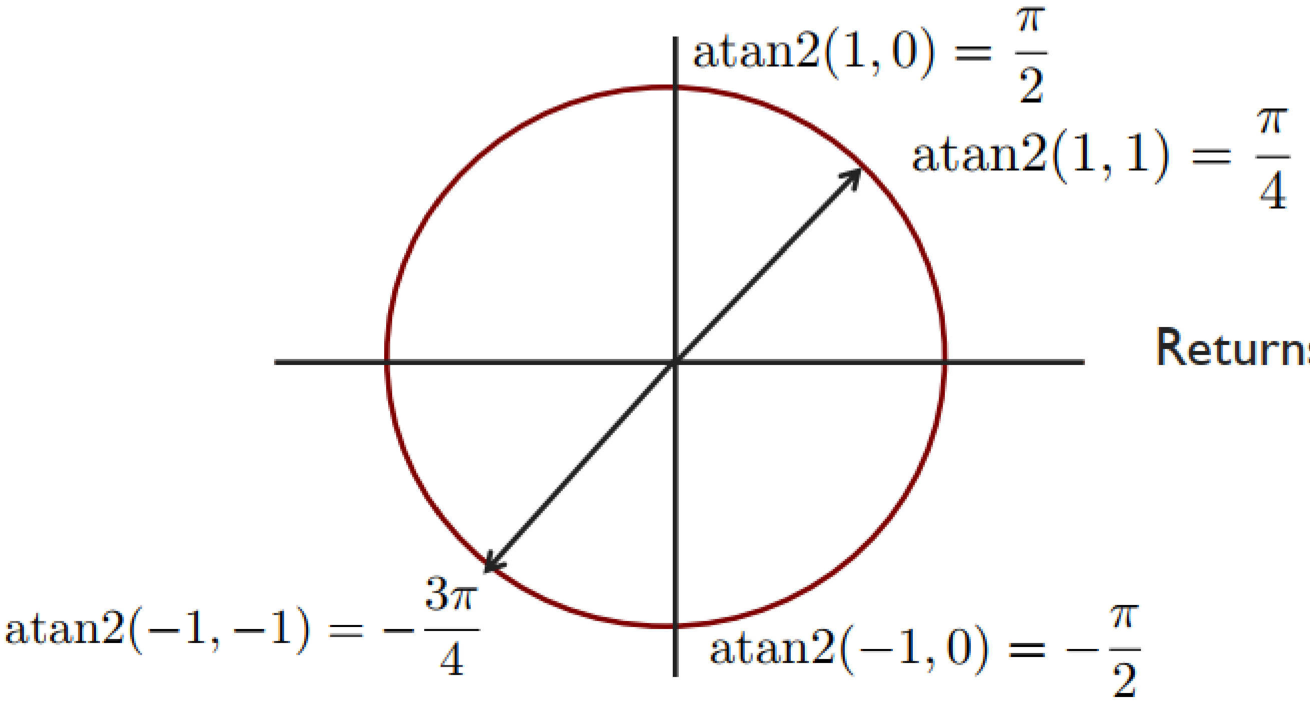
The atan function fails when $\theta = \pm\frac{\pi}{2}$.

Returns values in range $(-\frac{\pi}{2}, \frac{\pi}{2})$

Ambiguity

Atan Function

$\text{atan2}(y, x)$ is an implementation of the atan function that takes into account ratio and the signs of y and x.

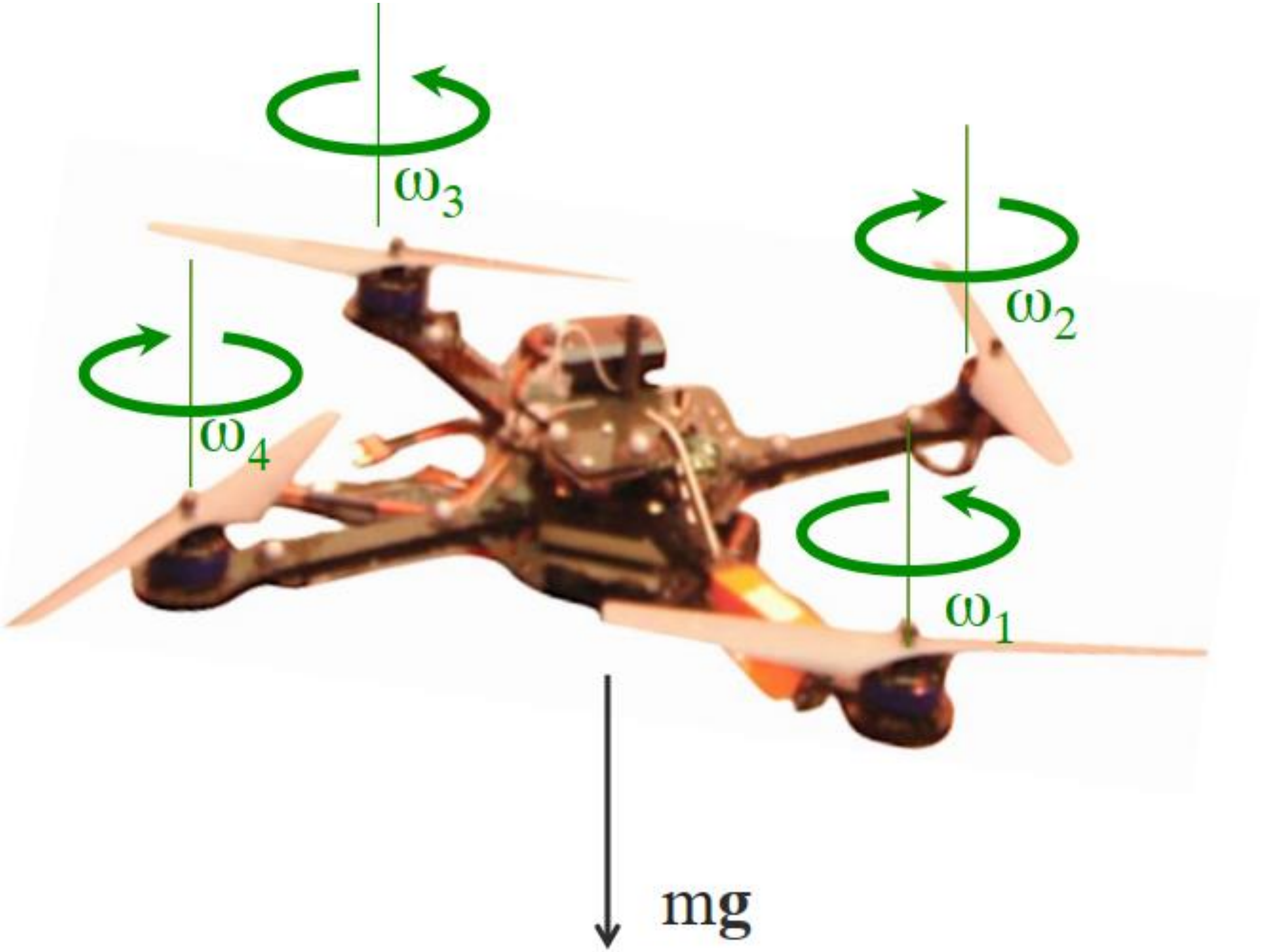


Returns values in range $(-\pi, \pi]$

Control of height

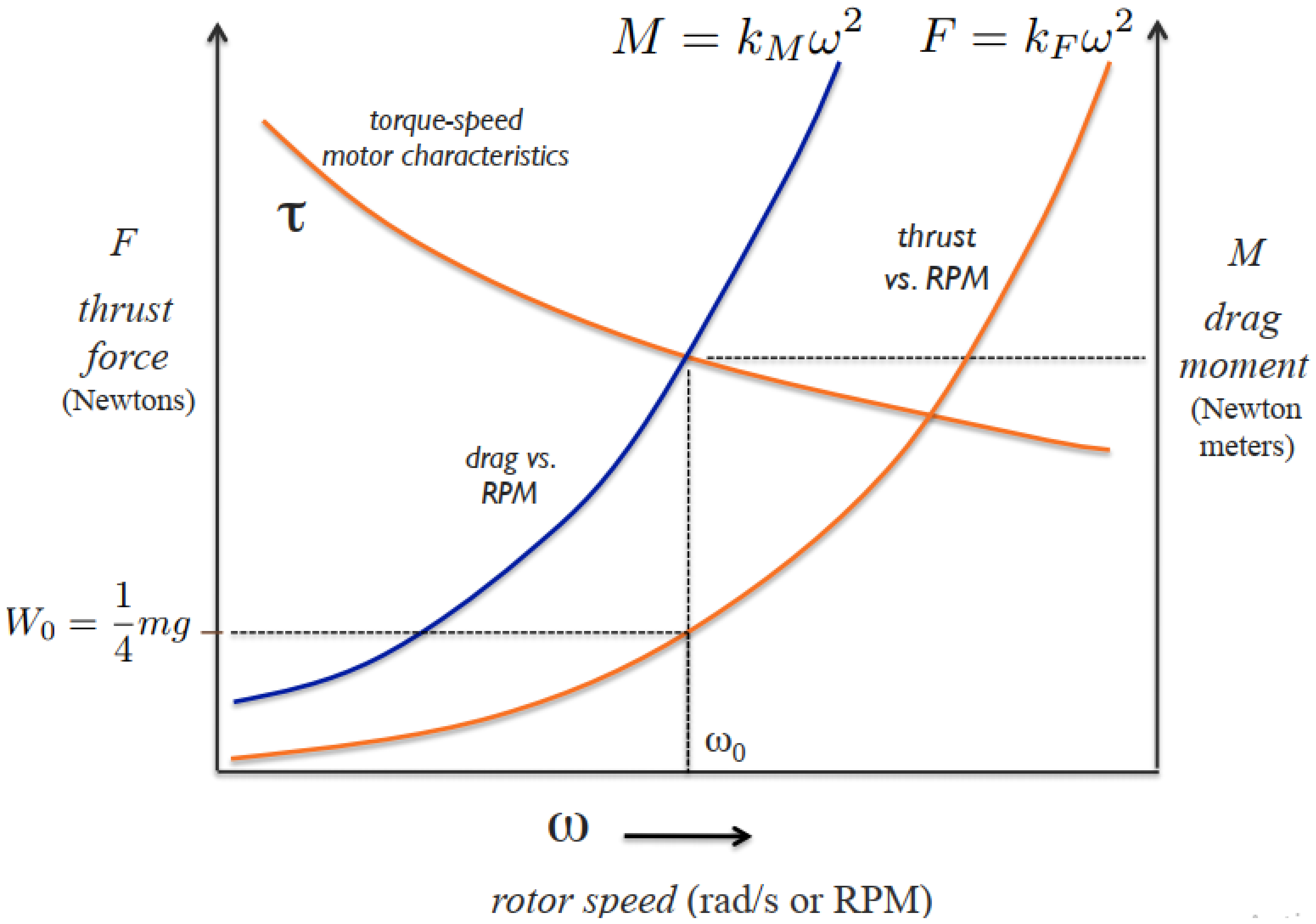
Kinematic

Basic Mechanics



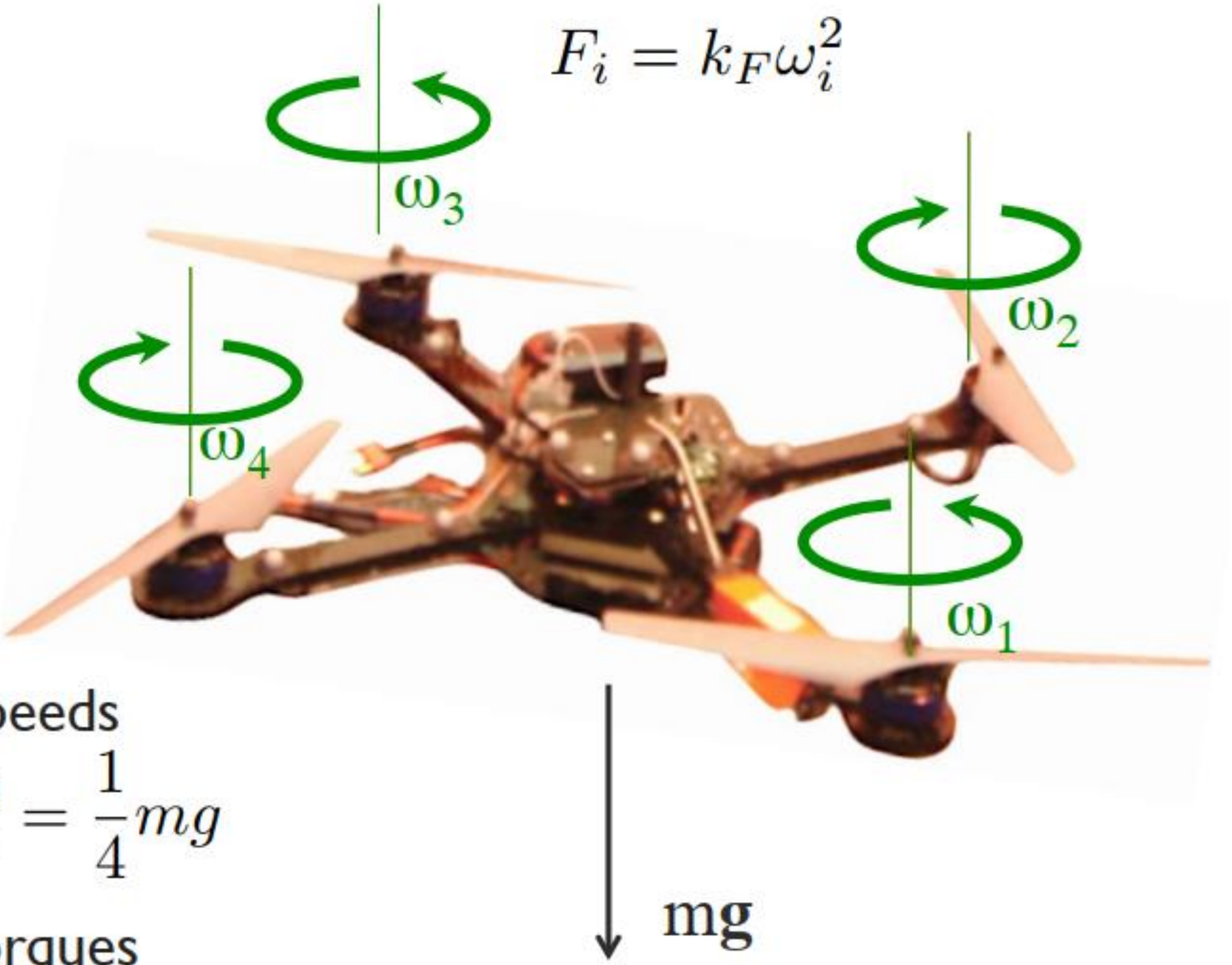
Kinematic

Rotor Physics



Kinematic

Basic Mechanics (Hover)



Motor Speeds

$$k_F \omega_i^2 = \frac{1}{4} mg$$

Motor Torques

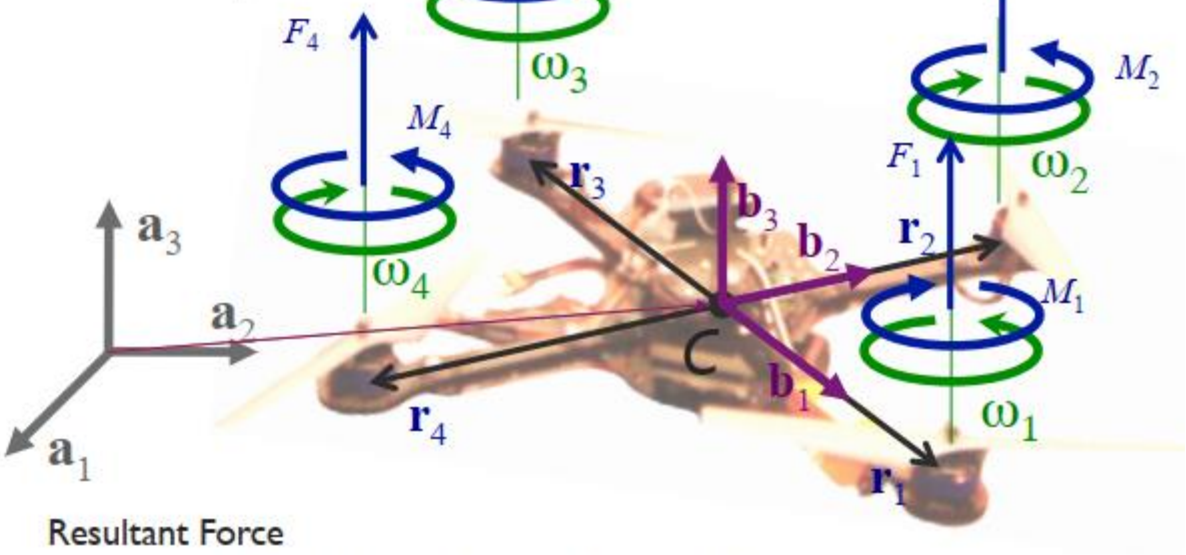
$$\tau_i = k_M \omega_i^2$$

Kinematic

Basic Mechanics (Hover)

$$F_i = k_F \omega_i^2$$

$$M_i = k_M \omega_i^2$$



Resultant Force

$$\mathbf{F} = \mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3 + \mathbf{F}_4 - mg\mathbf{a}_3$$

Resultant Moment

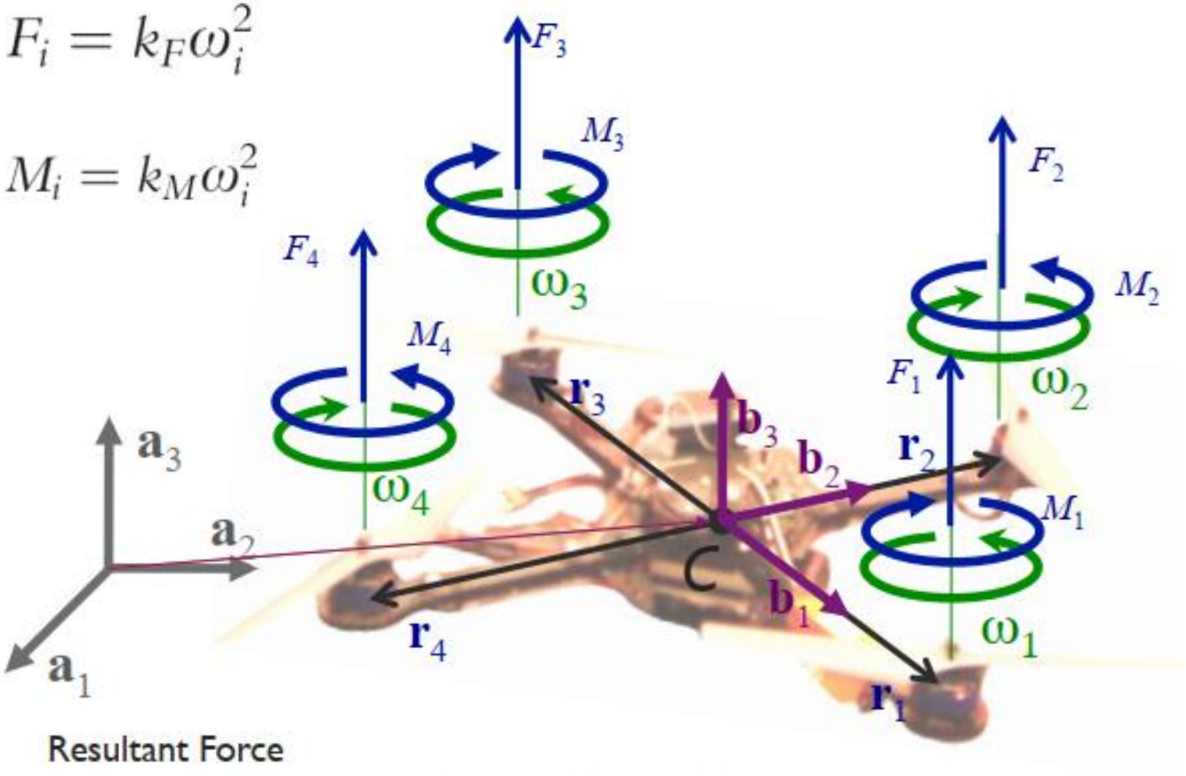
$$\mathbf{M} = \mathbf{r}_1 \times \mathbf{F}_1 + \mathbf{r}_2 \times \mathbf{F}_2 + \mathbf{r}_3 \times \mathbf{F}_3 + \mathbf{r}_4 \times \mathbf{F}_4 + \mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4$$

Kinematic

Basic Mechanics (Hover)

$$F_i = k_F \omega_i^2$$

$$M_i = k_M \omega_i^2$$



Resultant Force

$$\mathbf{F} = \mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3 + \mathbf{F}_4 - mg\mathbf{a}_3$$

Resultant Moment

$$\mathbf{M} = \mathbf{r}_1 \times \mathbf{F}_1 + \mathbf{r}_2 \times \mathbf{F}_2 + \mathbf{r}_3 \times \mathbf{F}_3 + \mathbf{r}_4 \times \mathbf{F}_4 + \mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4$$

Control

Control of altitude

$$\boxed{\sum_{i=1}^{4} k_F \omega_i^2 + m\mathbf{g}} = m\mathbf{a} \longrightarrow a = \frac{d^2x}{dt^2} = \ddot{x}$$

Input $u = \frac{1}{m}\left[\sum_{i=1}^{4} k_F \omega_i^2 + m\mathbf{g}\right]$

Second order dynamic system $u = \ddot{x}$

What input drives the robot to the desired position?

Control

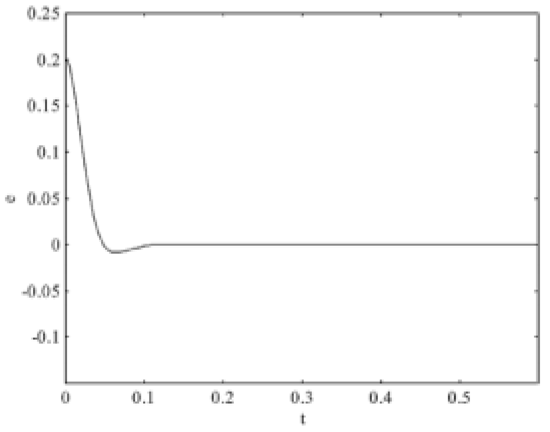
Control of altitude

Problem

State, input $x,\ u\ \in \mathbb{R}$

Plant model $\ddot{x} = u$

Want x to follow the desired trajectory $x^{des}(t)$



General Approach

Define error, $e(t)=x^{des}(t)- x(t)$

Want $e(t)$ to converge exponentially to zero

Strategy

Find u such that

$$\ddot{e} + K_v\dot{e} + K_p e = 0 \qquad K_p,\ K_v\ >\ 0$$

$$u(t) = \underbrace{\ddot{x}^{\mathrm{des}}(t)}_{\text{Feedforward}} + \underbrace{K_v\dot{e}(t)}_{\text{Derivative}} + \underbrace{K_p e(t)}_{\text{Proportional}}$$

Control

Control of altitude

PD control

$$u(t) = \ddot{x}^{\text{des}}(t) + K_v \dot{e}(t) + K_p e(t)$$

Proportional control acts like a spring (capacitance) response

Derivative control is a viscous dashpot (resistance) response

Large derivative gain makes the system overdamped and the system converges slow

PID control

In the presence of disturbances (e.g., wind) or modeling errors (e.g. unknown mass), it is often advantageous to use PID control

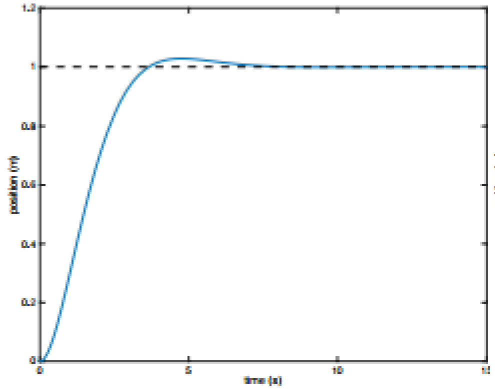
$$u(t) = \ddot{x}^{\text{des}}(t) + K_v \dot{e}(t) + K_p e(t) + K_i \underbrace{\int_0^t e(\tau) d\tau}_{\text{Integral}}$$

PID control generates a third-order closed-loop system

Integral control makes the steady-state error go to zero

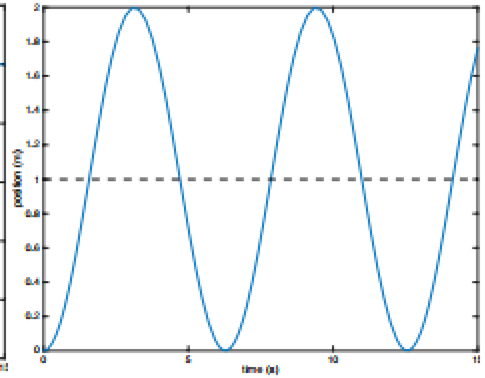
Control

Control of altitude



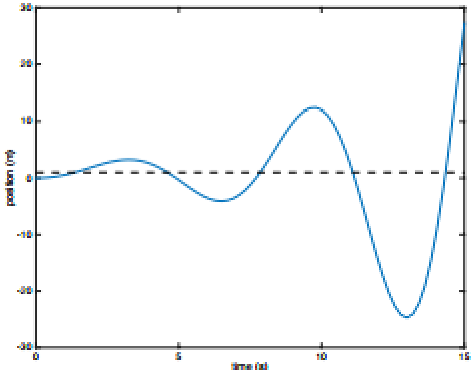
Stable

$$K_p,\ K_v\ >\ 0$$



Marginally Stable

$$K_p > \ 0, \quad K_v\ = 0$$



Unstable

$$K_p \text{ or } K_v\ < 0$$

Q & A