# Quadrotor Modeling, Simulation, and Control[1]

## AE 540 - Flight Dynamics and Control I - 232

Zeyad M. Manaa

Deparment of Aerospace Engineering, KFUPM, Dhahran 31261

11 and 13 February, 2024

[1]Supplmentary material availble here

# Part I - Sunday 11 Feb, 2024

# Modeling Dynamics

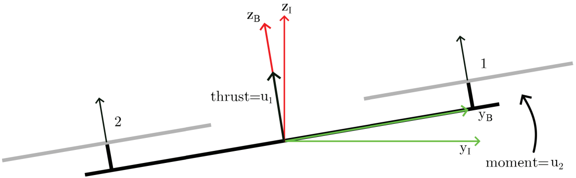


Figure: 3-DOF Quadrotor Configuration[2]

$$
{}^{\mathcal{I}}[\mathcal{R}]_{\mathcal{B}} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix}
$$

$$
\boldsymbol{\omega} = \dot{\phi}
$$

[2] Image source: Z. M. Manaa et. al. *AIAA SCITECH, 2024*

# Modeling Dynamics Contd'

$$m\begin{bmatrix}\ddot{y}\\ \ddot{z}\end{bmatrix} = \begin{bmatrix}0\\ -mg\end{bmatrix} + {}^{\mathcal{I}}[\mathcal{R}]_{\mathcal{B}}\begin{bmatrix}0\\ u_1\end{bmatrix} = \begin{bmatrix}0\\ -mg\end{bmatrix} + \begin{bmatrix}-u_1\sin(\phi)\\ u_1\cos(\phi)\end{bmatrix}$$

and

$$I_{xx}\ddot{\phi} = L(F_1 - F_2) = u_2$$

and

$$\begin{bmatrix}\ddot{y}\\ \ddot{z}\\ \ddot{\phi}\end{bmatrix} = \begin{bmatrix}0\\ -g\\ 0\end{bmatrix} + \begin{bmatrix}-\frac{1}{m}\sin(\phi) & 0\\ \frac{1}{m}\cos(\phi) & 0\\ 0 & \frac{1}{I_{xx}}\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix}$$

## General Objective

The aim is to integrate a first-order ordinary differential equation (ODE) on the form

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(t, \mathbf{x}(t)) \tag{1}$$

over a finite time interval $\Delta t$, to convert them to a difference equation,

$$\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \int_{t}^{t+\Delta t} f(\tau, \mathbf{x}(\tau)) d\tau \tag{2}$$

or alternatively, if we assume that $t_n = n\Delta t$ and $\mathbf{x}_n \triangleq \mathbf{x}_n(t_n)$,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \int_{n\Delta t}^{(n+1)\Delta t} f(\tau, \mathbf{x}(\tau)) d\tau \tag{3}$$

# Runge-Kutta Methods

Utilizing the discrete formulation of the derivative we can write,

$$\frac{d\mathbf{x}(t)}{dt} \approx \frac{\mathbf{x}(t+\Delta t) - \mathbf{x}(t)}{\Delta t} \approx \mathbf{f}(t, \mathbf{x}(t)) \tag{4}$$

The goal is to solve for the next step. So,

$$\mathbf{x}(t+\Delta t) \approx \mathbf{x}(t) + \Delta t \mathbf{f}(t, \mathbf{x}(t)) \tag{5}$$

In a discrete format we can approximate the next step by using the forward difference formula:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t \mathbf{f}(t_n, \mathbf{x}_n) \tag{6}$$

# Part II - Tuesday 13 Feb, 2024

# Taylor Series for General Multivariate Functions

$$f(\mathbf{x}) = f(\mathbf{x})\Big|_{\mathbf{x}=\mathbf{x}_0} + \nabla^\top f(\mathbf{x})\Big|_{\mathbf{x}=\mathbf{x}_0}(\mathbf{x} - \mathbf{x}_0) + \cdots + \epsilon \qquad (7)$$

Where

$$\nabla^\top f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix} \quad \text{and} \quad \epsilon \text{ is a truncation error}$$

- Example question: Expand $f(x_1, x_2)$ around $(1, 2)$ where

$$f(x_1, x_2) = e^{-(x_1^2 + x_2^2)}$$

# Linerization of System of Multivariate Functions

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \approx \mathbf{f}\left(\mathbf{x}_0, \mathbf{u}_0\right) + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0, \mathbf{u}_0} \delta \mathbf{x} + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{x}_0, \mathbf{u}_0} \delta \mathbf{u} \tag{8}$$

$$\delta \mathbf{x} = \mathbf{x} - \mathbf{x}_0, \quad \delta \mathbf{u} = \mathbf{u} - \mathbf{u}_0$$

$$\frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}, \quad \frac{\partial \mathbf{f}}{\partial \mathbf{u}} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \cdots & \frac{\partial f_2}{\partial u_m} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_n}{\partial u_1} & \frac{\partial f_n}{\partial u_2} & \cdots & \frac{\partial f_n}{\partial u_m} \end{bmatrix}$$

# Linerization of System of Multivariate Functions Contd'

$$\dot{\mathbf{x}} - \mathbf{f}\left(\mathbf{x}_0, \mathbf{u}_0\right) \approx \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0, \mathbf{u}_0} \delta\mathbf{x} + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{x}_0, \mathbf{u}_0} \delta\mathbf{u} \tag{9}$$

$$\delta\dot{\mathbf{x}} \approx \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0, \mathbf{u}_0} \delta\mathbf{x} + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{x}_0, \mathbf{u}_0} \delta\mathbf{u} \tag{10}$$

We can reformat this to be,

$$\dot{\mathbf{z}} \approx A\mathbf{z} + B\mathbf{w} \tag{11}$$

where,

$$A = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0, \mathbf{u}_0}, \quad B = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{x}_0, \mathbf{u}_0}, \quad \mathbf{z} = \delta\mathbf{x}, \quad \mathbf{w} = \delta\mathbf{u}$$

# Nominal Dynamics Revisited and Linearized

$$\begin{aligned}
\ddot{y} &= -\frac{u_1 \sin(\phi)}{m} = f_1(y, z, \phi, \dot{x}, \dot{y}, \dot{\phi}, u_1, u_2, g) \\
\ddot{z} &= -g + \frac{u_1 \cos(\phi)}{m} = f_2(y, z, \phi, \dot{x}, \dot{y}, \dot{\phi}, u_1, u_2, g) \\
\ddot{\phi} &= \frac{u_2}{J} = f_3(y, z, \phi, \dot{x}, \dot{y}, \dot{\phi}, u_1, u_2, g)
\end{aligned} \tag{12}$$

To linearize the $\ddot{y}$ we do,

$$\begin{aligned}
\delta\ddot{y} = & \left.\frac{\partial f_1}{\partial y}\right|_{\mathbf{x}=\mathbf{x}_0} \delta y + \left.\frac{\partial f_1}{\partial z}\right|_{\mathbf{x}=\mathbf{x}_0} \delta z + \left.\frac{\partial f_1}{\partial \phi}\right|_{\mathbf{x}=\mathbf{x}_0} \delta\phi + \left.\frac{\partial f_1}{\partial \dot{y}}\right|_{\mathbf{x}=\mathbf{x}_0} \delta\dot{x} \\
& + \left.\frac{\partial f_1}{\partial \dot{z}}\right|_{\mathbf{x}=\mathbf{x}_0} \delta\dot{y} + \left.\frac{\partial f_1}{\partial \dot{\phi}}\right|_{\mathbf{x}=\mathbf{x}_0} \delta\dot{\phi}
\end{aligned} \tag{13}$$

# Nominal Dynamics Revisited and Linearized Contd

$$\begin{aligned} \delta\ddot{y} &= -9.8 \times \delta\phi \\ \delta\ddot{z} &= -\delta g + \frac{\delta u_2}{m} \\ \delta\ddot{\phi} &= \frac{\delta u_2}{J} \end{aligned} \tag{14}$$

# Error Dynamics

**Problem**

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

Plant model $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$

Find a control input function $u(t)$ so that $x(t)$ follows the desired trajectory $\mathbf{x}^{\text{des}}(t)$

**General Approach**

Define error,

$$\mathbf{e}(t) = \mathbf{x}^{\text{des}}(t) - \mathbf{x}(t)$$

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

**Strategy**

Find $u(t)$ such that

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

# Trajectory tracking

Discussion

# Control Laws Following PD Controller

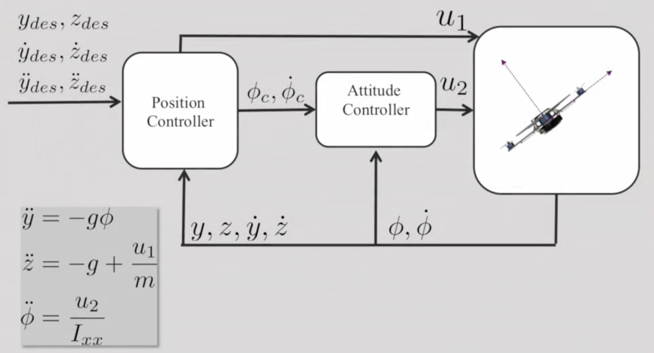


Figure: Control architecture[3]

[3]Image Sourse: GRASP Lab

# Control Laws Following PD Controller Contd

$$
\begin{aligned}
u_1 &= m\left[g - k_{v,z}\dot{z} + k_{p,z}\left(z_0 - z\right)\right] \\
u_2 &= I_{xx}\left(\ddot{\phi}_c + k_{v,\phi}\left(\dot{\phi}_c - \dot{\phi}\right) + k_{p,\phi}\left(\phi_c - \phi\right)\right) \\
\phi_c &= -\frac{1}{g}\left(k_{v,y}(-\dot{y}) + k_{p,y}\left(y_0 - y\right)\right)
\end{aligned} \tag{15}
$$

# Demonstration

Questions?