Contents

Lab 3 Notes

Quadcopter Basics

- Type:
  - rotorcraft
- Composed of:
  - 4 propellers arranged in a square formation
- Propeller:
  - driven by an individual electric motor "actuator"
  - Motor:
    - o controlled by FCU "Flight Control Unit"
    - o FCU: responsible for translating user input into aerial flight
    - o FCU uses: information from the vehicle's onboard sensors & connected motors using pre-programmed flight controllers and temporal logic
- Battery:
  - powers motors and FCU

Coordinate system of the Quadcopter

- Body reference frame
  - follows the quadcopter
- inertial reference frame
  - describes the space in which the quadcopter moves
    - o aka "our location as we watch the quadcopter"

Degrees of freedom:

- "this is all the ways it can move"
- 6 total
- Translational (3 Degrees):
  - Variables of the body: x_b, y_b, z_b
  - up,down,left,right
  - aka: x-direction (forward/back), y-direction (left/right), z-direction (up/down)
- Rotational (3 Degrees):
  - "rotating about one of the axes"
  - Variables of rotation: ψ, θ, ϕ
    - o called: psi, theta, phi

o aka "yaw, pitch, roll"

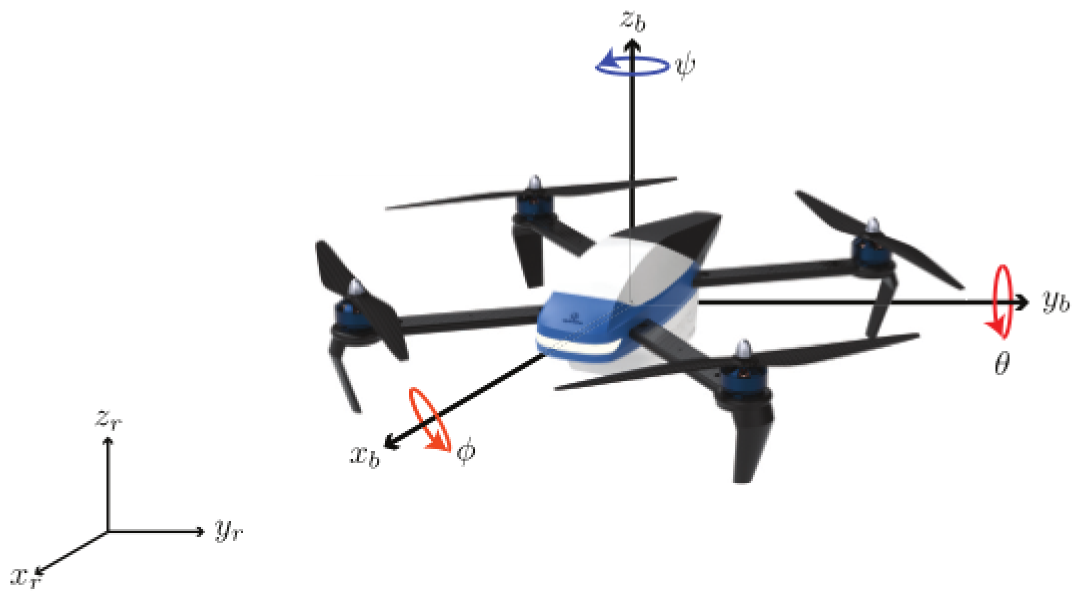

Figure 2: Quadcopter coordinate frame and state variables.

Figure 1: Subscripts r: inertial frame. Subscripts b: body frame

Sum of Translational Forces

- In the up-vertical direction: > .

$$F_i = k_F \omega_i^2 \quad i \in \{1, 2, 3, 4\}$$

  > .
  - Where:
    - o F_i is the force on the propeller
    - o ω_i is the rotational speed
    - o $\frac{d}{dt}$ of an angle
    - o k_F is a constant parameter
      - · it accounts for a number of physical parameters relating to the motor torques and aerodynamics
    - o Derivations not provided
  - So the sum of all is:

$$F_{up} = F_1 + F_2 + F_3 + F_4$$

- In the down-vertical direction:

$$F_{down} = -mg$$

  - where:
    - o m = mass of the quadcopter
    - o g = acceleration due to gravity
    - o mg = weight of the quadcopter
- To have the quadcopter hover in place the force up must equal the force down

$$|F_{up}| = |F_{down}|$$

$$|F_1 + F_2 + F_3 + F_4| = |F_{down}|$$

$$|F_1 + F_2 + F_3 + F_4| = |-mg|$$

> Aka:

$$\sum F_y = F_{up} + F_{down} = 0 \implies F_{up} = -F_{down}$$

> .

Sum of Rotational Forces

- Each propeller is moving in a circular manner and causing a rotational force
- Also known as moment of torque τ_{Mi} " torque moment "
  - This is the tendency of the quadcopter to spin about the main body vertical axis z_b by some angle yaw ψ > .

$$\tau_{Mi} = k_M \omega_i^2 \quad i \in \{1, 2, 3, 4\}$$

  > .
  - where k_M is a constant parameter, ignore for now
- How the propellers work:
  - adjacent (next to): spin in opposite directions
  - opposite ends: spin in the same direction
  - To hover the opposite ends need to counter act one another:

$$\tau_{M1} - \tau_{M2} + \tau_{M3} - \tau_{M4} = 0$$

  - where $\tau_{M1} = -\tau_{M2}$ and $\tau_{M2} = -\tau_{M4}$

1D Quadcopter Dynamics

`IMPORTANT: None of the previous theory is used in the coding of this because we use approximations (Forward Euler)`

- Movement up and down (only)
- We focus on forces in the z direction (up/down) and ignore rotation/torque
- Sum of forces tells us:

$$\sum F_{vertical} = m \cdot a_{vertical}$$

- Using what we defined for Hover:

$$F_{up} - F_{down} = m \cdot a_{vertical} = 0$$

- let vertical = z direction and sub in what we know

$$(F_1 + F_2 + F_3 + F_4) - mg = m \cdot a_z = F_z$$

> .

$$\sum_{i=1}^{4} F_i - mg = m \cdot a_z$$

> .

- where F_z is the net force in the z-direction
- assume: $F_1 = F_2 = F_3 = F_4 = F$ so

$$\sum_{i=1}^{4} F_i = 4F$$

Construct the ODE

- since $a_z = \frac{d^2 z}{dt^2} = \frac{dv_z}{dt}$ we can say,

$$\sum_{i=1}^{4} F_i - mg = m \cdot a_z$$

$$4F - mg = m \cdot \frac{dv_z}{dt}$$

- solving for the differential

$$\frac{dv_z}{dt} = \frac{1}{m} \cdot 4F - \frac{1}{m} \cdot mg$$
$$\frac{dv_z}{dt} = \frac{1}{m} \cdot 4F - g$$

Final equations for a 1D Copter

.

1. Velocity ODE
$$\frac{dv_z}{dt} = \frac{1}{m} \cdot 4F - g$$
2. Acceleration ODE
$$\frac{dv_z}{dt} = a_z = \frac{d^2z}{dt^2}$$
3. Sum of forces in z-direction
$$4F - mg = m \cdot a_z = F_z = F_{net}$$

.

Describe Hover, Ascent, Descent

- Hover:
  - Sum of forces in z-direction $F_z = F_{net} = 0$
$$4F - mg = m \cdot a_z = 0F_{net} = 4F - mg = m \cdot \frac{dv_z}{dt} = 0$$
$$\frac{dv_z}{dt} = \frac{F_{net}}{m} = \frac{0}{m}$$
  - The velocity ODE for this:
$$\frac{dv_z}{dt} = \frac{1}{m} \cdot 4F - g = 0$$
- Ascending:
  - Sum of forces in z-direction is positive $F_z = F_{net} > 0$
$$F_{net} = 4F - mg = m \cdot a_z > 0$$
$$\frac{dv_z}{dt} = \frac{F_{net} > 0}{m} > 0$$
  - The velocity ODE for this:
$$\frac{dv_z}{dt} = \frac{1}{m} \cdot 4F - g > 0$$
- Descending:
  - Sum of forces in z-direction is negative $F_z = F_{net} < 0$
$$F_{net} = 4F - mg = m \cdot a_z < 0$$
$$\frac{dv_z}{dt} = \frac{F_{net} < 0}{m} < 0$$
  - The velocity ODE for this:
$$\frac{dv_z}{dt} = \frac{1}{m} \cdot 4F - g < 0$$

Quadcopter Dynamics 2D

- Moving in the plane
- Requires 3 degrees of freedom
- Requires y-direction and rotation about y-z plane ϕ
- Equations of (y, z, ϕ): given not proved

$$F_y = ma_y = -\sum_{i=1}^{4} F_i \cdot \sin(\phi)$$

$$F_z = ma_z = -mg + \sum_{i=1}^{4} F_i \cdot \cos(\phi)$$

$$\tau_\phi = I_{xx} a_\phi = M_\phi$$

- Their ODE's:

$$\frac{dv_y}{dt} = -\frac{1}{m} \sum_{i=1}^{4} F_i \cdot \sin(\phi)$$

$$\frac{dv_z}{dt} = -g + \frac{1}{m} \sum_{i=1}^{4} F_i \cdot \cos(\phi)$$

$$\frac{dv_\phi}{dt} = \frac{M_\phi}{I_{xx}}$$

- Let $I_{xx} > 0$ some positive value and $M_\phi = 0$ to represent zero net torque on the system.

TA Email

- Problem 1 Requires Equation (5)

1. Velocity ODE

$$\frac{dv_z}{dt} = \frac{1}{m} \cdot \sum_{i=1}^{4} F_i - g$$

- Just pick random positive parameters for I and M
  - Q: but isn't I a tensor? are we just assigning it a random value? and then for the torque similarly, are you saying these values are just constants?
- How many degrees does 2D model have?
  - from the reading it has 3 degrees y,z, phi but again, these values of phi

MATLAB Formatting

`IMPORTANT: This example is in`x`but we use` z `and` y

Given example: used to extend to our models

$$\frac{dx}{dt} = f(t, x) \quad x(t_0) = x_0$$

Set the simulation horizon $[t_0, t_f]$

- Example Model:

$$ma = F$$

  ODEs:

$$\frac{dv_x(t)}{dt} = \frac{F(t)}{m} = a_x(t) \tag{1}$$

$$\frac{dx(t)}{dt} = v_x(t) \tag{2}$$

Convert to Matlab:

  First Matrix to make

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x(t) \\ v_x(t) \end{bmatrix}$$

  $X = [\quad x[1]; \quad x[2] \quad] = [\quad x(t); \quad v_x(t) \quad]$

ODEs:

  Function Resulting Matrix

$$\dot{X} = \begin{bmatrix} \frac{dx}{dt} \\ \frac{dv_x}{dt} \end{bmatrix} = \begin{bmatrix} v_x(t) \\ a_x(t) \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{F}{m} \end{bmatrix} = \begin{bmatrix} x_2 \\ a_x \end{bmatrix}$$

  $\dot{X} = [\quad x[1]; \quad x[2] \quad] = [\quad x_2; \quad \frac{F}{m} \quad]$

- Return an input v_x and a_x

Forward Euler

- Given Equations 1 & 2 we get: (used in the for loop)

$$x * k + 1 = x * k + \Delta v * x = x[1] \tag{3}$$

$$v * k + 1 = v\_k + \Delta a_x = x[2] \tag{4}$$

    From : $X = [x[1];   x[2]   ] $

- Then we plot the components of the vector X

MATLAB for our HW

Model for 1D: 1 Degree of Freedom (z)

Find: Acceleration in z: a_z

- In general:

$$a * zm = F_z = F * net$$
$$a_z = \frac{1}{m} \cdot F_z$$

$$\frac{dv_z}{dt} = \frac{1}{m} \cdot F_z$$

- 1D quadcopter acceleration

$$\frac{dv_z}{dt} = \frac{1}{m} \cdot \sum_{i=1}^{4} F_i - g$$

- Since

$$\begin{aligned} F * z &= \sum *i = 1^4 F_i - g \\ &= 4F - g \end{aligned} \tag{5}$$

- We get: (what we change in `myODE.m`)

$$a_z = \frac{1}{m}(4F - g) \tag{6}$$

Next we consider the 1D cases for Hover, Ascend, Descend using equation 5.

`IMPORTANT: g = 9.81 not -9.81 the negative was already accounted for`

- Hover
  - $F_{net} = F_z = 0 \implies 4F - g = 0$
  - $F = g/4$
- Ascend
  - $F_{net} = F_z > 0 \implies 4F - g > 0$
  - $F > g/4$
- Descend
  - $F_{net} = F_z < 0 \implies 4F - g < 0$
  - $F < g/4$

Conclusion for 1D:

- Make sure the vector X becomes a 2 row matrix

```
x = [0,z_2,....] -> z(t)    --> x[1]
    [0,v_2,....] -> v_z(t)  --> x[2]
```

- Change the `myODE.m` to have the correct a_z

```
Takes: x,F
Uses: F
Return
dx/dt = [..v_z(t)..] -> x[2] --> dx/dt[1]
        [--a_z(t)..]         --> dx/dt[2]
```

- Change the `myODE.m` to take F as an input and call it 3 times for the different conditions
  - Hover: $F = 9.81/4$
  - Ascend: $F = 9.81/4 + 1$
  - Descend: $F = 9.81/4 - 1$

2D Model : 3 Degrees of freedom (z, y, ϕ)

First we create our X vector and i_c vector

```
i_c = [0]
      [0]
      [0]
      [0]
      [0]
      [0]
```

```
x = [y1,      y2,  ...] -> y(t)      --> x[1]
    [vy1,    vy2,  ...] -> v_y(t)    --> x[2]
    [z1,      z2,  ...] -> z(t)      --> x[3]
    [vz1,    vz2,  ...] -> v_z(t)    --> x[4]
    [vphi1,vphi2,  ...] -> phi(t)    --> x[5]
    [phi1,  phi2,  ...] -> v_phi(t)  --> x[6]
```

- Where dx/dt function returns:

```
dx/dt = [..v_y(t)  ..]-> x[2] --> dx/dt[1]
        [..a_y(t)  ..]        --> dx/dt[2]
        [..v_z(t)  ..]-> x[4] --> dx/dt[3]
        [..a_z(t)  ..]        --> dx/dt[4]
        [..v_phi(t)..]-> x[6] --> dx/dt[5]
        [..a_phi(t)..]        --> dx/dt[6]
```

- dx/dt is used in the for loop for the Forward euler for the given equations:

`RECALL:` $\sum_{i=1}^{4} F_i$`= 4F but this has to be specific to each direction`

$$\frac{dv_y}{dt} = -\frac{1}{m}4F_y \cdot \sin(\phi) = a_y$$

$$\frac{dv_z}{dt} = -g + \frac{1}{m}4F_z \cdot \cos(\phi) = a_z$$

$$\frac{dv_\phi}{dt} = \frac{M_\phi}{I_{xx}} = a_\phi$$

$$y * k + 1 = y * k + \Delta v * y = x[1] \tag{7}$$

$$z * k + 1 = z * k + \Delta v_z = x[2] \tag{8}$$

$$\phi * k + 1 = \phi * k + \Delta a * \phi = x[3] \tag{9}$$

Next we need to change/create `myODE2D.m` and adjust the acceleration value

- similar to `myODE.m`
- `myODE(t,x,phi,F)`
- We need to consider the Forces needed for Hover:

In y:

$$\frac{dv_y}{dt} = -\frac{1}{m}4F_y \cdot \sin(\phi) = a_y$$

To get $a_y = 0$ we let

$$F_y = 0$$

In z:

$$\frac{dv_z}{dt} = -g + \frac{1}{m} 4F_z \cdot \cos(\phi) = a_z$$

To get $a_z = 0$ we let

$$-g + \frac{1}{m} 4F_z \cdot \cos(\phi) = 0$$

$$F_z = g \cdot m \frac{1}{4\cos\phi} \quad \cos\phi \neq 0$$

Since we are hovering $a_\phi = 0 \implies M_\phi = 0$

$$\frac{dv_\phi}{dt} = \frac{M_\phi}{I_{xx}} = a_\phi$$

`My only problem is reconciling the cosine term.`

Conclusions for 2D:

Vectors:

```
i_c = [0]
      [0]
      [0]
      [0]
      [0]
      [0]
```

```
x = [y1,      y2,  ...] -> y(t)      --> x[1]
    [vy1,    vy2,  ...] -> v_y(t)    --> x[2]
    [z1,      z2,  ...] -> z(t)      --> x[3]
    [vz1,    vz2,  ...] -> v_z(t)    --> x[4]
    [vphi1,vphi2,  ...] -> phi(t)    --> x[5]
    [phi1,  phi2,  ...] -> v_phi(t)  --> x[6]
```

```
myODE2D Returns:
dx/dt = [..v_y(t)  ..]-> x[2] --> dx/dt[1]
        [..a_y(t)  ..]        --> dx/dt[2]
        [..v_z(t)  ..]-> x[4] --> dx/dt[3]
        [..a_z(t)  ..]        --> dx/dt[4]
        [..v_phi(t)..]-> x[6] --> dx/dt[5]
        [..a_phi(t)..]        --> dx/dt[6]
```

dx/dt function:

```
Takes: t,x,fy,fz,M
Uses: fy,fz,M to calculate a_z,a_y,a_phi
Return: dx/dt
```