

ASEN 3128 LAB 3: Quadrotor Control

- Assigned: Friday 19 February
- Due: Friday 5 March, before start of lab section

For all assignments students will work in groups of two to four students. Groups are determined by the instructors. A single assignment should be submitted for the group.

OBJECTIVES

- Simulate and compare the nonlinear and linear quadrotor EOM.
- Implement feedback control

BACKGROUND

Calculating motor forces from control force and moments

In this lab assignment students will implement feedback control laws from aircraft state variables to the control moments L_c , M_c , and N_c . While it is straightforward to implement such a control law in simulation, in practice the motor thrust forces f_1 , f_2 , f_3 , and f_4 are what must be passed to the quadrotor actuators. Therefore we need to be able to calculate the motor control thrusts from the desired control forces and moments. Recall the following equations for the control force and moments

$$Z_c = -f_1 - f_2 - f_3 - f_4$$

$$L_c = \frac{R}{\sqrt{2}}(-f_1 - f_2 + f_3 + f_4)$$

$$M_c = \frac{R}{\sqrt{2}}(f_1 - f_2 - f_3 + f_4)$$

$$N_c = k_m(f_1 - f_2 + f_3 - f_4).$$

These equations are all linear in the motor thrust forces and can be rewritten in matrix form as:

$$\begin{bmatrix} Z_c \\ L_c \\ M_c \\ N_c \end{bmatrix} = \begin{bmatrix} -1 & -1 & -1 & -1 \\ \frac{R}{\sqrt{2}} & -\frac{R}{\sqrt{2}} & \frac{R}{\sqrt{2}} & \frac{R}{\sqrt{2}} \\ \frac{R}{\sqrt{2}} & \frac{R}{\sqrt{2}} & -\frac{R}{\sqrt{2}} & \frac{R}{\sqrt{2}} \\ k_m & -k_m & k_m & -k_m \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}.$$

The 4x4 matrix is constant and invertible, allowing us to solve for the motor thrust forces when given the control forces and moments:

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} = \begin{bmatrix} -1 & -1 & -1 & -1 \\ \frac{R}{\sqrt{2}} & -\frac{R}{\sqrt{2}} & \frac{R}{\sqrt{2}} & \frac{R}{\sqrt{2}} \\ \frac{R}{\sqrt{2}} & \frac{R}{\sqrt{2}} & -\frac{R}{\sqrt{2}} & \frac{R}{\sqrt{2}} \\ k_m & -k_m & k_m & -k_m \end{bmatrix}^{-1} \begin{bmatrix} Z_c \\ L_c \\ M_c \\ N_c \end{bmatrix}.$$

Quadrotor parameters

These are the parameters for the Parrot Mambo Minidrone:

Quadrotor mass, m :	0.068 kg
Radial distance from CG to propeller, R :	0.060 m
Control moment coefficient, k_m :	0.0024 N*m/(N)
Body x-axis Moment of Inertia, I_x :	6.8E-5 kg*m ²
Body y-axis Moment of Inertia, I_y :	9.2E-5 kg*m ²
Body z-axis Moment of Inertia, I_z :	1.35E-4 kg*m ²
Aerodynamic force coefficient v :	1E-3 N/(m/s) ²
Aerodynamic moment coefficient μ :	2E-6 N*m/(rad/s) ²

PROBLEMS

1. Write a function to plot the full aircraft states and control inputs that result from a simulation run:

function PlotAircraftSim(time, aircraft_state_array, control_input_array, fig, col)

This function is used to plot the results of a complete simulation once it is completed. It takes as input the length n vector which holds the time corresponding to the n th set of variables, the 12 by n array of aircraft state, the 4 by n array of control inputs, a vector of figure numbers, and the string `col` which indicates the plotting option used for every plot, e.g. `col = 'b-'`.

The function should create a total of 6 figures. There should be four figures each with three subplots for the position, Euler angles, inertial velocity, and angular velocity, respectively. There should be one figure with four subplots for each control input variable. Finally, there should be one figure that shows the three-dimensional path of the aircraft, with positive height upward in the figure. This figure should indicate the start (green) and finish (red) of the path with different colored markers.

The function is intended to plot both the linearized and the nonlinear responses to the same initial condition deviation, **not to plot multiple deviation responses at once.**

The function must be able to be called multiple times for different simulation runs with different col indicators. Thus, the function should call each figure before plotting and include the hold on command. However, it should not clear the figures before plotting.

For example

```
figure(fig(1));
subplot(311);
plot(time, aircraft_state_array(1,:), col); hold on;
subplot(312);
plot(time, aircraft_state_array(2,:), col); hold on;
...
```

2. Write a function to calculate the motor thrust forces given the control force and moments:

```
function motor_forces = ComputeMotorForces(Zc, Lc, Mc, Nc, R, km)
```

where `motor_forces` is the column vector $[f_1, f_2, f_3, f_4]^T$.

3. Simulate the response of the **non-linear quadrotor equations of motion** to initial condition deviations from the steady hover trim state as follows:
 - a. Deviation by +5deg in roll
 - b. Deviation by +5deg in pitch
 - c. Deviation by +5 deg in yaw
 - d. Deviation by +0.1 rad/sec in roll rate
 - e. Deviation by +0.1 rad/sec in pitch rate
 - f. Deviation by +0.1 rad/sec in yaw rate

Plot and discuss the resulting behavior. Does it make physical sense? Is the steady hover a stable flight condition? For this problem, call `PlotAircraftSim` once per deviation.

4. Develop a **linearized model of the full quadrotor dynamics** about a steady hover trim state. Repeat Problem 3) using the linearized dynamics model, and compare linearized

and non-linearized behaviors on the **same** plots. Note, the `initial.m` function in Matlab is useful for this problem.

5. Add a feedback control in your non-linear simulation such that the control moments about each body axis are proportional to the rotational rates about their respective axes, but in opposition to the sign of angular velocity with a gain of $0.004 \text{ Nm/(rad/sec)}$. The control force in the body z-direction should still equal the weight of the quadrotor. Repeat problems 3) d. through 3) f. What is the effect of this control law? Calculate and plot the motor thrust forces that result from this feedback control approach.

ASSIGNMENT

This assignment does not require a full lab report. Instead, submit **a single PDF file** that includes answers to all questions with plots generated for the problems and the text of any code created for the assignment. Only a single example of the code is expected even if students made multiple versions (e.g. each student wrote their own).

Be sure to add titles to all figures that include both the problem number and a description of the plot, e.g. 2.c Output x versus Time. The assignment will be evaluated based on i.) correct answers; ii.) proper commenting and documenting of code; and iii.) the quality of the figures submitted (e.g. labeling, axis, etc).

All lab assignments should include the Team Participation table and should be completed and acknowledged by all team members. Description of the Team Participation table is provided in a separate document.