ASEN 3801 | Lab 04 Task 1 Main

Creators: Natsumi Kakuda Assignment Due Date: 11/10/2025

```
clc;
clear;
close all;
```

Toggling Plots (1 on, 0 off)

```
P1_2 = 0; % Part 1.2
P1_3 = 0; % Part 1.3
P1_4 = 0; % Part 1.4
P1_5 = 1; % Part 1.5
```

Initialization of Data

```
% Initialization / Figure
fig1 = [1,2,3,4,5,6]; % Part 1.2
fig2 = [7,8,9,10,11,12]; % Part 1.3
fig3 = [13,14,15,16,17,18]; % Part 1.4
fig4 = [19,20,21,22,23,24]; % Part 1.4
fig5 = [25,26,27,28,29,30]; % Part 1.5
col1 = '-b'; % Part 1.2
col2 = '--b'; % Part 1.3
col3 = '-k'; % Part 1.4
col4 = '--k'; % Part 1.4
col5 = '-m'; % Part 1.5

% Givens
t = 10; % Period [s]
m = 0.068; % kg
g = 9.81; % m/s^2
I = diag([5.8e-5 7.2e-5 1.0e-4]); % kg*m^2
km = 0.0024; % N*m/(N)
vu = 1e-3; % N/m/s)^2
mu = 2e-6; % N*m/(rad/s)^2
d = 0.060; % m

% Hover Conditioning
f_i = (m*g)/4; % Zc = -(f1+f2+f3+f4) = -m*g
motor_forces_hover = f_i * ones(4,1);

Zc = -m*g;   % equals -m*g at hover
Lc = 0;
Mc = 0;
Nc = 0;
```

Problem 1.2

```
if P1_2 == 1
    % Initialization Stuff
    mu = 0; %to ignore Aerodynamic Forces/Moments
    vu = 0; %to ignore Aerodynamic Forces/Moments

    % Establishing State Vector
    x_e = 0;
    y_e = 0;
    z_e = 0;
    phi = 0;
    theta = 0;
    psi = 0;
    u_e = 0;
    v_e = 0;
    w_e = 0;
    p = 0;
    q = 0;
    r = 0;
    var1 = [x_e; y_e; z_e; phi; theta; psi; u_e; v_e; w_e; p; q; r];

    odefun = @(t,x) QuadrotorEOM(t, x, g, m, I, d, km, vu, mu,
motor_forces_hover);
    [t1, X1] = ode45(odefun, [0, t], var1);

    U2 = repmat([Zc, Lc, Mc, Nc], length(t1), 1); % size = length(t) x 4

    PlotAircraftSim(t1, X1, U2, fig1, col1)
end
```

Problem 1.3

```
if P1_3 == 1
    % Establishing State Vector
    x_e = 0;
    y_e = 0;
    z_e = 0;
    phi = 0;
    theta = 0;
    psi = 0;
    u_e = 0;
    v_e = 0;
    w_e = 0;
    p = 0;
    q = 0;
    r = 0;

    var2 = [x_e; y_e; z_e; phi; theta; psi; u_e; v_e; w_e; p; q; r];

    odefun = @(t,x) QuadrotorEOM(t, x, g, m, I, d, km, vu, mu,
motor_forces_hover);
```

```
    [t2, X2] = ode45(odefun, [0, t], var2);

    U2 = repmat([Zc, Lc, Mc, Nc], length(t2), 1); % size = length(t) x 4

    PlotAircraftSim(t2,X2,U2,fig2,col2)
end
```

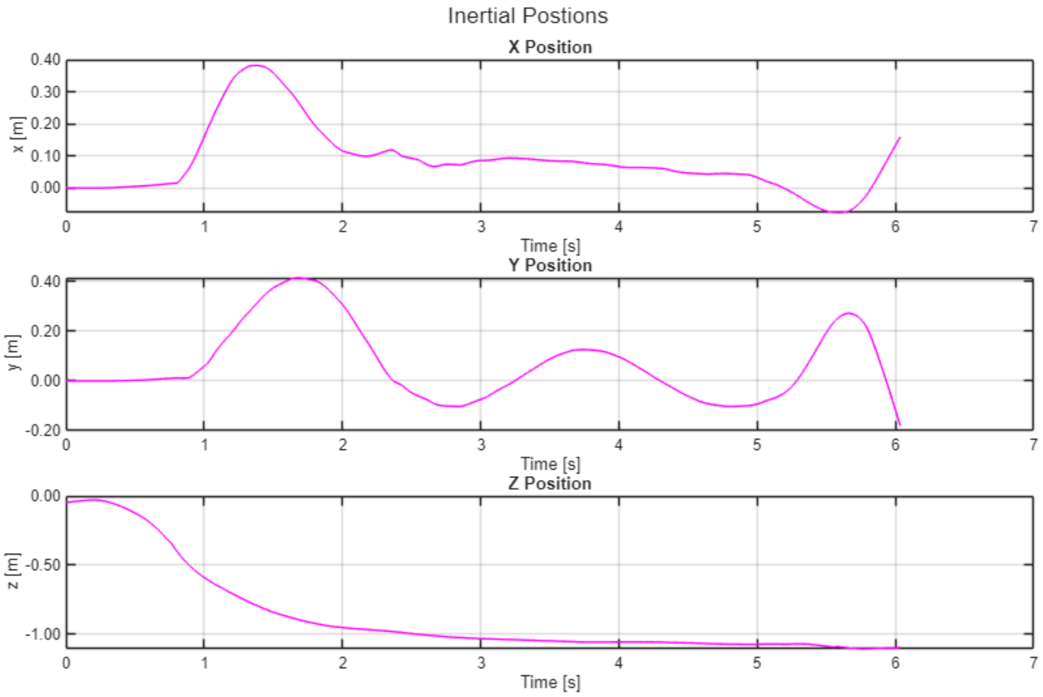
Problem 1.4

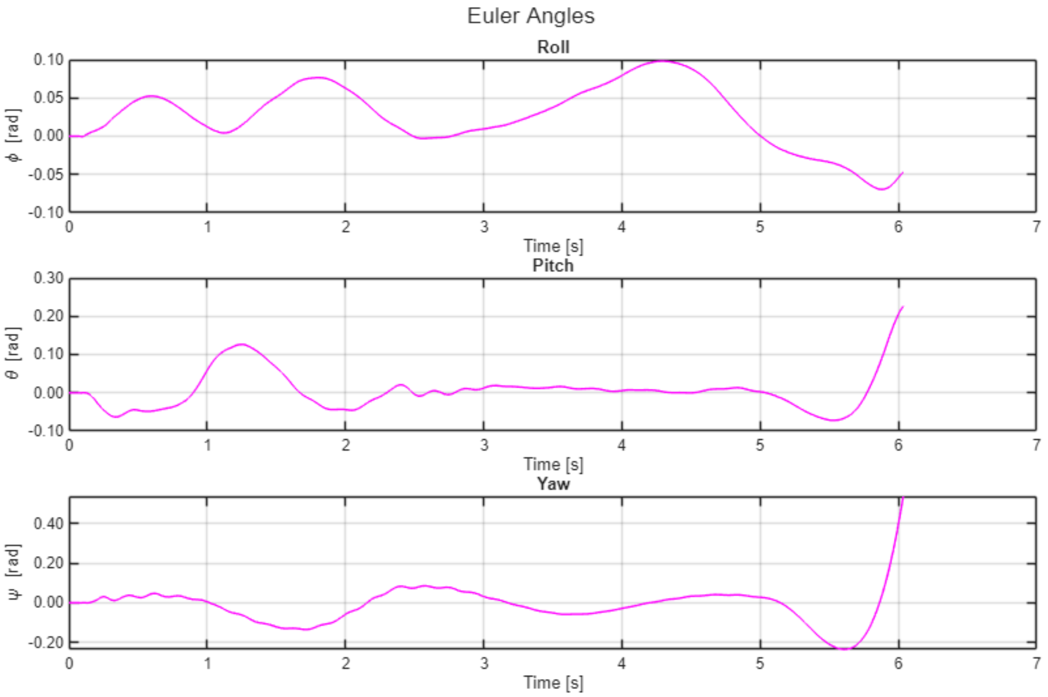
```
if P1_4 == 1
    % CASE 1: Establishing State Vector (ONLY CHANGING 5 m/s EAST)
    x_e1 = 0;
    y_e1 = 0;
    z_e1 = 0;
    phi1 = asin((vu*25) / (m*g));
    theta1 = 0;
    psi1 = 0;
    u_e1 = 0;
    v_e1 = 5;
    w_e1 = 0;
    p1 = 0;
    q1 = 0;
    r1 = 0;

    % Controls for this Trim
    %Zc31 = -m*g*cos(phi1);
    Zc31 = -m*g;
    Fc31 = [0, 0, Zc31];
    Lc31 = 0;
    Mc31 = 0;
    Nc31 = 0;
    Gc31 = [Lc31, Mc31, Nc31]; % [Lc, Mc, Nc]
    f_trim31 = MotorForces(Fc31, Gc31, d, km);

    var31 = [x_e1; y_e1; z_e1; phi1; theta1; psi1; u_e1; v_e1; w_e1; p1; q1;
r1];

    odefun = @(t,x) QuadrotorEOM(t, x, g, m, I, d, km, vu, mu, f_trim31);
    [t31, X31] = ode45(odefun, [0, t], var31);

    U31 = repmat([Zc31, Lc31, Mc31, Nc31], length(t31), 1); % size =
length(t) x 4

    PlotAircraftSim(t31,X31,U31,fig3,col3)

    % CASE 2: Establishing State Vector (with 90 deg Yaw)
    x_e2 = 0;
    y_e2 = 0;
    z_e2 = 0;
    phi2 = 0;
    theta2 = -asin((vu*25) / (m*g));
    psi2 = deg2rad(90);
    u_e2 = 5 * cos(theta2);
    v_e2 = 0;
    w_e2 = 5 * sin(theta2);
    p2 = 0;
    q2 = 0;
    r2 = 0;

    % Controls for this Trim
    Zc32 = -m*g*cos(theta2);
    Fc32 = [0, 0, Zc32];
    Gc32 = [0, 0, 0]; % [Lc, Mc, Nc] (same as the part above)
    f_trim32 = MotorForces(Fc32, Gc32, d, km);

    var32 = [x_e2; y_e2; z_e2; phi2; theta2; psi2; u_e2; v_e2; w_e2; p2; q2;
r2];

    odefun = @(t,x) QuadrotorEOM(t, x, g, m, I, d, km, vu, mu, f_trim32);
    [t32, X32] = ode45(odefun, [0, t], var32);

    U32 = repmat([Zc, Lc, Mc, Nc], length(t32), 1); % replicates a tile an
array. size = length(t) x 4

    PlotAircraftSim(t32,X32,U32,fig4,col4)
end
```

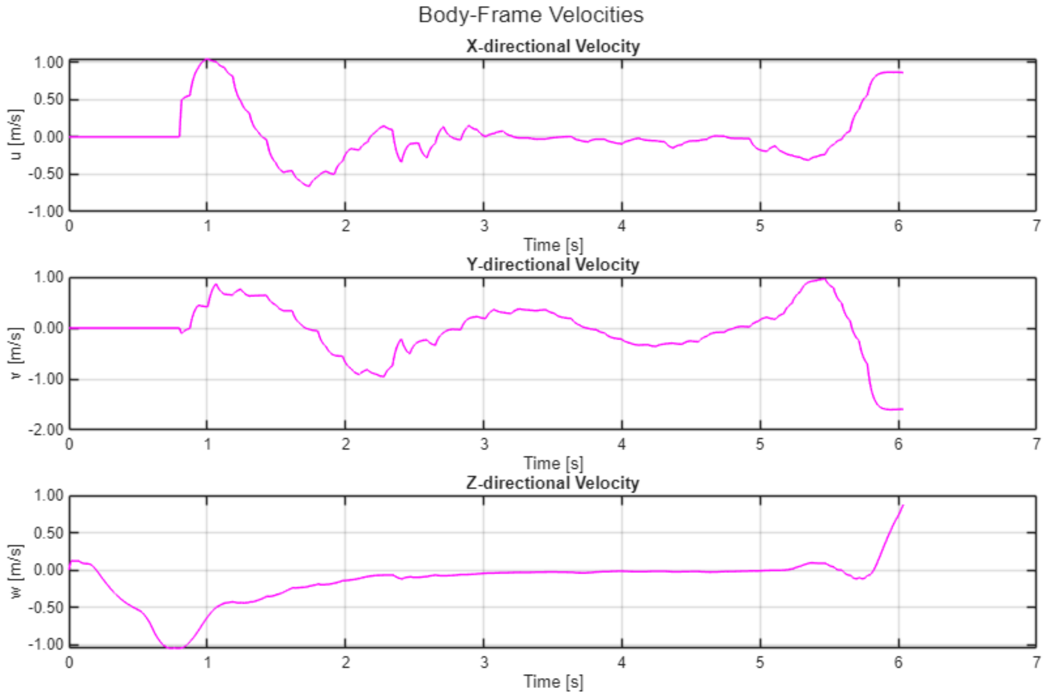
Problem 1.5

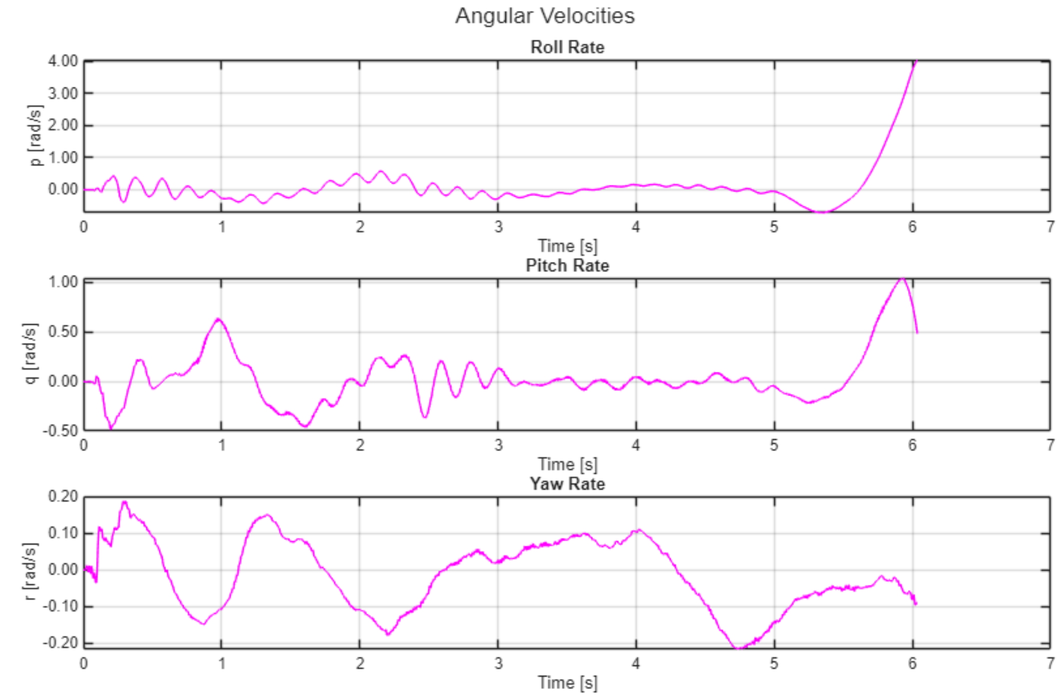
```
if P1_5 == 1;
    % Load Data
    RSdata = load('RSdata_nocontrol.mat');

    t = RSdata.rt_estim.time(:);
    Xest = RSdata.rt_estim.signals.values;
    xE = Xest(:,1);
    yE = Xest(:,2);
    zE = Xest(:,3);
    psi = Xest(:,4);
    theta = Xest(:,5);
    phi = Xest(:,6);
    uE = Xest(:,7);
    vE = Xest(:,8);
    wE = Xest(:,9);
    p = Xest(:,10);
    q = Xest(:,11);
    r = Xest(:,12);

    vec = [xE, yE, zE, psi, theta, phi, uE, vE, wE, p, q, r];

    Mot = RSdata.rt_motor.signals.values;
    omega1 = sqrt(Mot(:,1)*13840.4);
    omega2 = sqrt(Mot(:,2)*13840.4);
    omega3 = sqrt(Mot(:,3)*13840.4);
    omega4 = sqrt(Mot(:,4)*13840.4);
    omega_motor = [omega1, omega2, omega3, omega4];

    PlotAircraftSim(t, vec, omega_motor, fig5, col5)
end
```

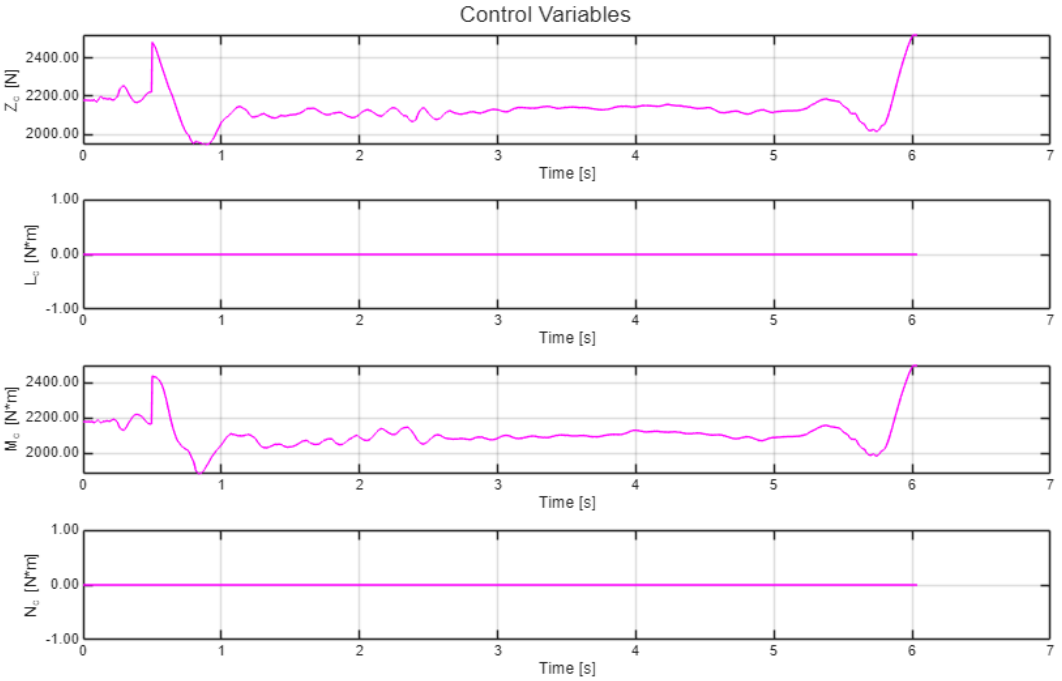
```
Warning: Imaginary parts of complex X and/or Y arguments ignored.
Warning: Imaginary parts of complex X and/or Y arguments ignored.
```

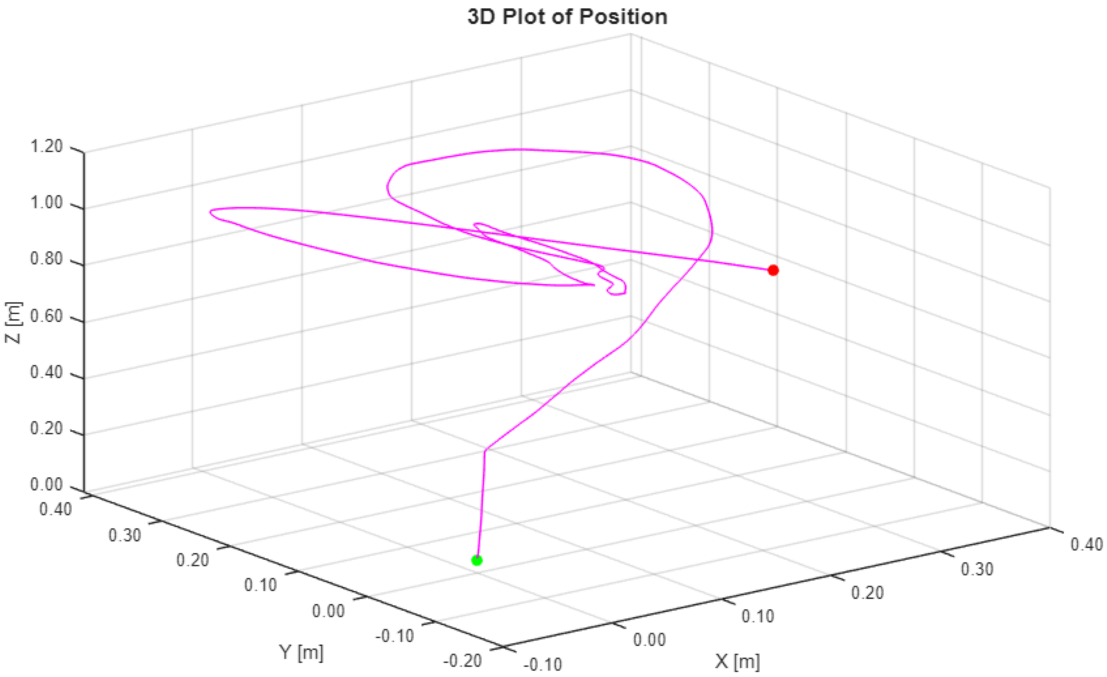



Euler Angles
Roll
ϕ [rad]
0.10
0.05
0.00
-0.05
-0.10
0
1
2
3
4
5
6
7
Time [s]
Pitch
θ [rad]
0.30
0.20
0.10
0.00
-0.10
0
1
2
3
4
5
6
7
Time [s]
Yaw
ψ [rad]
0.40
0.20
0.00
-0.20
0
1
2
3
4
5
6
7
Time [s]


Body-Frame Velocities
X-directional Velocity
u [m/s]
1.00
0.50
0.00
-0.50
-1.00
0
1
2
3
4
5
6
7
Time [s]
Y-directional Velocity
v [m/s]
1.00
0.00
-1.00
-2.00
0
1
2
3
4
5
6
7
Time [s]
Z-directional Velocity
w [m/s]
1.00
0.50
0.00
-0.50
-1.00
0
1
2
3
4
5
6
7
Time [s]


Angular Velocities
Roll Rate
p [rad/s]
4.00
3.00
2.00
1.00
0.00
0
1
2
3
4
5
6
7
Time [s]
Pitch Rate
q [rad/s]
1.00
0.50
0.00
-0.50
0
1
2
3
4
5
6
7
Time [s]
Yaw Rate
r [rad/s]
0.20
0.10
0.00
-0.10
-0.20
0
1
2
3
4
5
6
7
Time [s]


Control Variables
Z_c [N]
2400.00
2200.00
2000.00
0
1
2
3
4
5
6
7
Time [s]
L_c [N*m]
1.00
0.00
-1.00
0
1
2
3
4
5
6
7
Time [s]
M_c [N*m]
2400.00
2200.00
2000.00
0
1
2
3
4
5
6
7
Time [s]
N_c [N*m]
1.00
0.00
-1.00
0
1
2
3
4
5
6
7
Time [s]



Published with MATLAB® R2025a