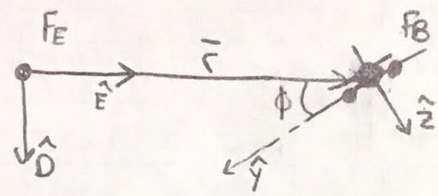
ASEN 3128 Homework 2

Jack Lambert[*]
University of Colorado Boulder, ASEN 3128-012, Group 22
Submitted 2/5/2018

[*]SID: 104414093

A. Questions 1-8

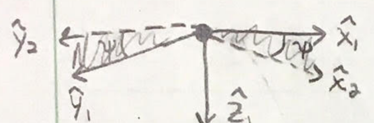
Jack Lambert | ASEN 3128 | Section 012 | HW-2



$|V^E| = 100$ m/s

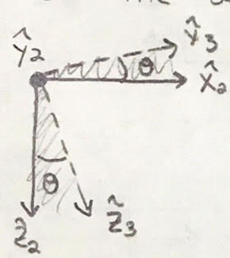
$|\bar{r}| = 1$ km $= 1000$ m

1.) ψ - The angle difference between $\hat{N}, \hat{E}$ & $\hat{X}, \hat{Y}$ rotated about the $\hat{D}$-axis



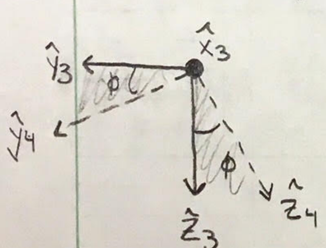
$\boxed{\psi = 180°}$ $\rightarrow$ $\hat{y}_1 = -\hat{y}_2$, $\hat{x}_1 = \hat{x}_2$

θ - The angle difference between $\hat{z}_2, \hat{x}_2$ & $\hat{z}_3, \hat{x}_3$ rotated about the $\hat{y}_2$-axis



$\boxed{\theta = 0°}$ $\longrightarrow$ $\hat{z}_2 = \hat{z}_3$, $\hat{x}_2 = \hat{x}_3$

ϕ - The angle difference between $\hat{y}_3, \hat{z}_3$ & $\hat{y}_4, \hat{z}_4$ rotated about the $\hat{x}_3$-axis



$\boxed{\phi = \phi_{Picture}}$

2.) $\dot{\psi}$ - Rate of change of ψ as body frame rotates about $\hat{D}$-axis

$\omega r = V$

$\omega = \frac{V}{r}$

$\dot{\psi} = \bar{\omega}\,\hat{D} = \frac{|V|}{r}\hat{D} = \boxed{0.1 \left(\frac{rad}{s}\right) \hat{D}}$

$\dot{\theta}$ - Rate of change of θ as body frame rotates about $\hat{y}_2$-axis

$\boxed{\dot{\theta} = 0 \left[\frac{rad}{s}\right]}$

$\dot{\phi}$ – Rate of change of ϕ as body frame rotates about the $\hat{X}_3$-axis

$$\boxed{\dot{\phi} = 0 \left(\frac{rad}{s}\right)}$$

3.) p: X- component of angular velocity in body coordinates

q: y- component of angular velocity in body coordinates

r: z - component of angular velocity in body coordinates

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$

$-\sin(\theta) = 0$ $\quad \dot{\psi} = 0.1\,\hat{z}$

$\cos(\theta) = 1$

$\dot{\phi} = \dot{\theta} = 0$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} - \dot{\psi}\sin\theta \\ \dot{\theta}\cos\phi + \dot{\psi}\sin\phi\cos\theta \\ \dot{\psi}\cos\phi\cos\theta - \dot{\theta}\sin\phi \end{bmatrix} = \boxed{\begin{bmatrix} 0 \\ 0.1\sin\phi \\ 0.1\cos\phi \end{bmatrix}} \left[\frac{rad}{s}\right]$$

4.) $\dot{p}$: x-component of angular acceleration in body coordinates

$\dot{q}$: y- component of angular acceleration in body coordinates

$\dot{r}$: z - component of angular acceleration in body coordinates

$$\boxed{\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \left[\frac{rad}{s^2}\right]}$$

5.) $\bar{G}$: The resoltant external moment about a body's center of gravity

- $\bar{G}_E$ — is not fixed since the moments of inertia become variables

$$\bar{G}_E = \frac{d^E}{dt}\left(\bar{h}^E\right)_E = \underbrace{\dot{I}_E}_{\text{variable} \longrightarrow \text{Not Fixed}}\omega + I_E\cancel{\dot{\omega}}^{=0}$$

- $\bar{G}_B$ — is fixed since the moments of inertia are constant in F_B

$$\bar{G}_B = \left(\frac{d^E}{dt} h^E\right)_B = \frac{d^B}{dt}\bar{h}^E + \bar{\omega}_B \times \bar{h}^E_B$$

$$= \frac{d}{dt}\bar{h}^E_B + \bar{\omega}_B \times \bar{h}^E_B$$

$$= \underbrace{I_B}_{\text{constant}}\frac{d}{dt}\bar{\omega}_B + \bar{\omega}_B \times \bar{h}^E_B \longrightarrow I_B \cdot \bar{\omega}_B$$

$$= I_B \cdot \dot{\omega}_B + \bar{\omega}_B \times \left(I_B \cdot \bar{\omega}_B\right)$$

$$\rightarrow \cancel{\begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \underbrace{\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}}_{=0}} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} pI_x \\ qI_y \\ rI_z \end{bmatrix} = \begin{array}{c} \begin{array}{ccc} \hat{x} & \hat{y} & \hat{z} \end{array} \\ \begin{bmatrix} p & q & r \\ pI_x & qI_y & rI_z \end{bmatrix} \end{array}$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 0 \\ 0.1\sin\phi \\ 0.1\cos\phi \end{bmatrix} \left(\frac{rad}{s}\right)$$

$$\rightarrow \begin{bmatrix} qr(I_z - I_y) \\ pr(I_x - I_z) \\ pq(I_y - I_x) \end{bmatrix} = \boxed{\begin{bmatrix} 0.01\sin\phi\cos\phi\,(I_z - I_y) \\ 0 \\ 0 \end{bmatrix} \left[kg - m^2\right]}$$

6.) $\bar{h}$: The angular momentum of the body

$\bar{h}_E$: $I_E \bar{\omega}_E \longrightarrow$ I_E is a variable, therefore, the angular momentum of the body is not fixed

$\bar{h}_B = I_B \bar{\omega}_B \longrightarrow$ I_B is constant since it is in the F_B coordinate frame, therefore, the angular momentum is fixed

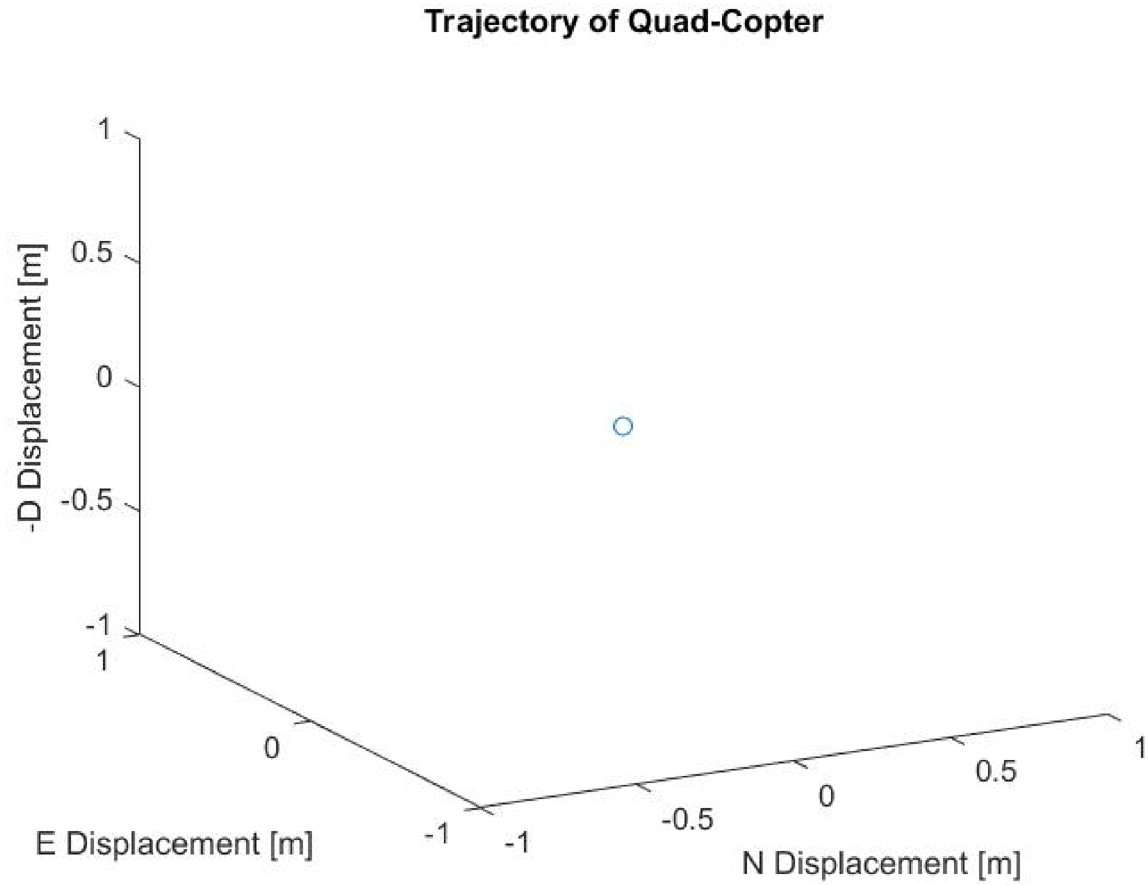
$$\bar{h}_B = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} p I_x \\ q I_y \\ r I_z \end{bmatrix} = \boxed{\begin{bmatrix} 0 \\ 0.1 \sin\phi \, I_y \\ 0.1 \cos\phi \, I_z \end{bmatrix} [kg\text{-}m^2]}$$

7.) If the aircraft from assignment 1 were replaced with a helicopter instead of an airplane the results from assignment 1 parts 1-8 & parts 1-6 of this assignment would remain unchanged. Since the only change would arrise from drag forces and angular momentum differences which of both are not accounted for for these problems.

8.) For a quadcopter, translation does not effect attitude. The quadcopter can move in any direction and It would not effect the copter's attitude since the moments would cancel and the force would be constant about the center, therefore the attitude would remain fixed. While translation does not effect attitude, the attitude most certainly effects translation. The orientation of the attitude is how quad copters change direction when flying, even with constant velocity.

B. Problem 9:

For this question we were asked to simulate the trajectory of a quad-copter including the azimuth, elevation and bank Euler angle attitude representation. This model includes all forces and moments produced by the quad-copter's four motors in conjunction to all aerodynamic and gravitational effects. This question also asks if the trajectory would change if there were no aerodynamic forces, in which it would not since the forces of drag do not act upon an object that has no velocity in any direction. The following is a representation of the quad-copter in steady hovering flight.



C. Problem 10:

For this question we were asked to delve into the process of finding the trim state that would give the quad copter a constant velocity of 5 meters per second in the east direction if the body coordinate frame has an azimuth of 0 degrees. We then had find a trim state that gives the quad copter a constant velocity of 5 meters per second east, but this time with an azimuth of 90 degrees. The two results gave the same plots as was requested by the question so only one plot is given. The difference in azimuth ultimately only affected the orientation of the trim state values - where the affected values were the body fixed velocities and Euler angles. The derivations, plot, and code are provided:

Code Derivations for Force & Moments

$$\begin{bmatrix} \dot{U}^E \\ \dot{V}^E \\ \dot{W}^E \end{bmatrix} = \begin{bmatrix} X/m - g\sin(\theta) \\ Y/m + g\cos(\theta)\sin(\phi) \\ Z/m + g\cos(\theta)\cos(\phi) \end{bmatrix} \rightarrow$$ Integrate this for Velocity

$V^E = V \rightarrow$ No wind

where:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} -\eta U_E^2 \\ -\eta V^2 \\ -\zeta W^2 - f_1 - f_2 - f_3 - f_4 \end{bmatrix}$$

$$\eta = \tfrac{1}{2} C_D A_x = \tfrac{1}{2} C_D A_y \qquad f_1 = f_2 = f_3 = f_4 = \frac{mg}{4}$$

$$\zeta = \tfrac{1}{2} C_D A_z \qquad \text{from } \Sigma F_{\hat{z}} = 0$$

$$\begin{bmatrix} \dot{x}_E \\ \dot{y}_E \\ \dot{z}_E \end{bmatrix} = L_{EB} \begin{bmatrix} U^E \\ \dot{V}^E \\ \dot{W}^E \end{bmatrix} \rightarrow$$ Integrate this for inertial Position

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix} = \begin{bmatrix} I_x \dot{p} + qr(I_z - I_y) \\ I_y \dot{q} + rp(I_x - I_z) \\ I_z \dot{r} + pq(I_y - I_x) \end{bmatrix}$$

$\rightarrow$ Terms w/ I_{zx} & $\dot{h} = 0$

$\rightarrow$ solve for $\dot{p}, \dot{q}, \dot{r}$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (L + qr(I_y - I_z))/I_x \\ (M + rp(I_z - I_x))/I_y \\ (N + pq(I_x - I_y))/I_z \end{bmatrix} \rightarrow$$ Integrate this for angular velocity

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix} = \begin{bmatrix} r(f_2 + f_3 - f_1 - f_4) - \alpha p^2 \\ r(f_3 + f_4 - f_2 - f_1) - \alpha q^2 \\ k(f_2 + f_4 - f_1 - f_2) - \beta r^2 \end{bmatrix}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = T \begin{bmatrix} p \\ q \\ r \end{bmatrix} \rightarrow$$ Integrate this. for Euler angles

Problem 10 Derivations-

Knowns:

$\psi = 0^\circ$

$\theta = 0^\circ$

$U^E = 0\,(m/s)$

$\dot{Y}_E = 5\ m/s$

$\dot{Z}_E = 0\ m/s$

Unknowns:

ϕ

F

V^E

W^E

$$\dot{Y}_E = \underbrace{U^E}_{=0} \cos\theta \sin\psi + V^E(\sin\phi \underbrace{\sin\theta}_{=0} \sin\psi + \cos\phi \underbrace{\cos\psi}_{=1}) + W^E(\cos\phi \underbrace{\sin\theta}_{=0} \cos\psi - \sin\phi \underbrace{\sin\psi}_{=0})$$

$$= V^E\cos\phi - W^E\sin\phi = \dot{Y}_E$$

$$\dot{Z}_E = -\underbrace{U^E}_{=0}\sin\theta + V^E\sin\phi\underbrace{\cos\theta}_{=1} + W^E\cos\phi\underbrace{\cos\theta}_{=1}$$

$$= V^E\sin\phi + W^E\cos\phi = 0$$

$$V^E = \frac{-W^E\cos\phi}{\sin\phi}$$

$$V^E = \frac{\dot{Y}_E + W^E\sin\phi}{\cos\phi}$$

$$\frac{\dot{Y}_E + W^E\sin\phi}{\cos\phi} = \frac{-W^E\cos\phi}{\sin\phi} \rightarrow \dot{Y}_E + W^E\sin\phi = \frac{-W^E\cos^2\phi}{\sin\phi}$$

$$-\dot{Y}_E = W^E\left(\sin\phi + \frac{\cos^2\phi}{\sin\phi}\right) = \frac{W^E}{\sin\phi}\left(\underbrace{\sin^2\phi + \cos^2\phi}_{=1}\right)$$

$$\boxed{W_E = -\dot{Y}_E\sin\phi}$$

$$V^E = \frac{-(-\dot{Y}_E\sin\phi)\cos\phi}{\sin\phi} \rightarrow \boxed{V^E = \dot{Y}_E\cos\phi}$$

Equating Forces in $\hat{y}$ direction

$V^E = \dot{Y}_E\cos\phi$, $\dot{V}^E = 0$ constant velocity

$$m\underbrace{\dot{V}^E}_{=0} = mg\underbrace{\cos\theta}_{=1}\sin\phi - {}^{mg}/\,V^{E^2}$$

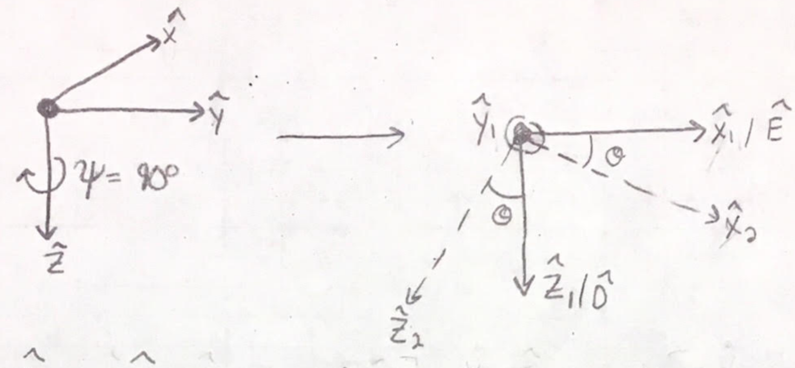
$$\boxed{{}^{mg}/\,(\dot{Y}_E\cos\phi)^2 = mg\sin\phi} \rightarrow \text{Solve for } \phi \text{ symbolically w/ MATLAB \& check that } \phi \text{ is in right quadrin}$$

Summing forces in $\hat{z}$ direction

$$F - {}^{mg}/\,W^{E^2}\,\mathrm{sign}(W^E) + mg\cos\phi$$

$$F = mg\cos\phi - {}^{mg}/\,W^E\,\mathrm{sign}(W^E) \longrightarrow f_1 = f_2 = f_3 = f_4 = F/4$$

Problem 10 Derivations -



$$\hat{E} = \hat{x}_2 \cos\theta - \hat{z}_2 \sin\theta$$

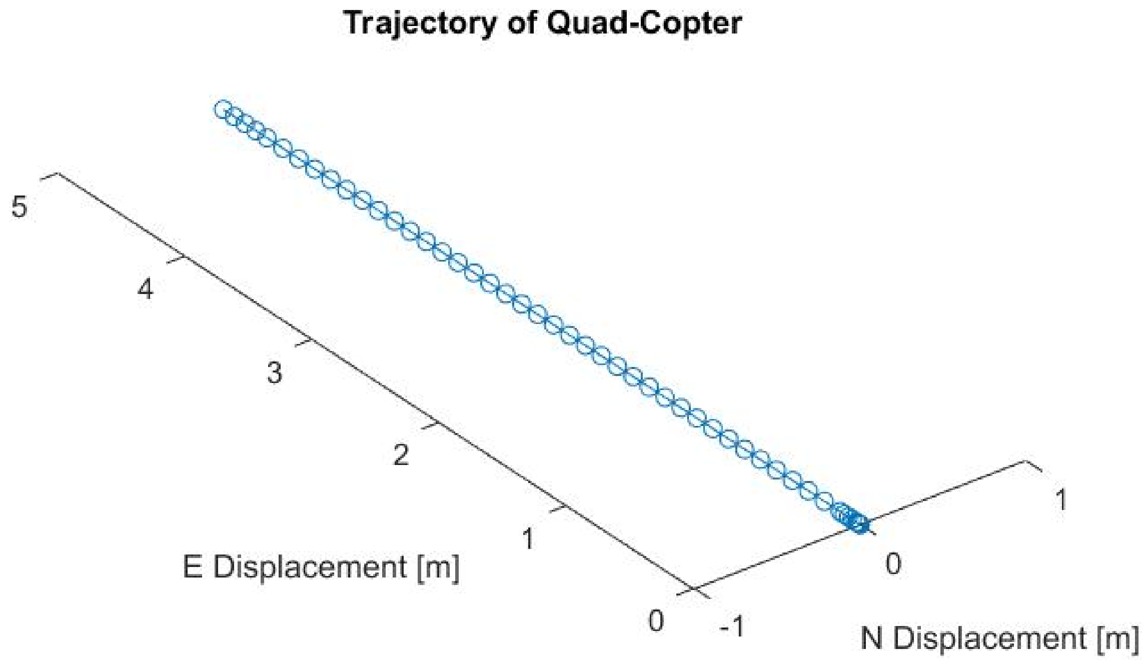
$$\dot{y}_E \hat{E} = \dot{y}_E (\cos\theta\, \hat{x}_2 - \sin\theta\, \hat{z}_2)$$

$$\boxed{U^E = \dot{y}_E \cos\theta, \quad W^E = -\dot{y}_E \sin\theta}$$ for $\theta = (+)$ sign

$\psi = 90°$

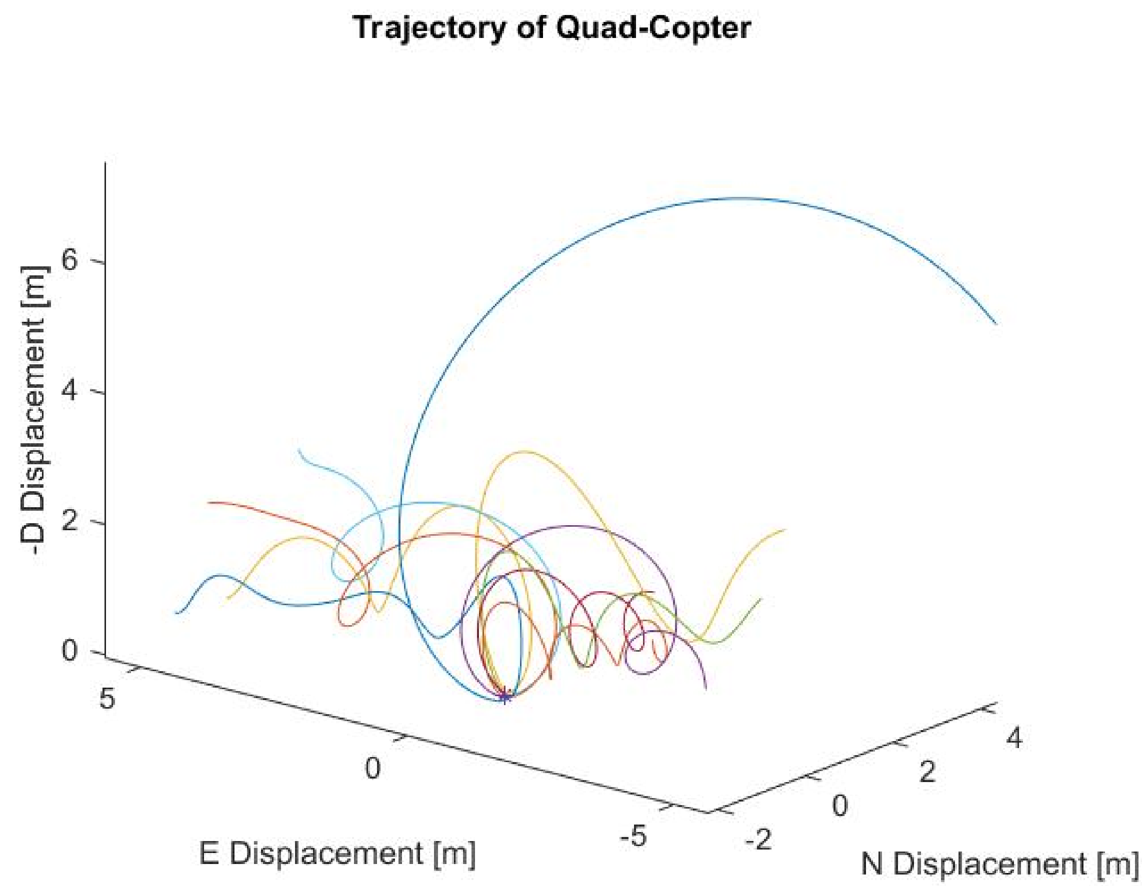
$\theta = -\phi \leftarrow$ from solving ϕ symbolically in MATLAB

- sign is negative due to $\hat{y}_1$ pointing out of the page in opposition to ϕ where $\hat{x}$ pointed into the page
- Force Remain same as quadcopter is just rotated about the azimuth

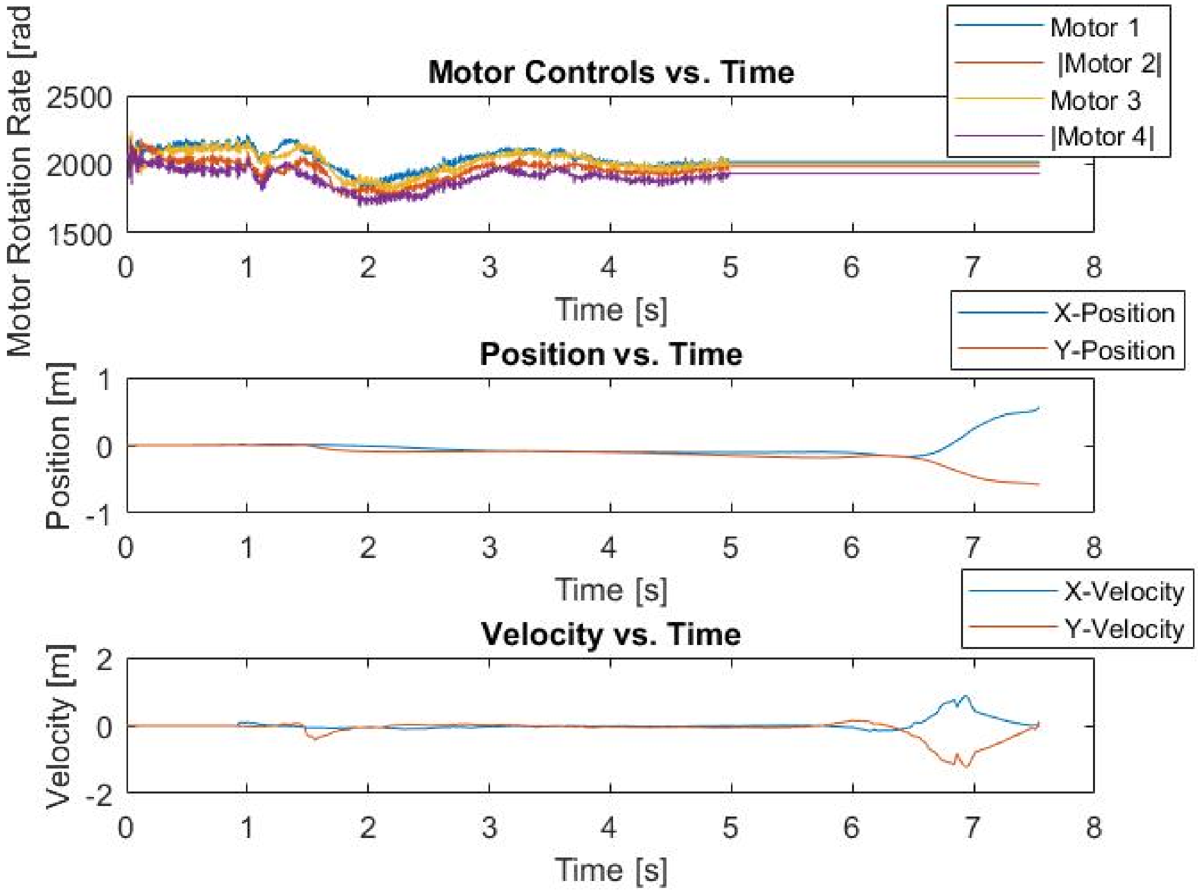


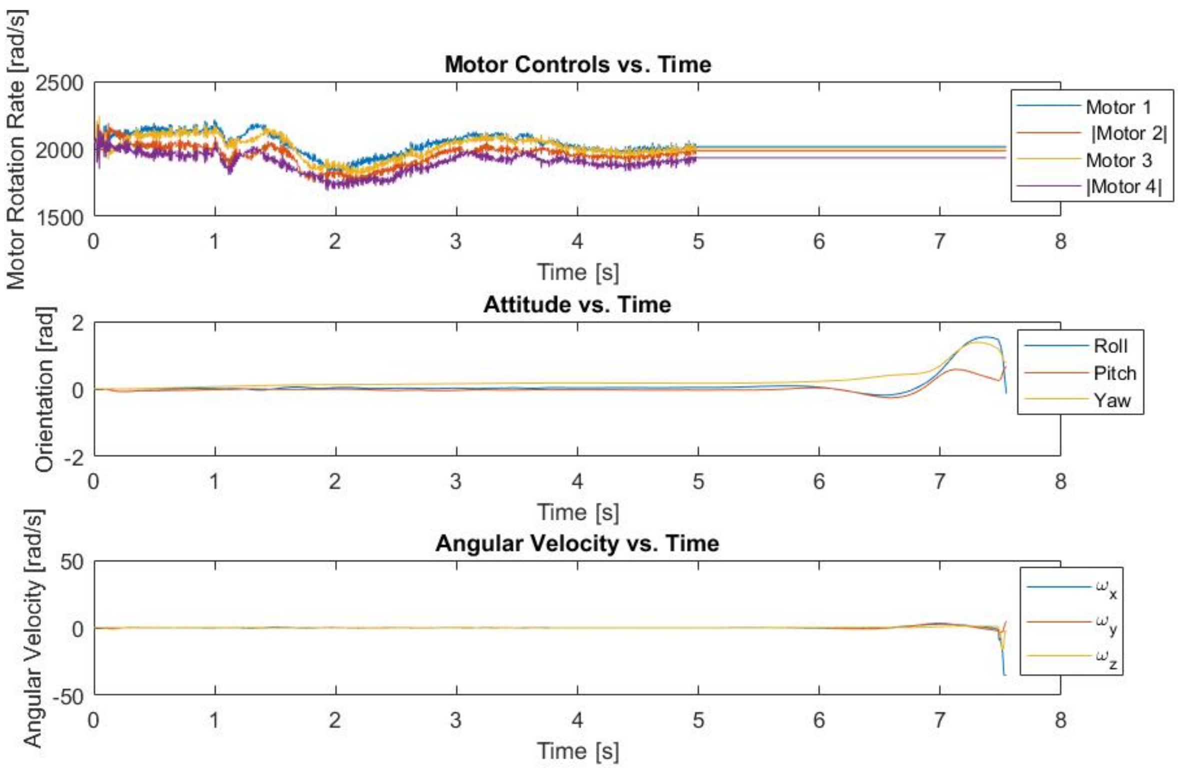
D. Problem 11:

To determine the stability of a quad copter in steady hovering flight with no control inputs, it was necessary to see what the quad-copter would do when there were small disturbances due to the imperfect nature of air conditions, forces from each of the motors and attitude. To see what would happen, the simulation was ran with a normal distribution of random numbers chosen to replicate the small disturbances around initial conditions that should have been zero. The corresponding plots resulted from the small perturbations about the hovering trim state with no controls. When simulating the stability of the quad-copter at a hovering trim state, with fixed controls it was noticed to be unstable. It was noticed that the quad-copter did not return to its original state or even a state of control, therefore the quad-copter is noted to be unstable at this fixed control hovering state.



Lastly, to show how the stability of a hovering quad-copter with no controls is unstable we tested with a PARROT mini quad-copter that had sensors attached and recorded data on the translation and orientation of the quad-copter as a function of time as well as the motor controls. As can be seen in the figures below - the quad-copters became unstable after the controls were turned off. It took a small moment in time for the variations in conditions to effect the state of the quad-copter, but once the small perturbations occurred - the copter could not correct itself since the controls were off. The translation and orientation change drastically and the quad-copter crashes - all due to the unstable trim state once the controls were off. These plots illustrate this below:


Motor Controls vs. Time
Motor Rotation Rate [rad
2500
2000
1500
0
1
2
3
4
5
6
7
8
Time [s]
Motor 1
|Motor 2|
Motor 3
|Motor 4|
Position vs. Time
Position [m]
1
0
-1
Time [s]
X-Position
Y-Position
Velocity vs. Time
Velocity [m]
2
0
-2
Time [s]
X-Velocity
Y-Velocity



E. Appendix A - MATLAB Code

1. Main Function for ODE45 to Call for all Problems

```
function [dydt] = main(t,y)
%% Global Variables
global mass g R k eta zeta alpha beta Ix Iy Iz f1 f2 f3 f4
%% Derivatives to be Integrated
% Translational Motion
dx = y(1); % N - location
dy = y(2); % E - location
dz = y(3); % -D - location
Vx = y(4); % u - component of velocity
Vy = y(5); % v compenent of velocity
Vz = y(6); % w component of velocity
% Rotational Motion
phi = y(7); % Attitude Euler Angles
theta = y(8); % Attitude Euler Angles
psi = y(9); % Attitude Euler Angles
p = y(10); % Angular velocity about the x-axis [rad/s]
q = y(11); % Angular Velocity about the y-axis [rad/s]
r = y(12); % Angular Velocity about the z-axis [rad/s]
%% Velocity
mag = sqrt(Vx^2+Vy^2+Vz^2); % Magnitude of velocity rel to body
% Redefining for context
u = Vx;
v = Vy;
w = Vz;
%% Forces to Acceleration
% Aerodynimc Forces
A_c = [0 0 -(f1+f2+f3+f4)]; % Control Forces
A_a = [-eta*u^2*sign(u) -eta*v^2*sign(v) -zeta*w^2*sign(w)]; % Aerodynamics Forces
A_b = A_c + A_a; % Combined Forces
% Kinematic Equations
```

```
dydt(1) = u*cos(theta)*cos(psi)+ v*(sin(phi)*sin(theta)*cos(psi)-cos(phi)*sin(psi))...
    + w*(cos(phi)*sin(theta)*cos(psi)+sin(phi)*sin(psi));
dydt(2) = u*cos(theta)*sin(psi)+ v*(sin(phi)*sin(theta)*sin(psi)+cos(phi)*cos(psi))...
    + w*(cos(phi)*sin(theta)*sin(psi)-sin(phi)*cos(psi));
dydt(3) = -u*sin(theta)+v*sin(phi)*cos(theta)+w*cos(phi)*cos(theta);
dydt(4) = (A_b(1)-mass*g*sin(theta))/mass;
dydt(5) = (A_b(2)+mass*g*cos(theta)*sin(phi))/mass;
dydt(6) = (A_b(3)+mass*g*cos(theta)*cos(phi))/mass;
%% Moments to Rotations
G_a = [-alpha*p^2 -alpha*q^2 -beta*r^2];
G_c = [R*(f2+f3-f1-f4) R*(f3+f4-f2-f1) k*(f2+f4-f1-f2)];
G_b = G_a + G_c;
% Kinemeatic Equations
dydt(7) = p + (q*sin(phi)+r*cos(phi))*tan(theta);
dydt(8) = q*cos(phi)-r*sin(phi);
dydt(9) = (q*sin(phi)+r*cos(phi))*sec(theta);
dydt(10) = (G_b(1)+q*r*(Iy - Iz))/Ix;
dydt(11) = (G_b(2)+r*p*(Iz-Ix))/Iy;
dydt(12) = (G_b(3)+p*q*(Ix-Iy))/Iz;

dydt = dydt';
```

2. *Problem 9 function*

```
%% Jack Lambert
% Aircraft Dynmaics Homework 2
% Problem 9
global mass g R k eta zeta alpha beta Ix Iy Iz f1 f2 f3 f4
%% Constants
mass = 68/1000; % [kg]
L_arm = 6/100; % [m]
eta = 1*10^(-3); % Aerodynamic Coefficient for drag [N /(m/s)^2]
zeta = 3*1^(-3);  % Aerodynamic Coefficient for drag [N /(m/s)^2]
alpha = 2*10^(-6); % Aerodynamic Coefficient for drag [N /(rad/s)^2]
beta = 1*10^(-6); % Aerodynamic Coefficient for drag [N /(rad/s)^2]
Ix = 6.8*10^(-5); % MOI about x-axis [kg*m^2]
Iy = 9.2*10^(-5); % MOI about x-axis [kg*m^2]
Iz = 1.35*10^(-4); % MOI about x-axis [kg*m^2]
R = sqrt(2)/2*L_arm; % Distance to COG [m]
k = 0.0024; % [m]
g = 9.81; % [m/s^2]
%% Initial Conditions
condition(1) = 0; % N - location [m]
condition(2) = 0; % E - location [m]
condition(3) = 0; % -D - location [m]
condition(4) = 0; % u - component of velocity [m/s]
condition(5) = 0; % v compenent of velocity [m/s]
condition(6) = 0; % w component of velocity [m/s]
% Rotational Motion
condition(7) = 0; % Phi Euler Angle [rad]
condition(8) = 0; % Theta Euler Angle [rad]
condition(9) = 0; % Psi Euler Angle [rad]
condition(10) = 0; % Angular velocity about the x-axis [rad/s]
condition(11) = 0; % Angular Velocity about the y-axis [rad/s]
condition(12) = 0; % Angular Velocity about the z-axis [rad/s]
%% Solving Differential Equations w/ ODE45
f1 = (mass*g)/4; % Force for steady Level Flight about Motor 1
f2 = (mass*g)/4; % Force for steady Level Flight about Motor 1
f3 = (mass*g)/4; % Force for steady Level Flight about Motor 1
f4 = (mass*g)/4; % Force for steady Level Flight about Motor 1
[t,z] = ode45('main',[0 5],condition);
plot3(z(:,1),z(:,2),-z(:,3),'-o')
title('Trajectory of Quad-Copter')
xlabel('N Displacement [m]')
ylabel('E Displacement [m]')
zlabel('-D Displacement [m]')
```

```
%% end
```

3. *Problem 10 Part 1*

```
%% Jack Lambert
% Aircraft Dynmaics Homework 2
% Problem 10
global mass g R k eta zeta alpha beta Ix Iy Iz f1 f2 f3 f4
%% Constants
mass = 68/1000; % [kg]
L_arm = 6/100; % [m]
eta = 1*10^(-3); % Aerodynamic Coefficient for drag [N /(m/s)^2]
zeta = 3*1^(-3);  % Aerodynamic Coefficient for drag [N /(m/s)^2]
alpha = 2*10^(-6); % Aerodynamic Coefficient for drag [N /(rad/s)^2]
beta = 1*10^(-6); % Aerodynamic Coefficient for drag [N /(rad/s)^2]
Ix = 6.8*10^(-5); % MOI about x-axis [kg*m^2]
Iy = 9.2*10^(-5); % MOI about x-axis [kg*m^2]
Iz = 1.35*10^(-4); % MOI about x-axis [kg*m^2]
R = sqrt(2)/2*L_arm; % Distance to COG [m]
k = 0.0024; % [m]
g = 9.81; % [m/s^2]
%% Initial Conditions
y_E = 5; % Interial y-component of Velocity
syms phi0
eqn = eta*y_E^2*cos(phi0)^2 == mass*g*sin(phi0);
phi = double(solve(eqn,phi0));
phi(imag(phi) ~= 0) = [];
i = find(phi<pi/2);
phi = phi(i); % [rad]

v_E = y_E*cos(phi);
w_E = -y_E*sin(phi);

Forcemag = -zeta*w_E^2*sign(w_E)+mass*g*cos(phi);

condition(1) = 0; % N - location [m]
condition(2) = 0; % E - location [m]
condition(3) = 0; % -D - location [m]
condition(4) = 0; % u - component of velocity [m/s]
condition(5) = v_E; % v compenent of velocity [m/s]
condition(6) = w_E; % w component of velocity [m/s]
% Rotational Motion
condition(7) = phi(1); % Phi Euler Angle [rad]
condition(8) = 0; % Theta Euler Angle [rad]
condition(9) = 0; % Psi Euler Angle [rad]
condition(10) = 0; % Angular velocity about the x-axis [rad/s]
condition(11) = 0; % Angular Velocity about the y-axis [rad/s]
condition(12) = 0; % Angular Velocity about the z-axis [rad/s]
%% Solving Differential Equations w/ ODE45
f1 = (Forcemag)/4; % Force for steady Level Flight about Motor 1
f2 = (Forcemag)/4; % Force for steady Level Flight about Motor 1
f3 = (Forcemag)/4; % Force for steady Level Flight about Motor 1
f4 = (Forcemag)/4; % Force for steady Level Flight about Motor 1
[t,z] = ode45('main',[0 1],condition);
plot3(z(:,1),z(:,2),z(:,3),'-o')
title('Trajectory of Quad-Copter')
xlabel('N Displacement [m]')
ylabel('E Displacement [m]')
zlabel('-D Displacement [m]')
axis equal
%% end
```

4. *Problem 10 Part 2*

```
%% Jack Lambert
% Aircraft Dynmaics Homework 2
% Problem 10
global mass g R k eta zeta alpha beta Ix Iy Iz f1 f2 f3 f4
%% Constants
mass = 68/1000; % [kg]
L_arm = 6/100; % [m]
eta = 1*10^(-3); % Aerodynamic Coefficient for drag [N /(m/s)^2]
zeta = 3*1^(-3);  % Aerodynamic Coefficient for drag [N /(m/s)^2]
alpha = 2*10^(-6); % Aerodynamic Coefficient for drag [N /(rad/s)^2]
beta = 1*10^(-6); % Aerodynamic Coefficient for drag [N /(rad/s)^2]
Ix = 6.8*10^(-5); % MOI about x-axis [kg*m^2]
Iy = 9.2*10^(-5); % MOI about x-axis [kg*m^2]
Iz = 1.35*10^(-4); % MOI about x-axis [kg*m^2]
R = sqrt(2)/2*L_arm; % Distance to COG [m]
k = 0.0024; % [m]
g = 9.81; % [m/s^2]
%% Initial Conditions
y_E = 5; % Interial y-component of Velocity
syms phi0
eqn = eta*y_E^2*cos(phi0)^2 == mass*g*sin(phi0);
phi = double(solve(eqn,phi0));
phi(imag(phi) ~= 0) = [];
i = find(phi<pi/2);
phi = phi(i); % [rad]

v_E = y_E*cos(phi);
w_E = -y_E*sin(phi);

Forcemag = -zeta*w_E^2*sign(w_E)+mass*g*cos(phi);

condition(1) = 0; % N - location [m]
condition(2) = 0; % E - location [m]
condition(3) = 0; % -D - location [m]
condition(4) = 0; % u - component of velocity [m/s]
condition(5) = v_E; % v compenent of velocity [m/s]
condition(6) = w_E; % w component of velocity [m/s]
% Rotational Motion
condition(7) = phi(1); % Phi Euler Angle [rad]
condition(8) = 0; % Theta Euler Angle [rad]
condition(9) = 0; % Psi Euler Angle [rad]
condition(10) = 0; % Angular velocity about the x-axis [rad/s]
condition(11) = 0; % Angular Velocity about the y-axis [rad/s]
condition(12) = 0; % Angular Velocity about the z-axis [rad/s]
%% Solving Differential Equations w/ ODE45
f1 = (Forcemag)/4; % Force for steady Level Flight about Motor 1
f2 = (Forcemag)/4; % Force for steady Level Flight about Motor 1
f3 = (Forcemag)/4; % Force for steady Level Flight about Motor 1
f4 = (Forcemag)/4; % Force for steady Level Flight about Motor 1
[t,z] = ode45('main',[0 1],condition);
plot3(z(:,1),z(:,2),z(:,3),'-o')
title('Trajectory of Quad-Copter')
xlabel('N Displacement [m]')
ylabel('E Displacement [m]')
zlabel('-D Displacement [m]')
axis equal
%% end
```

5. *Problem 11 Part 1*

```
%% Jack Lambert
% Aircraft Dynmaics Homework 2
% Problem 10
global mass g R k eta zeta alpha beta Ix Iy Iz f1 f2 f3 f4
%% Constants
mass = 68/1000; % [kg]
```

```
L_arm = 6/100; % [m]
eta = 1*10^(-3); % Aerodynamic Coefficient for drag [N /(m/s)^2]
zeta = 3*1^(-3);  % Aerodynamic Coefficient for drag [N /(m/s)^2]
alpha = 2*10^(-6); % Aerodynamic Coefficient for drag [N /(rad/s)^2]
beta = 1*10^(-6); % Aerodynamic Coefficient for drag [N /(rad/s)^2]
Ix = 6.8*10^(-5); % MOI about x-axis [kg*m^2]
Iy = 9.2*10^(-5); % MOI about x-axis [kg*m^2]
Iz = 1.35*10^(-4); % MOI about x-axis [kg*m^2]
R = sqrt(2)/2*L_arm; % Distance to COG [m]
k = 0.0024; % [m]
g = 9.81; % [m/s^2]
%% Varying Conditions for small pertubations Expirienced in None Homogenous Air
v_E_vec = randn(1,10)*.1; % Variation in the Inital Velocity of QuadCopter [m/s]
w_E_vec = randn(1,10)*.1; % Variation in the Inital Velocity of QuadCopter [m/s]
u_E_vec = rand(1,10)*.1; % Variation in the Inital Velocity of QuadCopter [m/s]
f1_vec = (randn(1,10).*.01+mass*g)*(1/4); % Varying the force from each motor
f2_vec = (randn(1,10).*.01+mass*g)*(1/4); % Varying the force from each motor
f3_vec = (randn(1,10).*.01+mass*g)*(1/4); % Varying the force from each motor
f4_vec = (randn(1,10).*.01+mass*g)*(1/4); % Varying the force from each motor
phi_vec = randn(1,10).*0.02; % Varying Inital Bank Angle
theta_vec = randn(1,10)*0.02; % Varying Inital Elevation Angle
psi_vec = randn(1,10)*0.02; % Varying Inital Azimuth Angle
for i = 1:10
    condition(1) = 0; % N - location [m]
    condition(2) = 0; % E - location [m]
    condition(3) = 0; % -D - location [m]
    condition(4) = u_E_vec(i); % u - component of velocity [m/s]
    condition(5) = v_E_vec(i); % v compenent of velocity [m/s]
    condition(6) = w_E_vec(i); % w component of velocity [m/s]
    % Rotational Motion
    condition(7) = phi_vec(i); % Phi Euler Angle [rad]
    condition(8) = theta_vec(i); % Theta Euler Angle [rad]
    condition(9) = psi_vec(i); % Psi Euler Angle [rad]
    condition(10) = 0; % Angular velocity about the x-axis [rad/s]
    condition(11) = 0; % Angular Velocity about the y-axis [rad/s]
    condition(12) = 0; % Angular Velocity about the z-axis [rad/s]
    f1 = f1_vec(i); % Force for steady Level Flight about Motor 1
    f2 = f2_vec(i); % Force for steady Level Flight about Motor 2
    f3 = f3_vec(i); % Force for steady Level Flight about Motor 3
    f4 = f4_vec(i); % Force for steady Level Flight about Motor 4
    figure(1)
    [t,z] = ode45('main',[0 3],condition);
    plot3(z(:,1),z(:,2),z(:,3))
    hold on
end
plot3(0,0,0,'-*')
title('Trajectory of Quad-Copter')
xlabel('N Displacement [m]')
ylabel('E Displacement [m]')
zlabel('-D Displacement [m]')
axis equal
hold off
%% end
```

6. *Problem 11 Part 2*

```
%% Jack Lambert
% Aircraft Hw 2
% Problem 11 Part 2
%% Motor Commands
timeMotor = rt_motor.time(:);
Motor1 = (rt_motor.signals.values(:,1)*13840.4).^(1/2); % Motor Rotation Rate [Rad/s]
Motor2 = (abs(rt_motor.signals.values(:,2)*13840.4)).^(1/2); % Motor Rotation Rate [Rad/s]
Motor3 = (rt_motor.signals.values(:,3)*13840.4).^(1/2); % Motor Rotation Rate [Rad/s]
Motor4 = (abs(rt_motor.signals.values(:,4)*13840.4)).^(1/2); % Motor Rotation Rate [Rad/s]
%% Translation
```

```
% Position
timeest = rt_estim.time(:);
xdata = rt_estim.signals.values(:,1); % X-Position [m]
ydata = rt_estim.signals.values(:,2); % Y-Position [m]
zdata = rt_estim.signals.values(:,3); % Z-Position [m]
% Velocity
Vx = rt_estim.signals.values(:,7); % X-Velocity [m/s]
Vy = rt_estim.signals.values(:,8); % Y-Velocity [m/s]
Vz = rt_estim.signals.values(:,9); % Z-Velocity [m/s]
%% Rotation
% Attitude
yaw = rt_estim.signals.values(:,4); % [Rad]
pitch = rt_estim.signals.values(:,5); % [Rad]
roll = rt_estim.signals.values(:,6); % [Rad]
% Angular Rates
p = rt_estim.signals.values(:,10); % Body Fixed frame rotation about x-axis [Rad/s]
q = rt_estim.signals.values(:,11); % Body Fixed frame rotation about y-axis[Rad/s]
r = rt_estim.signals.values(:,12); % Body Fixed frame rotation about z-axis [Rad/s]
%% Plots of Translation vs. Time in Correlation to Motor Controls
% Motor Controls
figure(1)
subplot(3,1,1)
plot(timeMotor,Motor1)
hold on
plot(timeMotor,Motor2)
plot(timeMotor,Motor3)
plot(timeMotor,Motor4)
title('Motor Controls vs. Time')
xlabel('Time [s]')
ylabel('Motor Rotation Rate [rad/s]')
legend('Motor 1',' |Motor 2|','Motor 3','|Motor 4|')
% Position
subplot(3,1,2,'replace')
plot(timeest,xdata)
hold on
plot(timeest,ydata)
hold off
title('Position vs. Time')
ylabel('Position [m]')
xlabel('Time [s]')
legend('X-Position','Y-Position')
% Velocity
subplot(3,1,3,'replace')
plot(timeest,Vx)
hold on
plot(timeest,Vy)
hold off
title('Velocity vs. Time')
ylabel('Velocity [m]')
xlabel('Time [s]')
legend('X-Velocity','Y-Velocity')
%% Plots of Rotation vs. Time in Correlation to Motor Controls
% Motor Controls
figure(2)
subplot(3,1,1)
plot(timeMotor,Motor1)
hold on
plot(timeMotor,Motor2)
plot(timeMotor,Motor3)
plot(timeMotor,Motor4)
title('Motor Controls vs. Time')
xlabel('Time [s]')
ylabel('Motor Rotation Rate [rad/s]')
legend('Motor 1',' |Motor 2|','Motor 3','|Motor 4|')
% Attitude
subplot(3,1,2,'replace')
plot(timeest,roll)
hold on
plot(timeest,pitch)
plot(timeest,yaw)
```

```
hold off
title('Attitude vs. Time')
ylabel('Orientation [rad]')
xlabel('Time [s]')
legend('Roll','Pitch','Yaw')
% Anglar Velocity in Body fixed Frame
subplot(3,1,3,'replace')
plot(timeest,p)
hold on
plot(timeest,q)
plot(timeest,r)
hold off
title('Angular Velocity vs. Time')
ylabel('Angular Velocity [rad/s]')
xlabel('Time [s]')
legend('\omega_{x}','\omega_{y}','\omega_{z}')
```