Table of Contents

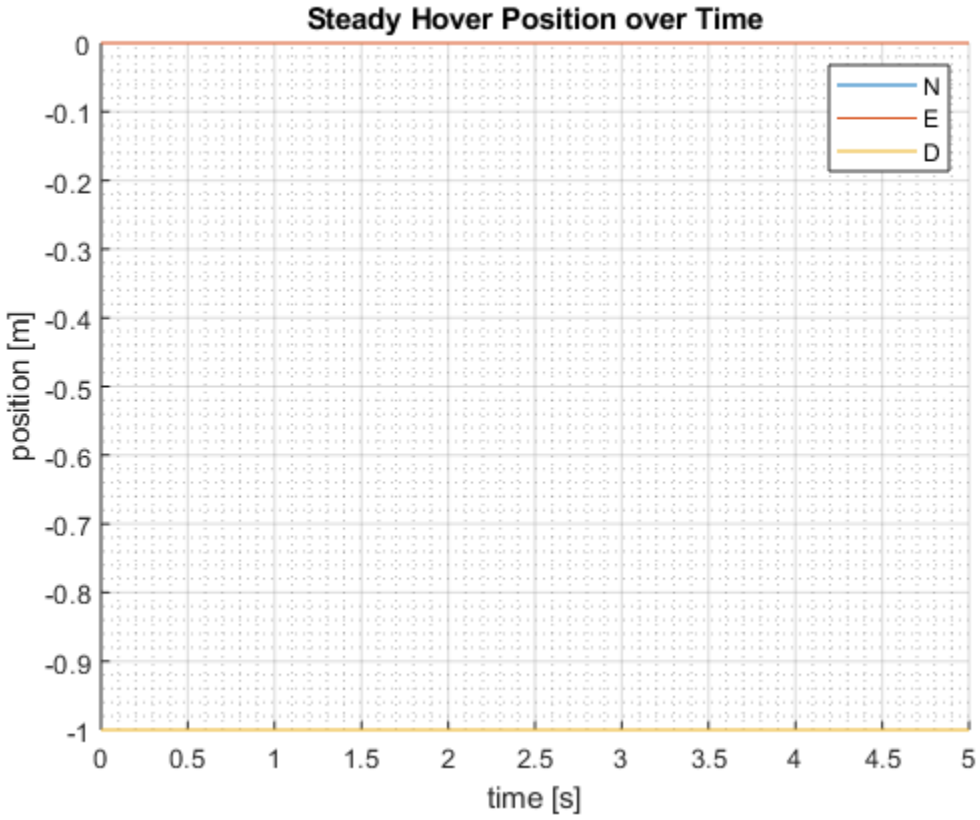

```
%Quadcopter simulation including attitude dynaics and kinematics using
%azimuth, elevation, and bank euler angle attitude rep.
clear all; close all; clc
global g m rad I_x I_y I_z k alpha eta f f1 f2 f3 f4 X Y Z
%Constants for every scenario
g = 9.81; %acceleration due to gravity, m/s^2
m = .068; %kg
rad = .06; %m
I_x = 6.8e-5; %Components of moment of inertia matrix
I_y = 9.2e-5;
I_z = 1.35e-4;
k = .0024;
alpha = 2e-6;
eta = 1e-3;
```

Problem 7 steady hover scenario w/o aero forces and moments

```
%forces for steady hover
f = -m*g;
f1 = f/4;
f2 = f/4;
f3 = f/4;
f4 = f/4;
%initial conditions for steady hover
X = 0;
Y = 0;
Z = -m*g;
u_E = 0;
v_E = 0;
w_E = 0;
p = 0;
q = 0;
r = 0;
phi = 0;
theta = 0;
psi = 0;
N = 0;
E = 0;
D = -1;
```

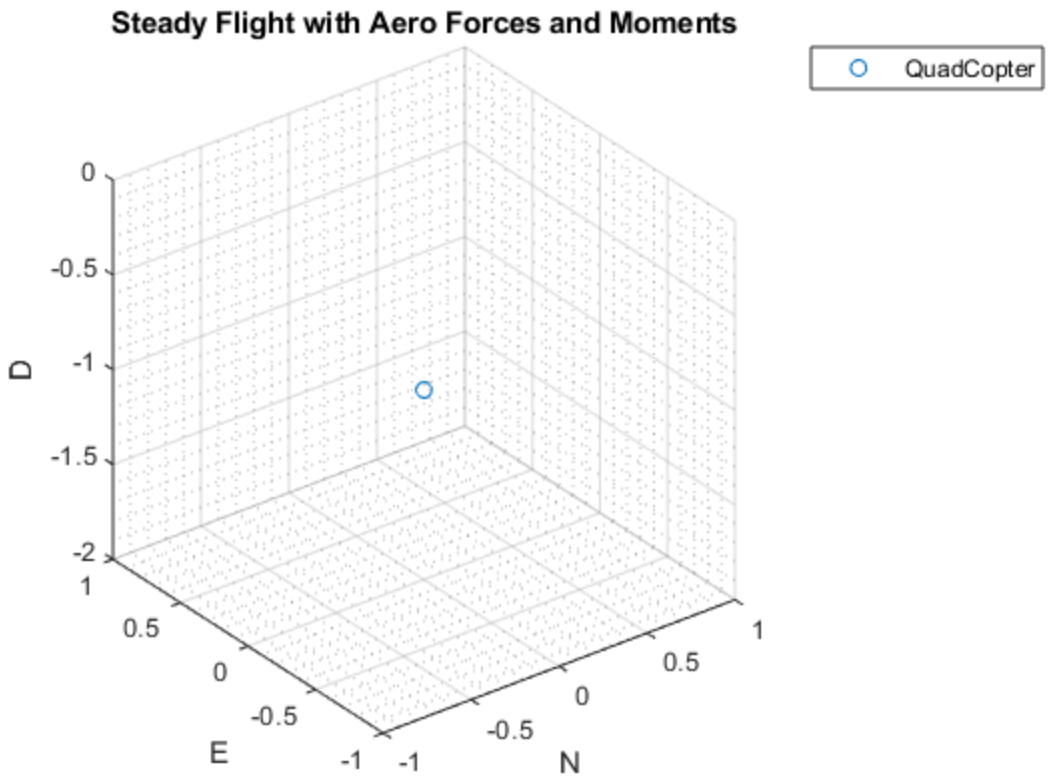
```
vel = [u_E v_E w_E];
omega = [p q r];
euler = [psi theta phi];
pos = [N E D];
inish_condish = [vel omega euler pos 0];
tspan = [0 5];
[t,y] = ode45('odequad',tspan,inish_condish);
figure
plot3(y(:,10),y(:,11),y(:,12),'o')
grid on
grid minor
title('Steady Flight, no Aero. Forces or Moments')
xlabel('N')
ylabel('E')
zlabel('D')
legend('Quad Copter')
figure
hold on
grid on
grid minor
plot(t,y(:,10))
plot(t,y(:,11))
plot(t,y(:,12))
title('Steady Hover Position over Time')
xlabel('time [s]')
ylabel('position [m]')
legend('N','E','D')
```

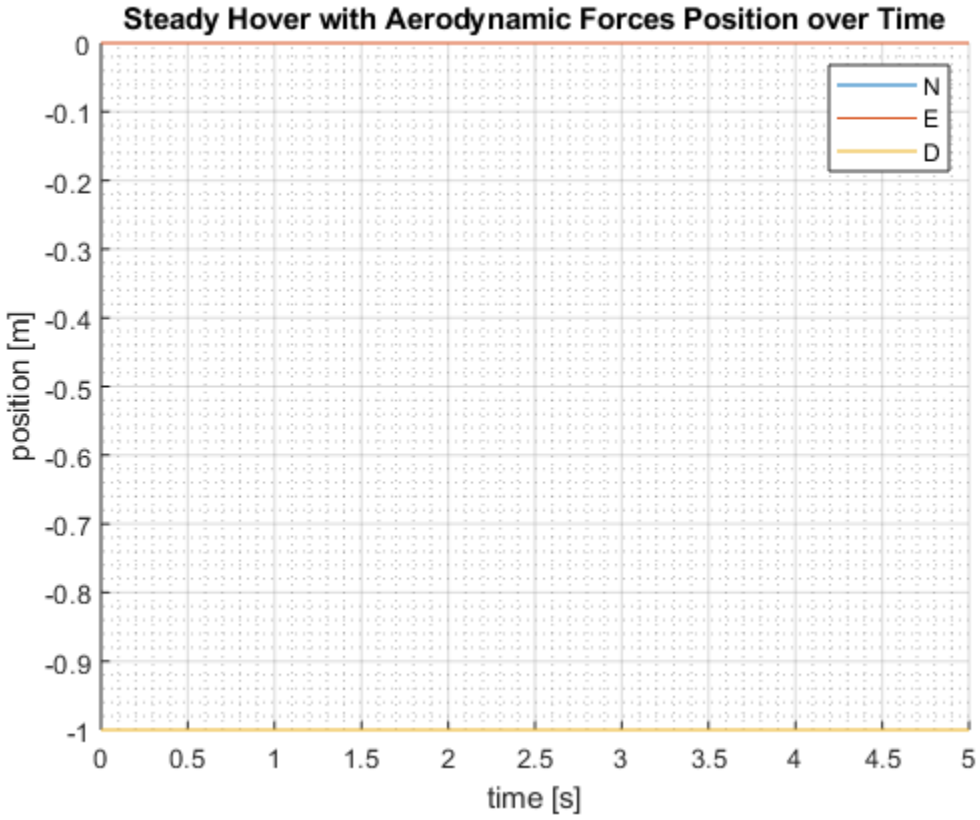

Steady Flight, no Aero. Forces or Moments
Quad Copter
D
0
-0.5
-1
-1.5
-2
E
1
0.5
0
-0.5
-1
N
1
0.5
0
-0.5
-1


Steady Hover Position over Time
N
E
D
position [m]
0
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
-0.7
-0.8
-0.9
-1
time [s]
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

Problem 8 adding aero forces/moments and steady translation

```
%initial conditions, adding in aero forces and moments, verify that
 this
%does not alter trim state
clear t y
u_E = 0;
v_E = 0; %m/s
w_E = 0;
X = 0 + -eta*u_E^2;
Y = 0 + -eta*v_E^2;
Z = -m*g + -eta*w_E^2;
p = 0;
q = 0;
r = 0;
phi = 0;
theta = 0;
psi = 0;
N = 0;
E = 0;
D = -1;
vel = [u_E v_E w_E];
omega = [p q r];
euler = [psi theta phi];
pos = [N E D];
inish_condish = [vel omega euler pos 1];
tspan = [0 5];
[t,y] = ode45('odequad',tspan,inish_condish);
figure
plot3(y(:,10),y(:,11),y(:,12),'o')
grid on
grid minor
title('Steady Flight with Aero Forces and Moments')
xlabel('N')
ylabel('E')
zlabel('D')
legend('QuadCopter')
figure
hold on
grid on
grid minor
plot(t,y(:,10))
plot(t,y(:,11))
plot(t,y(:,12))
title('Steady Hover with Aerodynamic Forces Position over Time')
xlabel('time [s]')
ylabel('position [m]')
legend('N','E','D')
```

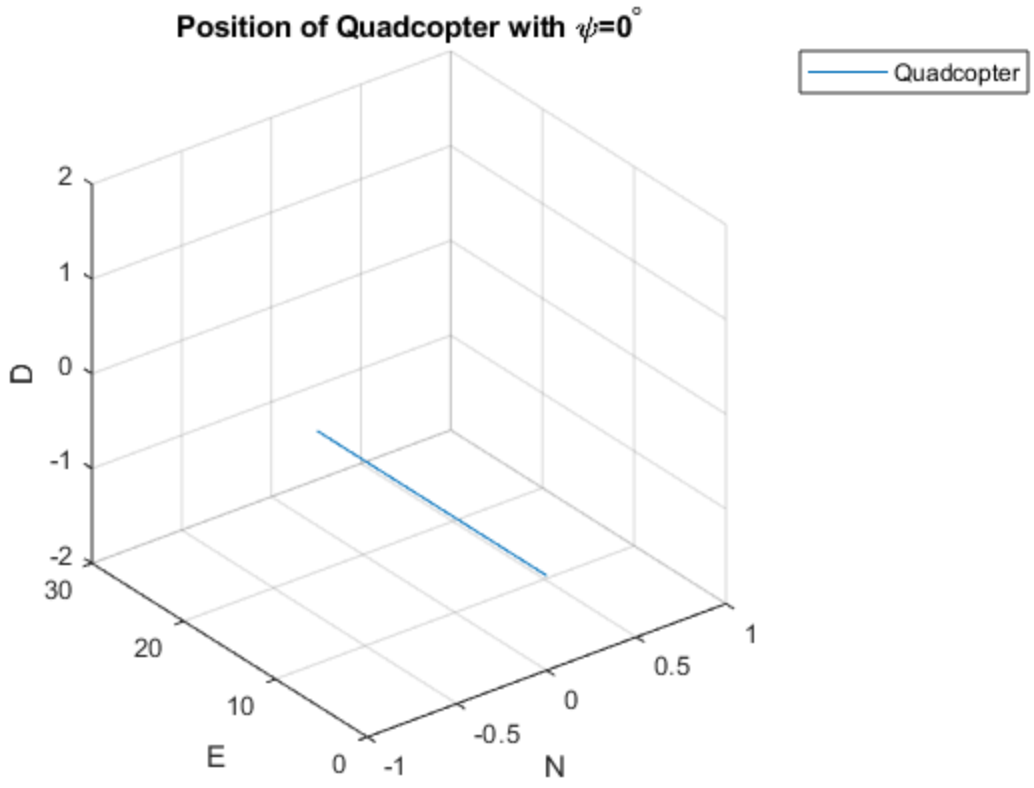

Steady Flight with Aero Forces and Moments
QuadCopter
D
0
-0.5
-1
-1.5
-2
E
1
0.5
0
-0.5
-1
N
1
0.5
0
-0.5
-1

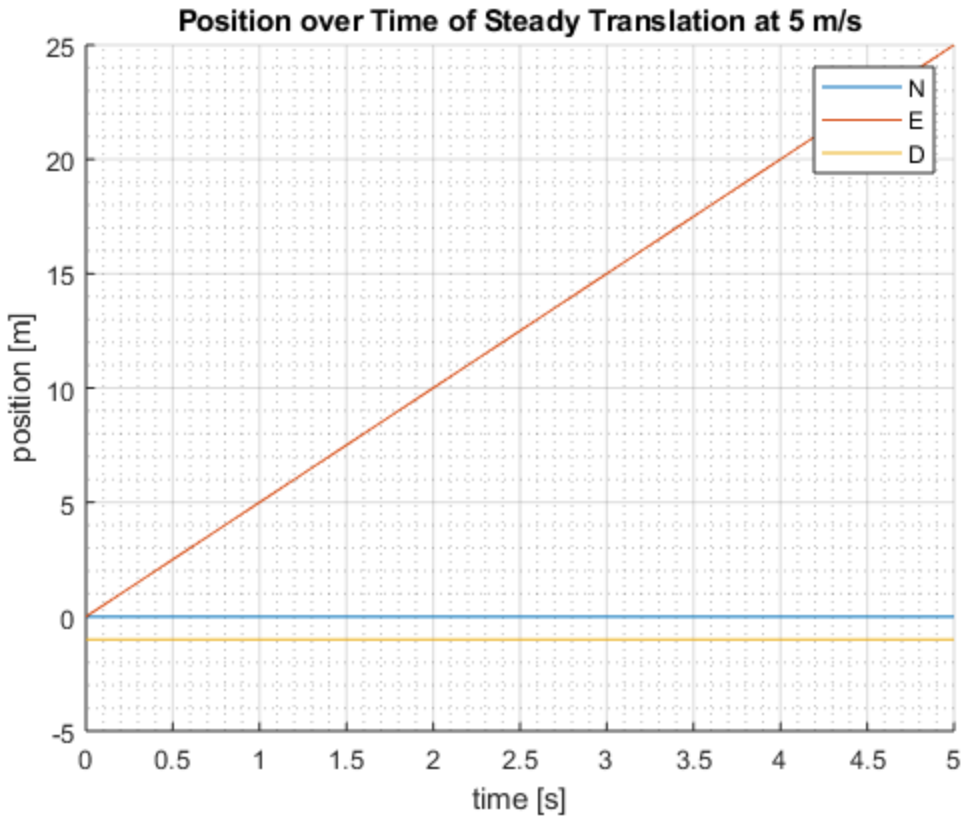

Steady Hover with Aerodynamic Forces Position over Time
N
E
D
position [m]
0
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
-0.7
-0.8
-0.9
-1
time [s]
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

steady translation east @ 5 m/s

```
%determine angle and force magnitude for steady translation east @ 5
 m/s,
%azimuth of 0 deg.
clear t y
u_E = 0;
v_E = 5;
w_E = 0;
f_d = -eta*v_E^2;
f_up = -m*g;
f = -sqrt(f_d^2 + f_up^2);
phi = atan(f_d/f_up);
u_E = 0;
v_E = 5*cos(phi);
w_E = -5*sin(phi);
f1 = f/4;
f2 = f/4;
f3 = f/4;
f4 = f/4;
aero_forces = -eta*norm([u_E v_E w_E])*[u_E v_E w_E];
X = aero_forces(1);
Y = aero_forces(2);
Z = f + aero_forces(3);
p = 0;
q = 0;
r = 0;
psi = 0;
theta = 0;
phi = atan(f_d/f_up);
N = 0;
E = 0;
D = -1;
vel = [u_E v_E w_E];
omega = [p q r];
euler = [psi theta phi];
pos = [N E D];
tspan = [0 5];
inish_condish = [vel omega euler pos 1];
[t,y] = ode45('odequad',tspan,inish_condish);
figure
plot3(y(:,10),y(:,11),y(:,12));
title('Position of Quadcopter with \psi=0^\circ')
grid on
grid minor
xlabel('N')
ylabel('E')
zlabel('D')
zlim([-2 2])
legend('Quadcopter')
grid on
grid minor
figure
```

```
hold on
grid on
grid minor
plot(t,y(:,10))
plot(t,y(:,11))
plot(t,y(:,12))
title('Position over Time of Steady Translation at 5 m/s')
xlabel('time [s]')
ylabel('position [m]')
legend('N','E','D')
```

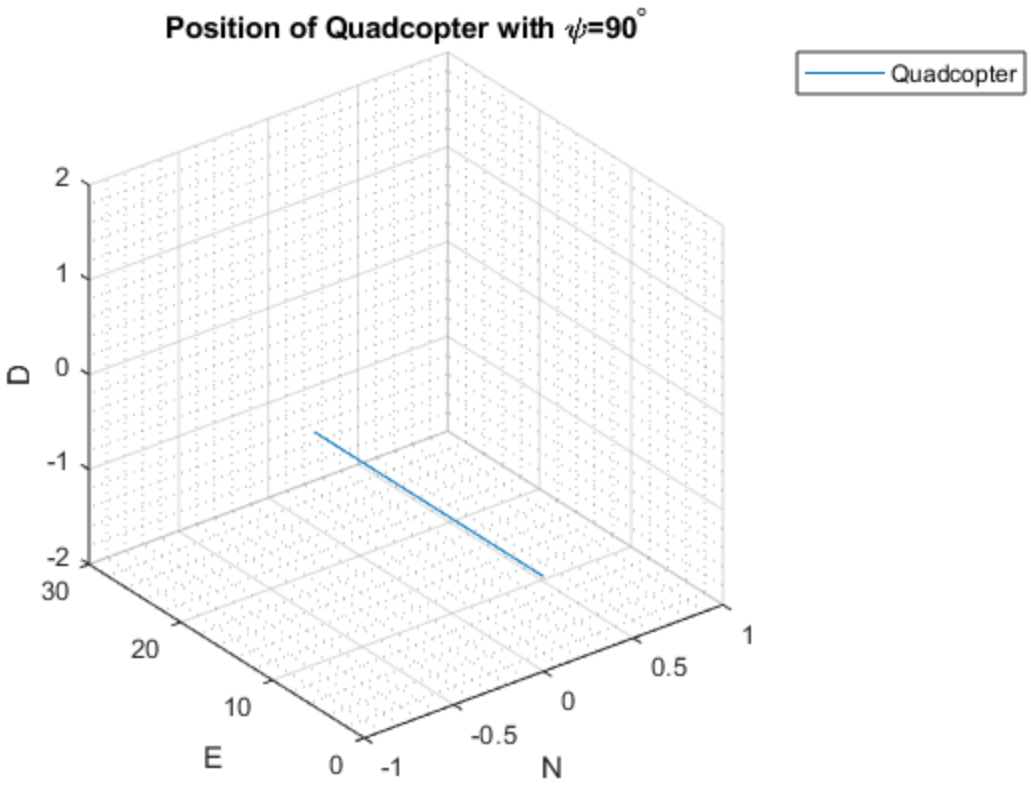


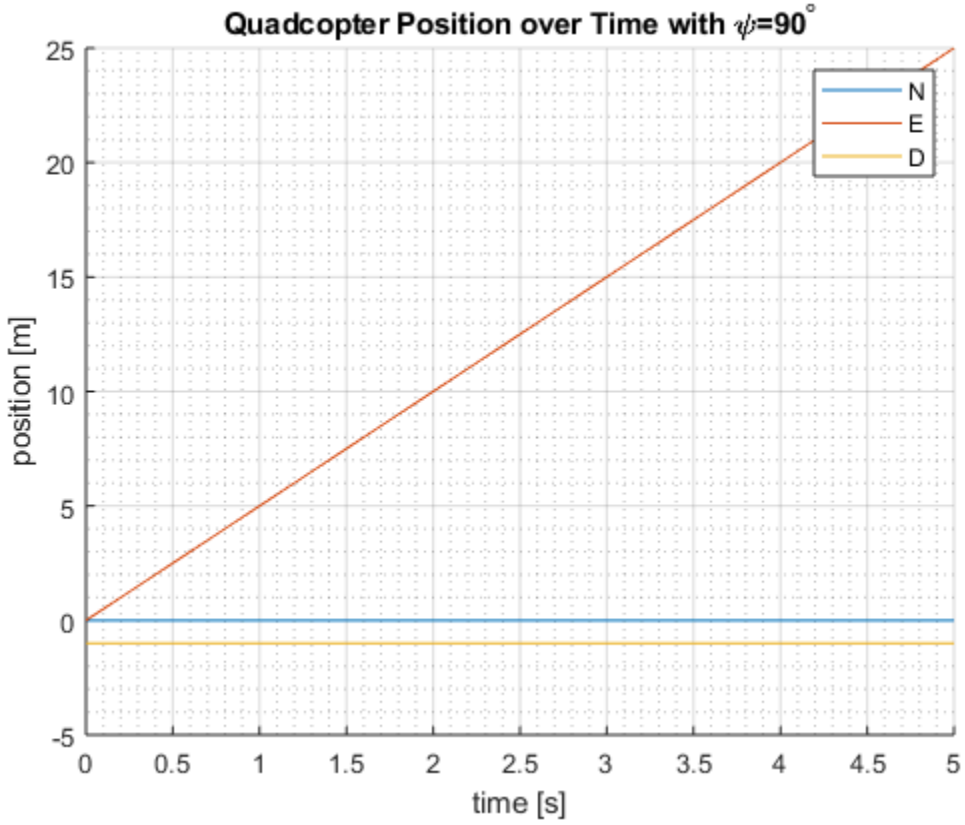


initial conditions for steady translation 5 m/s east with an azimuth of 90

```
%deg
clear t y
u_E = 0;
v_E = 5;
w_E = 0;
f_d = -eta*v_E^2;
f_up = -m*g;
f = -sqrt(f_d^2 + f_up^2);
theta = -atan(f_d/f_up);
u_E = 5*cos(theta);
v_E = 0;
w_E = 5*sin(theta);
f1 = f/4;
f2 = f/4;
f3 = f/4;
f4 = f/4;
aero_forces = -eta*norm([u_E v_E w_E])*[u_E v_E w_E];
X = aero_forces(1);
Y = aero_forces(2);
Z = f + aero_forces(3);
p = 0;
```

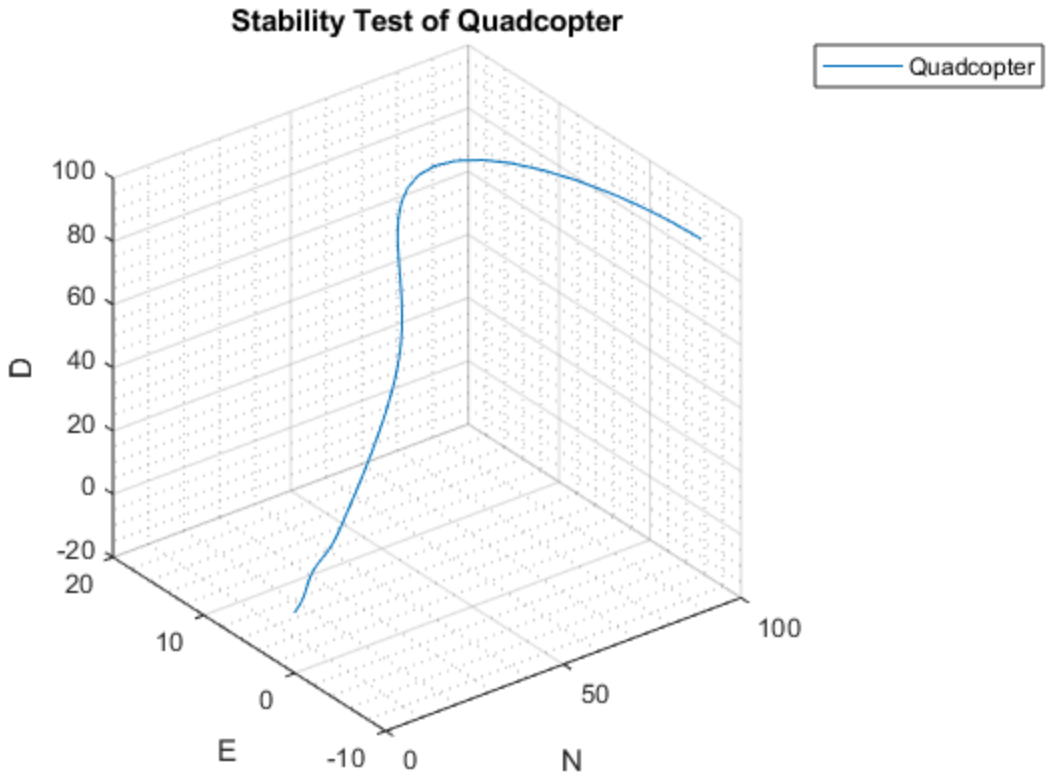
```
q = 0;
r = 0;
psi = (pi/2);
phi = 0;
N = 0;
E = 0;
D = -1;
vel = [u_E v_E w_E];
omega = [p q r];
euler = [psi theta phi];
pos = [N E D];
inish_condish = [vel omega euler pos 1];
tspan = [0 5];
[t,y] = ode45('odequad',tspan,inish_condish);
% close all
figure
plot3(y(:,10),y(:,11),y(:,12))
title('Position of Quadcopter with \psi=90^\circ')
xlabel('N')
ylabel('E')
zlabel('D')
xlim([-1 1])
zlim([-2 2])
legend('Quadcopter')
grid on
grid minor
figure
hold on
grid on
grid minor
plot(t,y(:,10))
plot(t,y(:,11))
plot(t,y(:,12))
title('Quadcopter Position over Time with \psi=90^\circ')
xlabel('time [s]')
ylabel('position [m]')
legend('N','E','D')
```

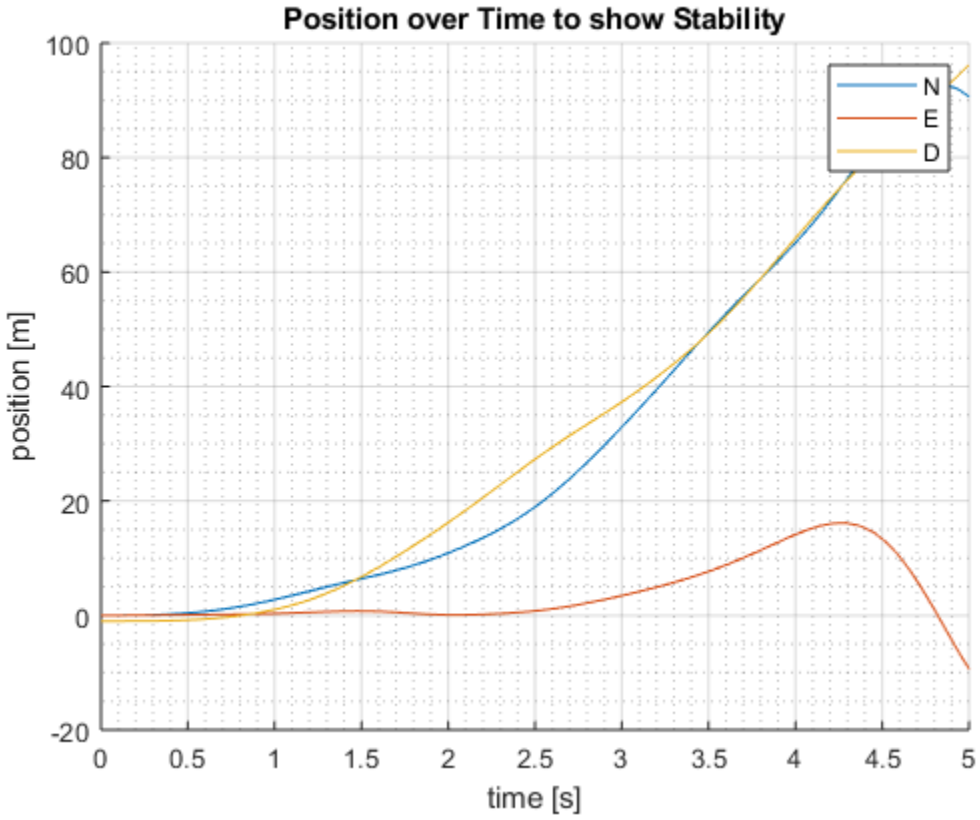

Position of Quadcopter with ψ=90°
Quadcopter
D
2
1
0
-1
-2
30
20
10
E
0
-1
-0.5
0
0.5
1
N


Quadcopter Position over Time with ψ=90°
N
E
D
position [m]
25
20
15
10
5
0
-5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
time [s]

Question 9, determine stability of steady hover through simulation, plots

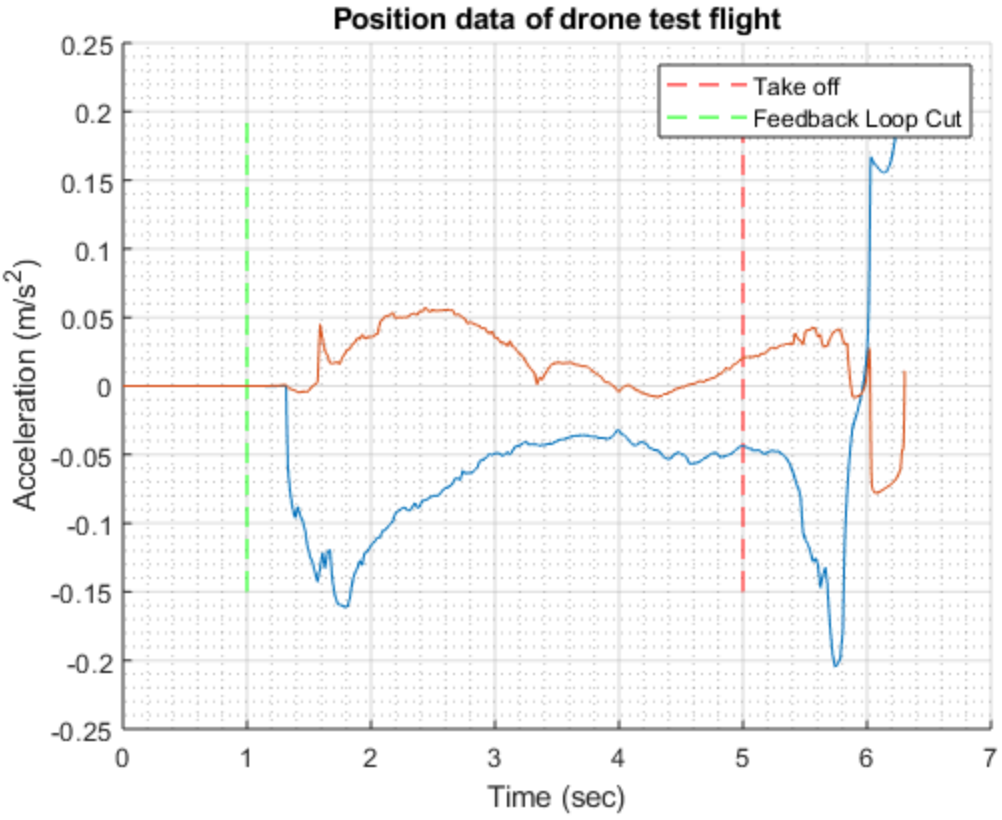
```
%of hardware data, and rotation and translation plots
clear t y
u_E = 0;
v_E = 0; %m/s
w_E = 0;
X = 0 + -eta*u_E^2;
Y = 0 + -eta*v_E^2;
Z = -m*g + -eta*w_E^2;
p = -1;
q = -1.5;
r = 1.7;
phi = 0;
theta = 0;
psi = 0;
N = 0;
E = 0;
D = -1;
vel = [u_E v_E w_E];
omega = [p q r];
euler = [psi theta phi];
pos = [N E D];
inish_condish = [vel omega euler pos 1];
tspan = [0 5];
[t,y] = ode45('odequad',tspan,inish_condish);
figure
plot3(y(:,10),y(:,11),y(:,12))
title('Stability Test of Quadcopter')
xlabel('N')
ylabel('E')
zlabel('D')
legend('Quadcopter')
grid on
grid minor
figure
hold on
grid on
grid minor
plot(t,y(:,10))
plot(t,y(:,11))
plot(t,y(:,12))
title('Position over Time to show Stability')
xlabel('time [s]')
ylabel('position [m]')
legend('N','E','D')
```

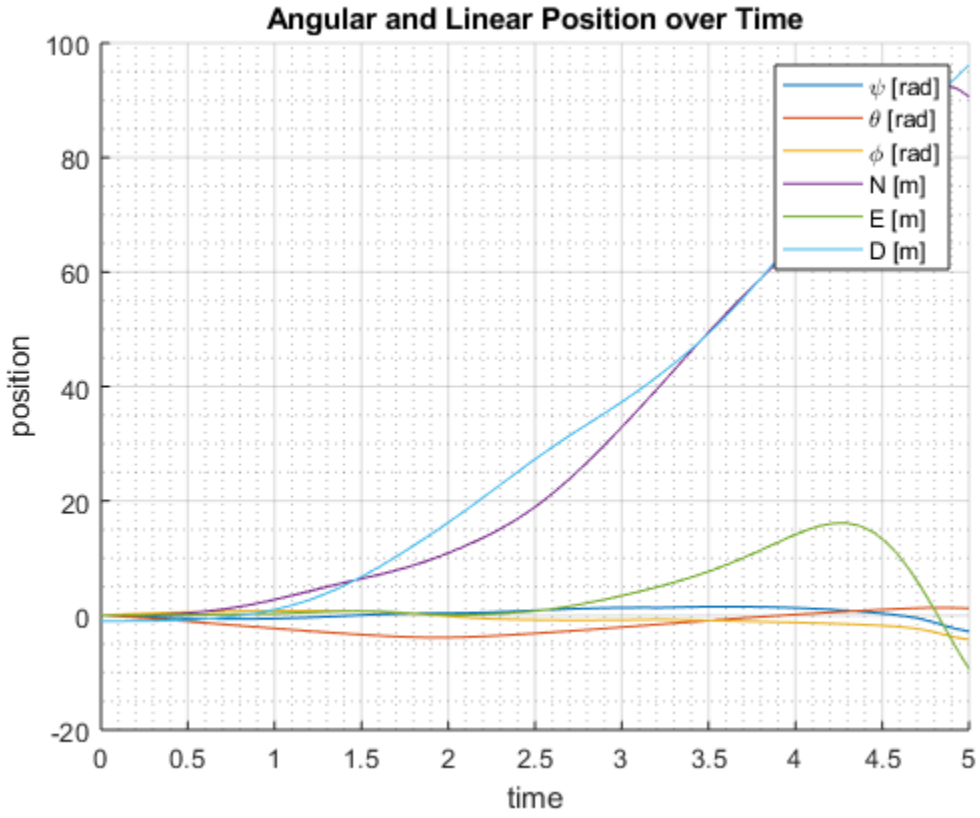

Stability Test of Quadcopter
Quadcopter
D
100
80
60
40
20
0
-20
E
20
10
0
-10
N
0
50
100

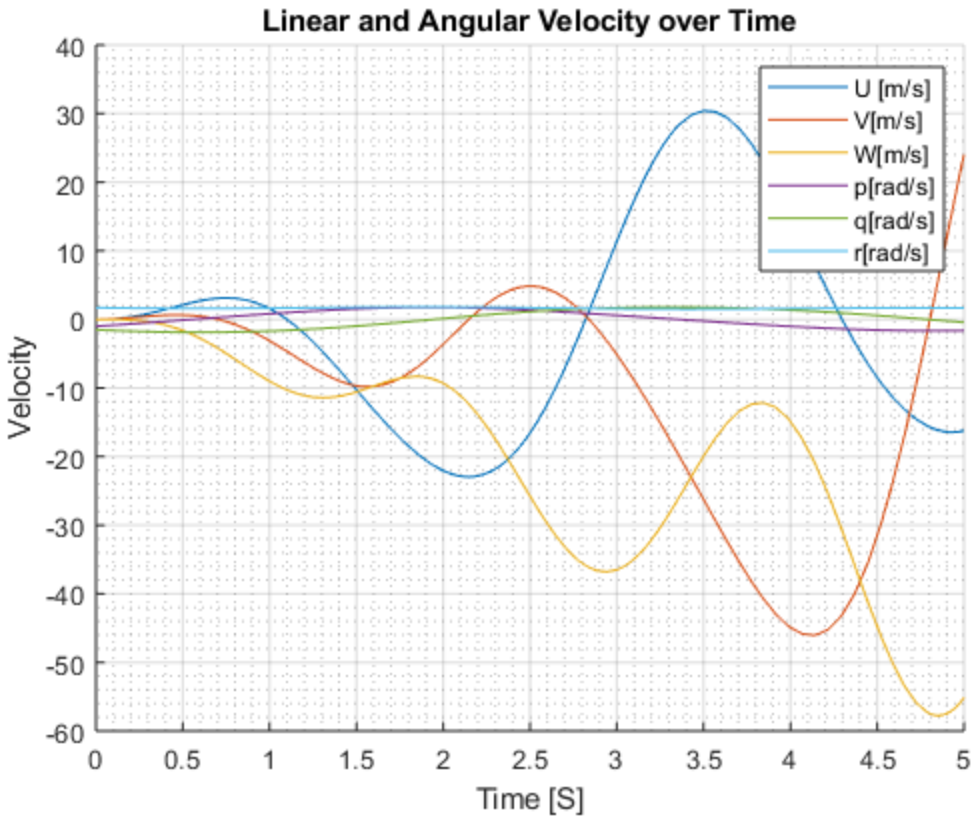

Position over Time to show Stability
N
E
D
position [m]
100
80
60
40
20
0
-20
time [s]
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

Plots of stability of quadcopter from hardware data

```
data = load('W_1213_A2.mat');
figure
hold on
grid on
grid minor
plot([5,5],[-.15,.2],'--r')
plot([1,1],[-.15,.2],'--g')
plot(data.rt_posref.time,data.rt_posref.signals.values(:,7))
plot(data.rt_posref.time,data.rt_posref.signals.values(:,8))
legend('Take off','Feedback Loop Cut')
title('Position data of drone test flight')
xlabel('Time (sec)')
ylabel('Acceleration (m/s^2)')
figure
hold on
grid on
grid minor
plot(t,y(:,7))
plot(t,y(:,8))
plot(t,y(:,9))
plot(t,y(:,10))
plot(t,y(:,11))
plot(t,y(:,12))
title('Angular and Linear Position over Time')
xlabel('time')
ylabel('position')
legend('\psi [rad]','\theta [rad]','\phi [rad]','N [m]','E [m]','D
 [m]')
figure
hold on
grid on
grid minor
plot(t,y(:,1))
plot(t,y(:,2))
plot(t,y(:,3))
plot(t,y(:,4))
plot(t,y(:,5))
plot(t,y(:,6))
title('Linear and Angular Velocity over Time')
xlabel('Time [S]')
ylabel('Velocity')
legend('U [m/s]','V[m/s]','W[m/s]','p[rad/s]','q[rad/s]','r[rad/s]')
```


Position data of drone test flight
Take off
Feedback Loop Cut
Acceleration (m/s^2)
Time (sec)
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
-0.25
0
1
2
3
4
5
6
7


Angular and Linear Position over Time
ψ [rad]
θ [rad]
ϕ [rad]
N [m]
E [m]
D [m]
position
time
100
80
60
40
20
0
-20
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5



Published with MATLAB® R2018b