FRTN65 Modeling and Learning from Data

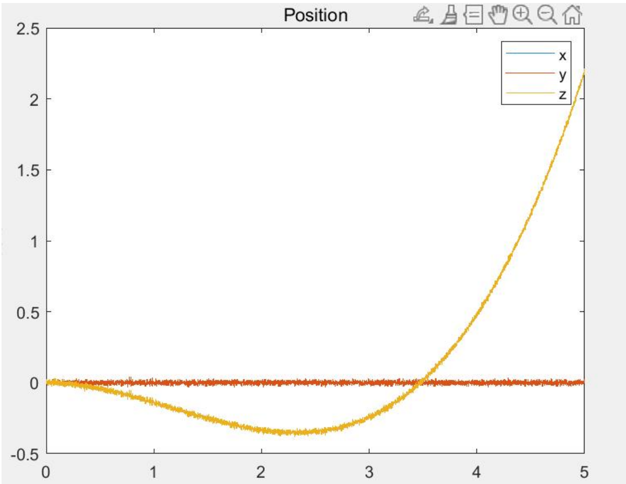
Laboratory Exercise 3

Learning Quadcopter Flight Dynamics

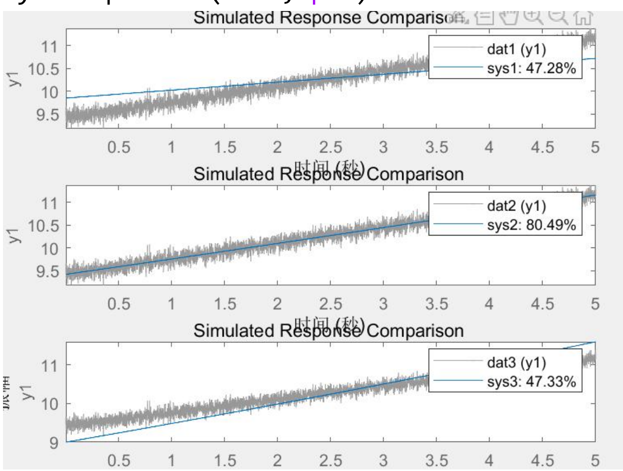
Task 1

```
Omega_in.signals.values =
linspace(1700,2000,length(Omega_in.time))'*ones(1,4)
```

The input trajectory Ω is chosen because it shows the process that rotor speed increasing causes the flight change from going down to going up. And it can also be used to comparing the three models.




```
dat1 = iddata(out.acc.data(:,3)+g,4*out.Omega.data(:,1).^1,inner_h);
dat2 = iddata(out.acc.data(:,3)+g,4*out.Omega.data(:,1).^2,inner_h);
dat3 = iddata(out.acc.data(:,3)+g,4*out.Omega.data(:,1).^3,inner_h);
sys1 = process(dat1,'p0')
sys2 = process(dat2,'p0')
sys3 = process(dat3,'p0')
```



It shows that the second model is the correct one.

```
k_est = sys2.Kp*m
```

k_est =

    2.2003e-08

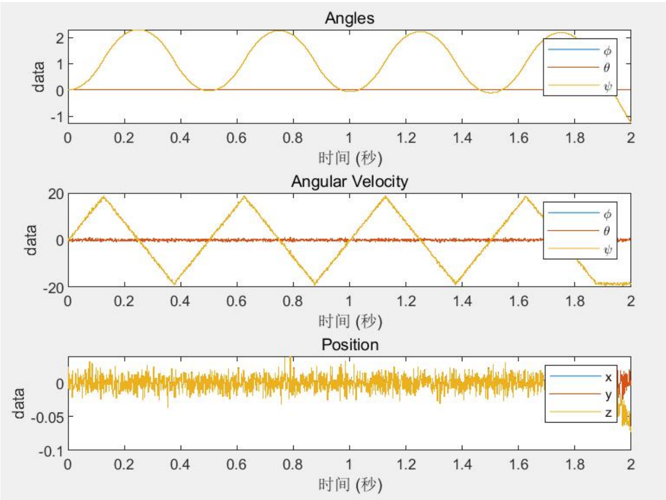
k =

    2.2000e-08

The estimation of k is close to the true value of k.

Task 2

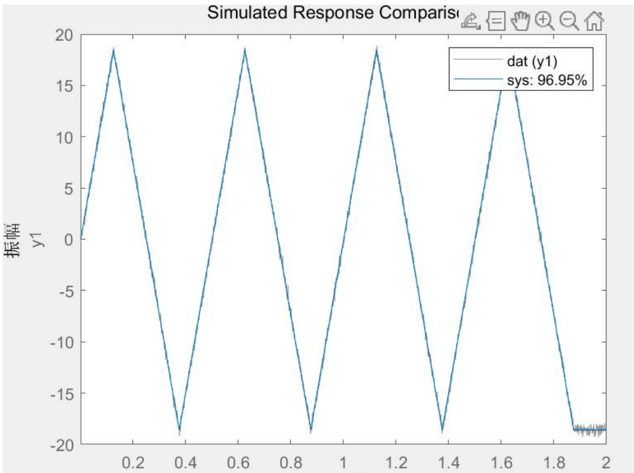
```
Omega1 = sqrt((m*g)/(k*(2*c^2+2)));
c = 1.2
segments = 8
```



I choose the value of c and segments to make the estimating value closest to the true value.

The quadcopter hovers and the yaw angle stays in the interval $(-\pi, \pi)$.

This experiment can not be used to find both b and I3 because in the equation constructed by torque $\tau\psi$, there are two unknown value b and I3, we can only estimate one of them with another when it is known.




```
b_est = I(3)*sys.Kp
```

b_est =   1.9983e-09

true_b = 2.0000e-09

Task 3

```
Omega_H = sqrt(m*g/(4*k));
c = 1.02;
Omega2 = sqrt(2*Omega_H^2/(1+c^2));
segments = 10;
dat = iddata(Phidot,out.Omega.data(:,4).^2-out.Omega.data(:,2).^2,inner_h);
sys = procest(dat,'p0I')
I1_est = k*l/sys.Kp
```

I1_est =

    1.6606e-05

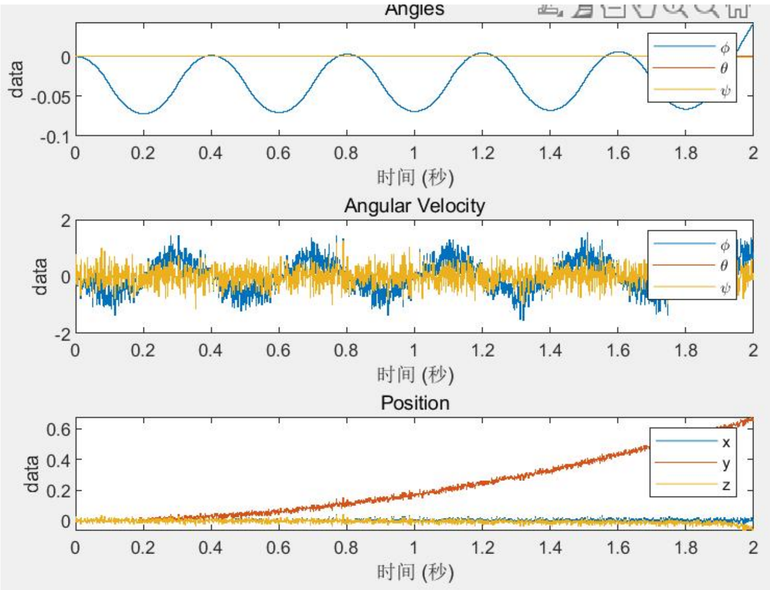
true_I =

    1.6600e-05

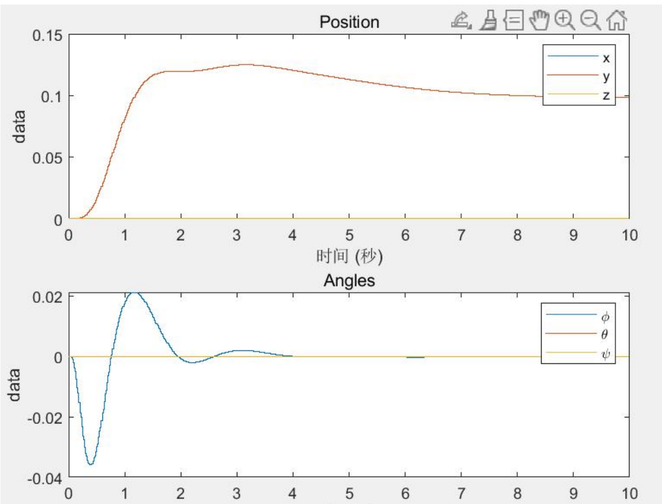
I1 estimated value is close to the true value. I2 is equal to I1.

Ω_H is the value which makes the quadcopter hover so $mg=4k\Omega^2$

The value of c and segments are chosen to make the angle φ stays quite small so that the quadcopter is stable.



Task 4

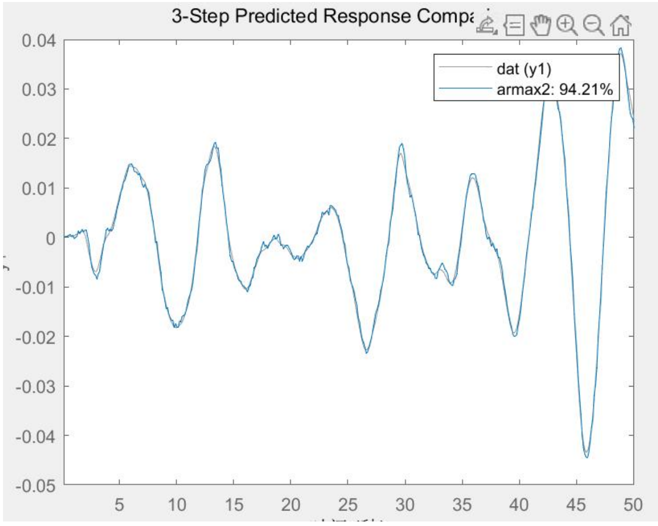


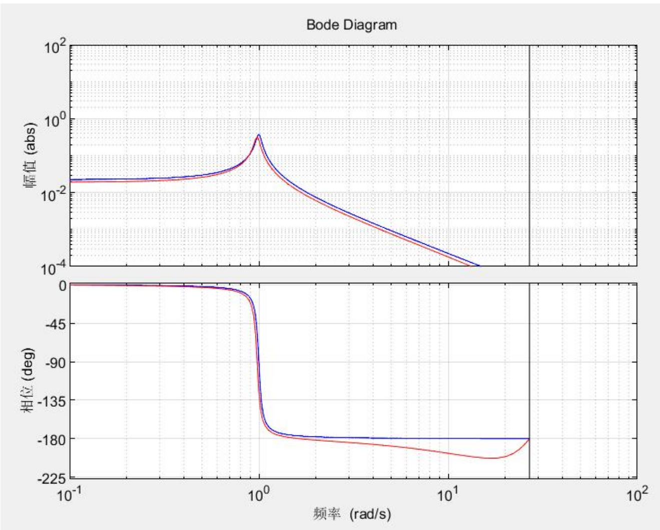
With the help of this z controller, the quadcopter moves over 0.1m in y direction while maintain the same position in z and x axis. It works well.

Task 5

```
R = 7
armax2 = armax(dat,[2,1])
```

I tried different values for R. 7 gave the best results. The model structure for the ARMA Estimation is a second-order ARMA model with a first-order C polynomial. The estimation of the disturbance model is good.





$$\frac{2.625 s + 102.5}{s^2 + 0.05789 s + 0.9608}$$

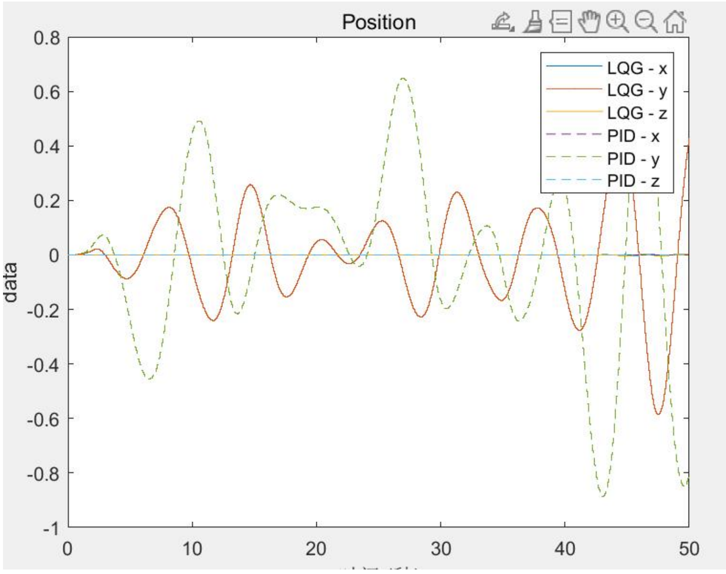
The true denominator in wind.denum is 1, 0.06, 1. They are very close.

Task 6

$\omega 0 = 0.9608$

epsilon = 0.02953

The result of LQG controller is much better than the PID controller, the effect of disturbance of wind is decreased a lot.


Position
LQG - x
LQG - y
LQG - z
PID - x
PID - y
PID - z
data
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
10
20
30
40
50