Table of Contents


```
A = [-0.038   18.984   0      -32.174;...
     -0.001  -0.632    1       0;...
      0       -0.759   -0.518   0;...
      0        0        1       0];

B =   [10.1       0;...
         0       -0.0086;...
         0.025   -0.011;...
         0        0];
```

a)

Assuming that you can only measure the airspeed of the vehicle, is this system observable? Detectable?

```
C = [1 0 0 0];  % C Matrix

rank(obsv(A,C));
```

b)

Now assume that you can measure both the airspeed and pitch angle of the aircraft. Design an observer feedback controller that will regulate the system to its trim point (states are equal to zero). Create an observer function by solving the differentiall euqation. x_hat_dot = A*x_hat + Bu _ L(y_hat -y).

```
C = [1 0 0 0;...
     0 0 0 1];
```

c)

Using 'place', set your controller poles to p = (-5+/-1j, -3 +/- 0.14j). Set your observer poles to be 10x that of your controller poles.

```
p = [-5+j,-5-j,-3+0.14j,-3-0.14j];
K = place(A,B,p);
L = place(A',C',p*10)';
```

d)

```
zo = [20 0.1 0.1 0.3]';
```

```
z_hat_o = [22 0, 0,0]';
xo = [zo;z_hat_o];
xd = [0 0 0 0]';

[t,x] = ode45(@(t,x) aircraftDynamics(t,x,xd*0,A,B,C,K,L),[0 5],xo);

t = t(1:10:length(t));
x = x(1:10:length(x),:);
estimation_error = x(end,1:4) - x(end,5:8)
```

e)

plots

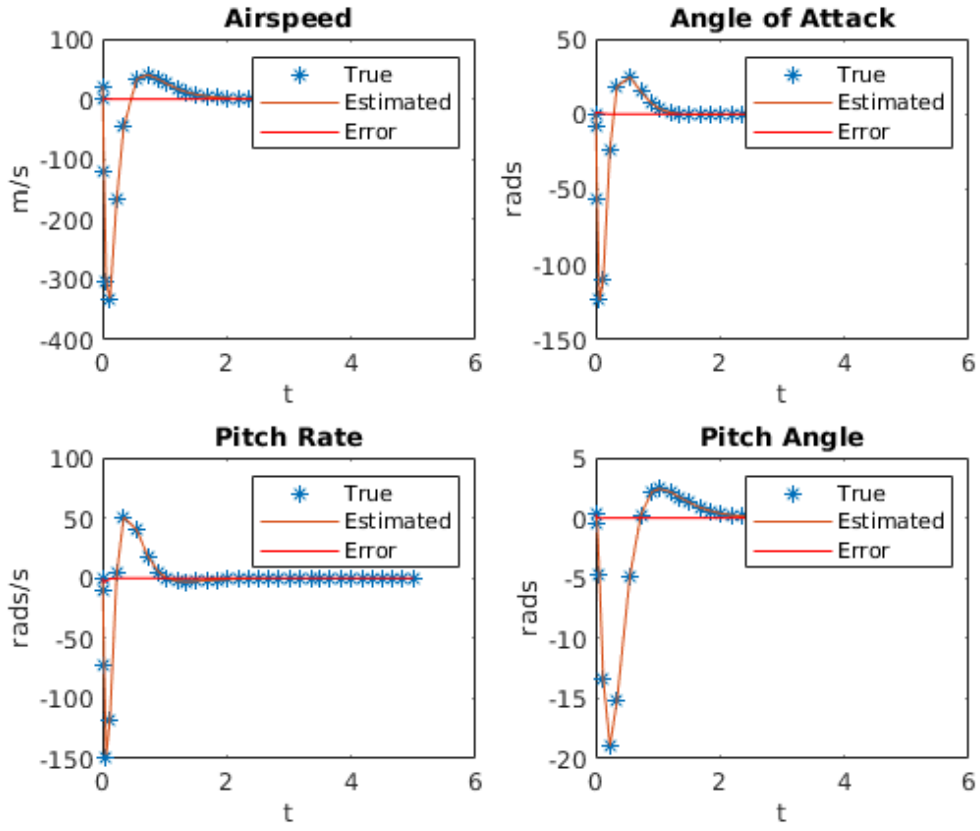
```
figure(1)
clf;
subplot(2,2,1);
plot(t,x(:,1),'*')
hold on
plot(t,x(:,5))
plot(t,x(:,1) - x(:,5),'r')
title('Airspeed')
legend('True','Estimated','Error');
xlabel('t')
ylabel('m/s')

subplot(2,2,2);
plot(t,x(:,2),'*')
hold on
plot(t,x(:,6))
plot(t,x(:,2) - x(:,6),'r')
title('Angle of Attack')
legend('True','Estimated','Error');
xlabel('t')
ylabel('rads')

subplot(2,2,3);
plot(t,x(:,3),'*')
hold on
plot(t,x(:,7))
plot(t,x(:,3) - x(:,7),'r')
title('Pitch Rate')
legend('True','Estimated','Error');
xlabel('t')
ylabel('rads/s')

subplot(2,2,4);
plot(t,x(:,4),'*')
hold on
plot(t,x(:,8))
plot(t,x(:,4) - x(:,8),'r')
title('Pitch Angle')
legend('True','Estimated','Error');
```

```
xlabel('t')
ylabel('rads')
```



f

```
xd = [20 0 0 0]';
[t,x] = ode45(@(t,x) aircraftDynamics(t,x,xd,A,B,C,K,L),[0 5],xo);

t = t(1:10:length(t));
x = x(1:10:length(x),:);

figure(2)
clf;
subplot(2,2,1);
plot(t,x(:,1),'*')
hold on
plot(t,x(:,5))
plot(t,x(:,1) - x(:,5),'r')
plot(t,x(:,1)-xd(1)*ones(length(x(:,1)),1))
title('Airspeed')
legend('True','Estimated','Estimated Error', 'Desired Error');
xlabel('t')
ylabel('m/s')
```

```
subplot(2,2,2);
plot(t,x(:,2),'*')
hold on
plot(t,x(:,6))
plot(t,x(:,2) - x(:,6),'r')
plot(t,x(:,2)-xd(2)*ones(length(x(:,1)),1))
title('Angle of Attack')
legend('True','Estimated','Estimated Error', 'Desired Error');
xlabel('t')
ylabel('rads')

subplot(2,2,3);
plot(t,x(:,3),'*')
hold on
plot(t,x(:,7))
plot(t,x(:,3) - x(:,7),'r')
plot(t,x(:,3)-xd(3)*ones(length(x(:,1)),1))
title('Pitch Rate')
legend('True','Estimated','Estimated Error', 'Desired Error');
xlabel('t')
ylabel('rads/s')

subplot(2,2,4);
plot(t,x(:,4),'*')
hold on
plot(t,x(:,8))
plot(t,x(:,4) - x(:,8),'r')
plot(t,x(:,4)-xd(4)*ones(length(x(:,1)),1))
title('Pitch Angle')
legend('True','Estimated','Estimated Error', 'Desired Error');
xlabel('t')
ylabel('rads')

error_final = x(end,1:4)' - xd

function dxdt = aircraftDynamics(t,x,xd,A,B,C,K,L)

    z = x(1:4);          % System states
    z_hat = x(5:8);     % Estimated system states

    u = -K*(z_hat-xd);            % Input
    z_dot = A*(z) + B*u;   % System dynamics

    z_hat_dot = A*(z_hat) + B*u - L*C*(z_hat - z);

    dxdt = [z_dot;z_hat_dot];

end


estimation_error =

   1.0e-08 *
```

```
   -0.4459   -0.6511    0.6231    0.0460
```

Published with MATLAB® R2018a