4 Optimal State Estimation

Consider the LTI system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_w w(t), \\ y(t) &= C_y x(t) + D_{yw} w(t), \\ z(t) &= C_z x(t) \end{aligned}$$

Problem:
Compute $(\hat{A}, F)$ such that the output of the *state estimator*

$$\begin{aligned} \dot{\hat{x}}(t) &= \hat{A}\,\hat{x}(t) - Fy(t), \\ \hat{z}(t) &= C_z \hat{x}(t) \end{aligned}$$

stabilizes the state estimation error and minimizes the cost function

$$J := \lim_{t\to\infty} E\left[(z(t) - \hat{z}(t))^T (z(t) - \hat{z}(t))\right].$$

Assumptions:

a) (A, C_y) detectable;

b) $w(t)$ is a Gaussian zero mean white noise with variance $W \succ 0$.

Solution:
Define

$$e(t) := x(t) - \hat{x}(t)$$

so that

$$z(t) - \hat{z}(t) = C_z e(t).$$

Now write the dynamics of the estimation error

$$\begin{aligned}\dot{e}(t) &= \dot{x}(t) - \dot{\hat{x}}(t), \\ &= Ax(t) - \hat{A}\hat{x}(t) + Fy(t) + B_w w(t), \\ &= (A - \hat{A} + FC_y)x(t) + \hat{A}e(t) + (B_w + FD_{yw})w(t)\end{aligned}$$

First observation: if A is unstable $x(t)$ is unbounded and so is $e(t)$, with one exception. If

$$\hat{A} = A + FC_y \qquad \Longrightarrow \qquad A - \hat{A} + FC_y = 0$$

and

$$\dot{e}(t) = (A + FC_y)e(t) + (B_w + FD_{yw})w(t).$$

That is, the dynamics of $e(t)$ and $x(t)$ are *decoupled*!
(note that $z(t) - \hat{z}(t) = C_z e(t)$ is also decoupled from $x(t)$).
This choice of $\hat{A}$ produces the *state observer*

$$\begin{aligned}\dot{\hat{x}} &= A\hat{x} + F(\hat{y} - y), \\ \hat{y} &= C_y\hat{x}, \\ \hat{z} &= C_z\hat{x}.\end{aligned}$$

in which the only unknown is the *observer gain* F. The problem has been reduced to "find the gain F".

Simplified problem:
Compute the stabilizing state estimation gain F so as to minimize

$$J := \lim_{t\to\infty} E\left[e(t)^T C_z^T C_z e(t)\right]$$

where $e(t)$ is the state of the LTI system

$$\dot{e}(t) = (A + FC_y)e(t) + (B_w + FD_{yw})w(t).$$

Assumptions:

a) (A, C_y) is detectable;

b) $w(t)$ is a Gaussian zero mean white noise with variance $W \succ 0$.

Solution:
As before

$$J = \mathrm{trace}\left[X(B_w + FD_{yw})W(B_w + FD_{yw})^T\right]$$

where X is the Gramian

$$(A + FC_y)^T X + X(A + FC_y) + C_z^T C_z = 0.$$

However, in this form, F appears simultaneously at the cost and at the Lyapunov equation, and we can not use completion of squares to determine F.
Use duality!

4.1 Duality

Assume $W \succeq 0$. Consider the cost function

$$J = \operatorname{trace}\left(XBWB^T\right),$$

where X is the solution to the Lyapunov equation

$$A^T X + XA + C^T C = 0.$$

An alternative (dual) expression for J can be obtained from

$$\begin{aligned} J &= \operatorname{trace}\left(XBWB^T\right), \\ &= \operatorname{trace}\left(\left[\int_0^\infty e^{A^T t} C^T C e^{At}\, dt\right] BWB^T\right), \\ &= \int_0^\infty \operatorname{trace}\left(Ce^{At}BWB^T e^{A^T t} C^T\right)\, dt, \\ &= \operatorname{trace}\left(C\left[\int_0^\infty e^{At}BWB^T e^{A^T t}\, dt\right] C^T\right). \end{aligned}$$

That is

$$J = \operatorname{trace}\left(CYC^T\right),$$

where Y is the solution to the Lyapunov equation

$$AY + YA^T + BWB^T = 0.$$

Recall that Y is also the state covariance matrix!

Back to the determination of F, using the dual formulation

$$J = \operatorname{trace}\left(C_z Y C_z^T\right)$$

where

$$(A + FC_y)Y + Y(A + FC_y)^T + (B_w + FD_{yw})W(B_w + FD_{yw})^T = 0.$$

The gain F now appears only in the Lyapunov equation!
Assuming that

$$B_w W D_{yw}^T = 0, \qquad D_{yw} W D_{yw}^T \succ 0,$$

(for simplicity only, without loss of generality!) the above equation becomes

$$(A + FC_y)Y + Y(A + FC_y)^T + B_w W B_w^T + F D_{yw} W D_{yw}^T F^T = 0.$$

We can now complete the squares to obtain the ARE

$$AY + YA^T - YC_y^T (D_{yw} W D_{yw}^T)^{-1} C_y Y + B_w W B_w^T = 0$$

and the associated optimal gain

$$F = -YC_y^T (D_{yw} W D_{yw}^T)^{-1}.$$

4.2 Summary on Estimation

Problem:
Given the LTI system

$$\begin{aligned}\dot{x}(t) &= Ax(t) + B_w w(t), \\ y(t) &= C_y x(t) + D_{yw} w(t), \\ z(t) &= C_z x(t)\end{aligned}$$

compute the estimation gain F that stabilizes the state estimation erros and that the output of the *state estimator*

$$\begin{aligned}\dot{\hat{x}}(t) &= A\,\hat{x}(t) + F(\hat{y}(t) - y(t)), \\ \hat{y}(t) &= C_y \hat{x}(t), \\ \hat{z}(t) &= C_z \hat{x}(t)\end{aligned}$$

minimizes the cost function

$$J := \lim_{t \to \infty} E\left[(z(t) - \hat{z}(t))^T (z(t) - \hat{z}(t))\right].$$

Assumptions:

a) (A, C_y) detectable;

b) $w(t)$ is a Gaussian zero mean white noise with variance $W \succ 0$;

c) $B_w W D_{yw}^T = 0$, $\quad D_{yw} W D_{yw}^T \succ 0$.

Solution:
Find the stabilizing solution to the ARE

$$AY + YA^T - YC_y^T (D_{yw} W D_{yw}^T)^{-1} C_y Y + B_w W B_w^T = 0.$$

The optimal gain is

$$F = -YC_y^T (D_{yw} W D_{yw}^T)^{-1}.$$

Things we can infer from the form of the solution:

1) $A + FC_y$ is stable even if A is not!
What happens if A is unstable and $\hat{A} = A + FC_y + \delta$?

2) The assumption $D_{yw} W D_{yw}^T \succ 0$ reads "no combination of the measurement is error free".

3) If $D_{yw} W D_{yw}^T \succeq 0$, then part of the state could be reconstructed exactly from the measurements.
While this might sounds easier, we do not know how to solve this problem :).

4) State estimation is "equivalent" to the state feedback problem

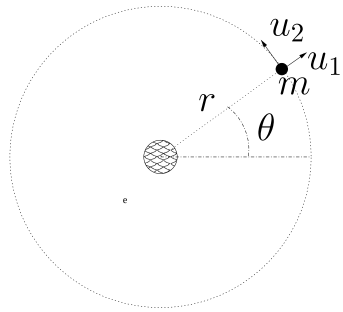
$$\min J = \lim_{t \to \infty} E\left[x^T B_w W B_w^T x + u^T D_{yw} W D_{yw}^T u\right],$$

for the LTI system

$$\begin{aligned} \dot{x} &= A^T x + C_y^T u + C_z^T w, \\ u &= F^T x \end{aligned}$$

5) The optimal gain F does not depend on C_z! The optimal cost does!

4.3 Example: estimating the state of a satellite in circular orbit



Satellite of mass m with thrust in the radial direction u_1 and in the tangential direction u_2. Continuing...

$$m(\ddot{r} - r\dot{\theta}^2) = u_1 - \frac{km}{r^2} + w_1,$$
$$m(2\dot{r}\dot{\theta} + r\ddot{\theta}) = u_2 + w_2,$$

where w_1 and w_2 are independent white noise disturbances with variances δ_1 and δ_2. For the purpose of estimation we set $u_1 = u_2 = 0$ for the moment. As before, putting in state space and linearizing

$$\dot{x} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 3\,\bar{\omega}^2 & 0 & 0 & 2\,\bar{r}\bar{\omega} \\ 0 & 0 & -2\,\bar{\omega}/\bar{r} & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1/m & 0 \\ 0 & 1/(m\bar{r}) \end{bmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

Consider that you have a noisy measurement of θ (x_2)

$$y = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} + v$$

where $E\{vv^T\} = \delta_3$.
Problem: Given

$$m = 100 \text{ kg}, \qquad \bar{r} = R + 300 \text{ km}, \qquad \bar{k} = GM$$

where $G \approx 6.673 \times 10^{-11}$ N m^2/kg^2 is the universal gravitational constant, and $M \approx 5.98 \times 10^{24}$ kg and $R \approx 6.37 \times 10^3$ km are the mass and radius of the earth. If the variances $\delta_1 = \delta_2 = \delta_3 = 0.1N$, estimate the state of the satelite.

```
% MAE 280 B - Linear Control Design
% Mauricio de Oliveira
%
% State estimation - Part I
%
m = 100;               % 100 kg
r = 300E3;             % 300 km
R = 6.37E6;            % 6.37 10^3 km
G = 6.673E-11;         % 6.673 N m^2/kg^2
M = 5.98E24;           % 5.98 10^24 kg
k = G * M;             % gravitational force constant
w = sqrt(k/((R+r)^3)); % angular velocity (rad/s)
v = w * (R + r);       % "ground" velocity (m/s)

% linearized system matrices
A = [0 0 1 0; 0 0 0 1; 3*w^2 0 0 2*(r+R)*w; 0 0 -2*w/(r+R) 0];
Bu = [0 0; 0 0; 1/m 0; 0 1/(m*r)];
Bw = [0 0; 0 0; 1/m 0; 0 1/(m*r)];

% noise variances
W = 0.1 * eye(2)
W =
   1.0000e-01            0
            0   1.0000e-01

% any measurement that does not include x2 (theta) is not observable!
% for instance
Cy = [1 0 0 0; 0 0 1 0; 0 0 0 1];

% system
sys = ss(A, Bw, Cy, 0)

a =
               x1           x2           x3           x4
   x1           0            0            1            0
   x2           0            0            0            1
   x3   4.034e-06            0            0    1.547e+04
   x4           0            0   -3.477e-10            0

b =
              u1          u2
   x1          0           0
   x2          0           0
   x3       0.01           0
   x4          0   3.333e-08

c =
       x1  x2  x3  x4
   y1   1   0   0   0
   y2   0   0   1   0
   y3   0   0   0   1

d =
```

```
       u1  u2
   y1   0   0
   y2   0   0
   y3   0   0

Continuous-time model.
% take the comment out of next line to see error messages!
%est = kalman(sys, W, eye(size(Cy,1)));
% back to the problem
% measuring x2 (theta)
Cy = [0 1 0 0];

% augment noise matrices
Bwa = [Bw zeros(4,1)]
Bwa =
            0            0            0
            0            0            0
   1.0000e-02            0            0
            0   3.3333e-08            0
Dywa = [zeros(1,2) 1/r]
Dywa =
            0            0   3.3333e-06

% scale
T = diag([1 r 1 r])
T =
           1            0            0            0
           0       300000            0            0
           0            0            1            0
           0            0            0       300000

% similarity transformation
At = T * A / T
At =
            0            0   1.0000e+00            0
            0            0            0   1.0000e+00
   4.0343e-06            0            0   5.1565e-02
            0            0  -1.0432e-04            0
But = T * Bu
But =
            0            0
            0            0
   1.0000e-02            0
            0   1.0000e-02
Bwt = T * Bwa
Bwt =
            0            0            0
            0            0            0
   1.0000e-02            0            0
            0   1.0000e-02            0
Cyt = Cy / T
Cyt =
            0   3.3333e-06            0            0
Dywt = Dywa
```

```
Dywt =
            0            0   3.3333e-06

Ww = W
Ww =
   1.0000e-01            0
            0   1.0000e-01
Wv = 0.1
Wv =
   1.0000e-01

Wt = [Ww zeros(2, 1); zeros(1, 2) Wv]
Wt =
   1.0000e-01            0            0
            0   1.0000e-01            0
            0            0   1.0000e-01

% compute using dual state feedback
[F,X,S] = lqr(At', Cyt', Bwt * Wt * Bwt', Dywt * Wt * Dywt');
F = - F'
F =
   5.9160e+07
  -4.3621e+04
   1.1664e+05
  -3.1713e+03

% compute using matlab's kalman
sys = ss(At, Bwt(:,1:2), Cyt, 0);
est = kalman(sys, Ww, Wv/r^2);
F = -est.b
F =
   5.9160e+07
  -4.3621e+04
   1.1664e+05
  -3.1713e+03

% error dynamics
eig(At + F * Cyt)
ans =
  -1.9571e-03
  -2.0614e-03
  -7.0692e-02 + 7.0730e-02i
  -7.0692e-02 - 7.0730e-02i

% simulate estimator
sys = ss(T * A / T, T * Bwa, Cy / T, Dywa);
filt = est * sys;

Tmax = 20000;
T = 0 : 0.1 : Tmax;
w = [randn(length(T),2)*sqrtm(Ww) randn(length(T),1)*sqrtm(Wv)];

% estimate stationary position
figure(1)
```

```
x0 = [0; 0; 0; 0]
x0 =
     0
     0
     0
     0
xhat0 = [0; 0; -6; 0]
xhat0 =
     0
     0
    -6
     0
[y,t,x] = lsim(sys, w, T, x0);
[yf,tf,xf] = lsim(filt, w, T, [xhat0; x0]);
xhat = xf(:,1:4);

figure(1)
plot(t, x(:,[1 2]), 'g', t, xhat(:,[1 2]), 'b'),
title('system and estimator trajectory (x_1 and x_2)')
xlim([0, Tmax])
grid

figure(2)
plot(t, x(:,[3 4]), 'g', t, xhat(:,[3 4]), 'b'),
title('system and estimator trajectory (x_3 and x_4)')
xlim([0, Tmax])
grid

figure(3)
plot([0 T(end)], [0 0], 'g', t, x(:,[1 2])-xhat(:,[1 2]), 'b'),
title('estimation error (x_1 and x_2)')
xlim([0, Tmax])
grid

figure(4)
plot([0 T(end)], [0 0], 'g', t, x(:,[3 4])-xhat(:,[3 4]), 'b'),
title('estimation error (x_3 and x_4)')
xlim([0, Tmax])
grid

diary off
```

4.4 Example: estimating position

Dynamics of particle of mass $m = 100$ Kg in a line:

$$m\ddot{x} = w_1,$$

where w_1 is a zero-mean white noise disturbances with variance $\delta_1 = (100 \text{ N})^2$.
As a measurement we have a noisy measurement of position, say through GPS

$$y_1 = x + w_2$$

where w_2 is a white noise disturbances with variance $\delta_2 = (10 \text{ m})^2$.
In state space

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 1/m & 0 \end{bmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}, \\ y &= \begin{bmatrix} 1 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \end{aligned}$$

Problem #1: Estimate the state x.
Problem #2: Can an accelerometer improve the estimate?

```
% MAE 280 B - Linear Control Design
% Mauricio de Oliveira
%
% State estimation - Part II
%
m = 100;               % 100 kg

% linearized system matrices
A = [0 1; 0 0]
A =
     0     1
     0     0
Bw = [0; 1/m]
Bw =
            0
   1.0000e-02

% position
Cy = [1 0]
Cy =
     1     0
Dyw = 0
Dyw =
     0

% noise variances
Ww = 100^2
Ww =
       10000

% augmented system
Bwa = [Bw zeros(2,1)]
Bwa =
            0             0
   1.0000e-02             0
Dywa = [zeros(1,1) 1]
Dywa =
     0     1

% measurement noise variances
Wv = 10^2
Wv =
   100

Wa = [Ww zeros(1, 1); zeros(1, 1) Wv]
Wa =
       10000           0
           0         100

% compute using dual state feedback
[F,X,S] = lqr(A', Cy', Bwa * Wa * Bwa', Dywa * Wa * Dywa');
F = - F'
F =
```

```
  -4.4721e-01
  -1.0000e-01

% compute using matlab’s kalman
sys = ss(A, Bw, Cy, Dyw);
est = kalman(sys, Ww, Wv);
F = -est.b
F =
  -4.4721e-01
  -1.0000e-01

% error dynamics
eig(A + F * Cy)
ans =
  -2.2361e-01 + 2.2361e-01i
  -2.2361e-01 - 2.2361e-01i

% try to simulate estimator
sys1 = ss(A, Bwa, Cy, Dywa);
filt1 = est * sys1;

Tmax = 240;
T = 0 : 0.1 : Tmax;
w = [randn(length(T),1)*sqrtm(Ww) randn(length(T),1)*sqrtm(Wv)];

% estimate stationary position (no noise)
figure(1)
x0 = [10; 0]
x0 =
    10
     0
xhat0 = [0; 0]
xhat0 =
     0
     0
[y,t,x] = initial(sys1, x0, T);
[yf,tf,xf] = initial(filt1, [xhat0; x0], T);
xhat = xf(:,1:2);

figure(1)
plot(t, x, t, xhat),
title(’system and estimator trajectory’)
xlim([0, Tmax])
grid

figure(2)
plot(t, x-xhat(:,1:2)),
title(’estimation error’)
xlim([0, Tmax])
grid
pause

% estimate stationary position (with measurement noise)
wm = w;
```

```
wm(:,1) = 0;
x0 = [10; 0]
x0 =
    10
     0
xhat0 = [5; 0]
xhat0 =
     5
     0
[y,t,x] = lsim(sys1, wm, T, x0);
[yf,tf,xf] = lsim(filt1, wm, T, [xhat0; x0]);
xhat = xf(:,1:2);

figure(1)
plot(t, x, 'g', t, xhat, 'b'),
title('system and estimator trajectory')
xlim([0, Tmax])
grid

figure(2)
plot([0 T(end)], [0 0], 'g', t, x-xhat(:,1:2), 'b'),
title('estimation error')
xlim([0, Tmax])
grid
pause

% estimate moving position (with measurement and process noise)
x0 = [0; 0]
x0 =
     0
     0
xhat0 = zeros(2,1)
xhat0 =
     0
     0
[y,t,x] = lsim(sys1, w, T, x0);
[yf,tf,xf] = lsim(filt1, w, T, [xhat0; x0]);
xhat = xf(:,1:2);

figure(1)
plot(t, x, 'g', t, xhat, 'b'),
title('system and estimator trajectory')
xlim([0, Tmax])
grid

figure(2)
plot([0 T(end)], [0 0], 'g', t, x-xhat(:,1:2), 'b'),
title('estimation error')
xlim([0, Tmax])
grid

diary off
```

```
% MAE 280 B - Linear Control Design
% Mauricio de Oliveira
%
% State estimation - Part III
%
m = 100;               % 100 kg

% linearized system matrices
A = [0 1; 0 0]
A =
     0     1
     0     0
Bw = [0; 1/m]
Bw =
            0
   1.0000e-02

% position
Cy1 = [1 0]
Cy1 =
     1     0
Dyw1 = 0
Dyw1 =
     0

% accelerometer
P = [0 1];
Cy2 = P*A
Cy2 =
     0     0
Dyw2 = P*Bw
Dyw2 =
   1.0000e-02

Cy = [Cy1; Cy2];
Dyw = [Dyw1; Dyw2];

% noise variances
Ww = 100^2
Ww =
       10000
Wv = diag([10^2 100]);

% augmented system matrices
Bwa = [Bw zeros(2,2)]
Bwa =
            0            0            0
   1.0000e-02            0            0
Dywa = [zeros(2,1) eye(2)]
Dywa =
     0     1     0
     0     0     1
Wa = [Ww zeros(1, 2); zeros(2, 1) Wv]
```

```
Wa =
       10000           0           0
           0         100           0
           0           0         100

% compute using matlab's kalman
sys = ss(A, Bw, Cy, Dyw);
est = kalman(sys, Ww, Wv);
F = -est.b
F =
  -4.4610e-01            0
  -9.9504e-02  -9.9010e-03

% error dynamics
eig(A + F * Cy)
ans =
  -2.2305e-01 + 2.2305e-01i
  -2.2305e-01 - 2.2305e-01i

% try to simulate estimator
sys2 = ss(A, Bwa, Cy, Dywa);
filt2 = est * sys2;

Tmax = 240;
T = 0 : 0.1 : Tmax;
w = [randn(length(T),1)*sqrtm(Ww) randn(length(T),2)*sqrtm(Wv)];

% estimate stationary position (with measurement noise)
wm = w;
wm(:,1) = 0;
x0 = [10; 0]
x0 =
    10
     0
xhat0 = [5; 0]
xhat0 =
     5
     0
[y,t,x] = lsim(sys2, wm, T, x0);
if exist('filt1')
  [yf1,tf1,xf1] = lsim(filt1, wm(:,1:2), T, [xhat0; x0]);
  xhat1 = xf1(:,1:2);
end
[yf2,tf2,xf2] = lsim(filt2, wm, T, [xhat0; x0]);
xhat2 = xf2(:,1:2);

figure(1)
if exist('filt1')
  plot(t, x, 'g', t, xhat1, 'b', t, xhat2, 'r'),
end
title('system and estimator trajectory')
xlim([0, Tmax])
grid
```

```
figure(2)
if exist('filt1')
  plot([0 T(end)], [0 0], 'g', t, x-xhat1(:,1:2), 'b', t, x-xhat2(:,1:2), 'r'),
end
title('estimation error')
xlim([0, Tmax])
grid
pause

% estimate moving position (with measurement and process noise)
x0 = [10; 0]
x0 =
    10
     0
xhat0 = [5; 0]
xhat0 =
     5
     0
[y,t,x] = lsim(sys2, w, T, x0);
if exist('filt1')
  [yf1,tf1,xf1] = lsim(filt1, w(:,1:2), T, [xhat0; x0]);
  xhat1 = xf1(:,1:2);
end
[yf2,tf2,xf2] = lsim(filt2, w, T, [xhat0; x0]);
xhat2 = xf2(:,1:2);

figure(1)
if exist('filt1')
  plot(t, x, 'g', t, xhat1, 'b', t, xhat2, 'r'),
end
title('system and estimator trajectory')
xlim([0, Tmax])
grid

figure(2)
if exist('filt1')
  plot([0 T(end)], [0 0], 'g', t, x-xhat1(:,1:2), 'b', t, x-xhat2(:,1:2), 'r'),
end
title('estimation error')
xlim([0, Tmax])
grid

diary off
```