Solutions to Exercise 1. Control in Matlab

1.1

```
>> A = [0 1; 1 0];
>> B = [1 0]';
>> C = [0 1];
>> D = 0;
>> eig(A)

ans =

  -1
   1
```

1.2

```
>> sys = ss(A,B,C,D);
>> tf(sys)

Transfer function:
   1
-------
s^2 - 1
```

1.3

```
>> [z,p,k] = zpkdata(sys,'v')

z =

   Empty matrix: 0-by-1

p =

    -1
     1

k =

     1
```

1.4

```
>> s = tf('s');
>> P = 1/(s^2+0.6*s+1);
>> P.InputDelay = 1.5

Transfer function:
                     1
exp(-1.5*s) * ---------------
              s^2 + 0.6 s + 1

>> bode(P)
>> grid
>> nyquist(P)
>> step(P)
```

As seen in the step response the *open-loop* system is stable. The Bode and Nyquist plots show that the *closed-loop* system will be unstable.

1.5

```
>> Wc = ctrb(A,B);
>> rank(Wc)
```

```
   ans =

        2
```

Since the controllability matrix has full rank, the system is controllable.

```
>> p=[1 1.4 1];
>> L=place(A,B,roots(p))

   L =

       1.4000    2.0000
```

1.6

```
>> C = 0.5*(1+4*s);
>> margin(C*P)
```

1.7

```
>> CLSYS = feedback(C*P,1)

   Transfer function:
       2 s + 0.5
   -----------------
   s^2 + 2.6 s + 1.5

>> CLSYS = minreal(C*P/(1+C*P))

   Transfer function:
       2 s + 0.5
   -----------------
   s^2 + 2.6 s + 1.5
```

1.8

```
>> step(CLSYS)
>> dcgain(CLSYS)

   ans =
       0.3333
```

1.9

```
>> A=[-1 1 0 -1/2 0; 4 -1 0 -25 8; 0 1 0 0 0; 0 0 0 -20 0; 0 0 0 0 -20];
>> B=[0 0; 3/2 1/2; 0 0; 20 0; 0 20];
>> C=[0 1 0 0 0; 0 0 1 0 0];
>> jas=ss(A,B,C,[0 0; 0 0]);
>> pole(jas)
ans =
         0
    1.0000
   -3.0000
  -20.0000
  -20.0000
>> rank(ctrb(jas))
ans =
     5
>> rank(obsv(jas))
ans =
     4
```

We see that the system is unstable. This means that without some type of control, the plane will crash. Fortunately, the system is controllable, which means that it is possible to stabilise the aircraft with the given actuators.

However, since we do not have observability, we need to have some other combination of sensors if we want to use state feedback.
To get the transfer function, we use

```
  >> G=tf(jas)

Transfer function from input 1 to output...
      1.5 s^2 - 468.5 s - 510
 #1:  ------------------------
      s^3 + 22 s^2 + 37 s - 60

        1.5 s^2 - 468.5 s - 510
 #2:  ----------------------------
      s^4 + 22 s^3 + 37 s^2 - 60 s

Transfer function from input 2 to output...
      0.5 s^2 + 170.5 s + 170
 #1:  ------------------------
      s^3 + 22 s^2 + 37 s - 60

        0.5 s^2 + 170.5 s + 170
 #2:  ----------------------------
      s^4 + 22 s^3 + 37 s^2 - 60 s

>> G(1,2) % To output 1 from input 2 (note the order of indexing)

Transfer function:
0.5 s^2 + 170.5 s + 170
------------------------
s^3 + 22 s^2 + 37 s - 60
```

1.10

```
>> G1 = 1/(s+1)^3

   Transfer function:
          1
 ---------------------
 s^3 + 3 s^2 + 3 s + 1

>> pole(G1)

  ans =

      -1.0000
      -1.0000 + 0.0000i
      -1.0000 - 0.0000i

>> G2 = zpk(1/(s+1)^3)

   Zero/pole/gain:
         1
      -------
      (s+1)^3

>> pole(G2)
```

```
  ans =

      -1.0000
      -1.0000 + 0.0000i
      -1.0000 - 0.0000i

>> G3 = 1/(s^3+2.99*s^2+3*s+1);
>> pole(G3)

  ans =

      -1.0888 + 0.2131i
      -1.0888 - 0.2131i
      -0.8124

>> G4 = 1/(s+0.99)^3;
>> pole(G4)

  ans =

      -0.9900 + 0.0000i
      -0.9900 - 0.0000i
      -0.9900
```

We see that the same small modification in a parameter, causes larger changes in the dynamics when the system is represented as G_3. The transfer function format of G_4 (three poles in the same spot as for G_2), which can be kept with the `zpk` command, is in general better numerically compared to the format in which G_3 is represented (the same form as the command `tf` gives).

1.11

```
>> Wo = obsv(A,C)

   ans =

     3    4
    -3   -4

>> rank(Wo)

   ans =

        1

>> rank(ctrb(A,B))

   ans =

        1
```

Since neither the observability matrix nor the controllability matrix has full rank, the system is neither observable nor controllable. It can be seen directly from the state equations, where we have two states that are completely discoupled from each other and have the same eigenvalue. This means that evolution of the states will look exactly the same for any control signal $u(t)$. (More specifically: $2(x_1(t) - x_1(0)) = x_2(t) - x_2(0)$). Therefore we will never be able to control these states arbitrarily. We will only be able to control them along some controllable subspace. The same goes for the observability.

1.12 The transfer function for the mass-spring system will be

```
>> zpk(ss(A,B,C,D))

   Zero/pole/gain:
       2
  -----------
  (s^2  + 20)
```

The transfer function of a PID controller is

$$R = \frac{K(sT_i + 1 + s^2 T_d T_i)}{sT_i}$$

and the closed loop transfer function is

$$G_{cl}(s) = \frac{R(s)P(s)}{1 + R(s)P(s)} = \frac{2K(s^2 T_d + s + 1/T_i)}{s^3 + s^2 2KT_d + s(20 + 2K) + 2K/T_i}$$

The closed loop characteristic equation is then

$$s^3 + s^2(2KT_d) + s(20 + 2K) + 2K/T_i = 0$$

Identify the coefficients and solve for K, T_i and T_d as functions of ω and ζ:

$$K = 0.5(\omega^2 + 2\zeta\omega^2 - 20)$$
$$T_i = \frac{2K}{\omega^3}$$
$$T_d = \frac{2\zeta\omega + \omega}{2K}$$

The closed loop system is then

```
>> G_cl = feedback(R*P,1);
>> step(G_cl)
```

The specification is met for many different choices of ω and ζ. One choice can be $\omega = 6$ and $\zeta = 0.7$.

1.13

```
>> s = tf('s');
>> P = (3-s)/((s+1)*(s+2));
>> [A,B,C,D] = ssdata(P);
>> rank(ctrb(A,B))

   ans =

       2

>>  p = [1 5.6 16];
>>  L = place(A,B,roots(p));
```

The system is controllable, since the controllability matrix has full rank. With the control law $u(t) = -Lx + r$, the closed loop system get the following appearance

```
>> A_cl = A-B*L;
>> B_cl = B;
>> C_cl = C;
>> D_cl = 0;
>> G_cl = ss(A_cl,B_cl,C_cl,D_cl);
>> step(G_cl)
>> dcgain(G_cl)

ans =

    0.1875
```

The system is non-minimum phase, which we can see directly since the process has a zero in the right half plane.

Solutions to Exercise 2. System Representations and Stability

2.1 A state-space representation of the system is given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -2 & 1 \\ 0 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

2.2 Laplace transformation of the differential equation gives

$$Y(s) = \frac{(b_{11}s + b_{12})}{(s^2 + a_1 s + a_2)} U_1(s) + \frac{(b_{21}s + b_{22})}{(s^2 + a_1 s + a_2)} U_2(s)$$

The transfer matrix becomes

$$\begin{pmatrix} \frac{b_{11}s+b_{12}}{s^2+a_1s+a_2} & \frac{b_{21}s+b_{22}}{s^2+a_1s+a_2} \end{pmatrix}$$

2.3

a. The equation can be written as

$$y = g * u \tag{2.1}$$

where $g(t) = te^{-2t}$, $t \geq 0$. Taking the Laplace transform of (2.1) gives

$$Y(s) = \frac{1}{(s+2)^2}(R(s) - Y(s))$$
$$Y(s) = \frac{1}{s^2 + 4s + 5} R(s)$$

b. The transfer function has poles in

$$s_1 = -2 + i$$
$$s_2 = -2 - i$$

Since all poles have negative part the system is input-output stable.

Another way of checking stability of a second order system with characteristic equation $s^2 + a_1 s + a_2$ is that $a_1, a_2 > 0$.

c. The L_2-gain is given by the supremum of the transfer function gain, so we want to find the frequency where the gain peaks.

The easiest way to do this is to plot the Bode diagram for $G_c(s)$ and find the peak in the gain curve.

Alternatively, one can find the frequency that maximizes the gain by the following reasoning: Since it is a second order system, it can be written as

$$G_c(s) = \frac{K}{s^2 + 2\zeta\omega s + \omega^2}$$

In our case $\zeta = 2/\sqrt{5} \approx 0.9$. This means that the system is well damped and that it does not have a resonance peak in the gain curve. Since the gain is decreasing with frequency, the maximum gain can thus be found at $\omega = 0$.

$$|G_c(i \cdot 0)| = \frac{1}{5}$$

2.4

a.

$$\begin{aligned} S(s) = \frac{1}{1+CP} &= \frac{s^3 + 2s^2 + s}{s^3 + 2s^2 + 2.4s + 1.4} = \\ &= \frac{(s+1)(s^2+s)}{(s+1)(s^2+s+1.4)} = \frac{s^2+s}{s^2+s+1.4} \end{aligned}$$

Remark: Notice that we have a 3rd order system (with a 1st order controller and 2nd order plant), but the transfer functions $S(s)$ is only of 2nd order! Looking at the block-diagram of the system one can clearly see the pole–zero cancellation of the term $(s+1)$ for $P \cdot C$. These kind of pole-zero cancellations imply loss of either observability or loss of controllability. Which is it in this case?

$$T(s) = \frac{CP}{1+CP} = \frac{1.4(s+1)}{s^3 + 2s^2 + 2.4s + 1.4} = \frac{1.4}{s^2+s+1.4}$$

Remark: Also for T there has been a pole-zero cancellation of $(s+1)$, but a corresponding cancellation **does not** appear in for instance $G_{d->y} = \frac{P}{1+PC}$. The transfer function from n to y is $T(s)$ (the minus sign can be ignored since we could just as well say that the unknown noise is given by $-n$). This means that the reference and the measurement noise have the same effect on the output.

b. We know that $S(s)$ is the transfer function from load disturbance to output. Since the control system should remove the effects of load disturbances, which often are of low frequency character, it would seem reasonable if the curve representing $S(s)$ decreases as we move to the left. This corresponds to the upper curve.

We could also look at the function $S(s)$ that we just determined. We see that

$$\lim_{s \to 0} S(s) = 0$$

Comparing with the upper curve, which has a gain that goes to zero for low frequencies, we conclude that this represents the sensitivity function.

c. In order to have good tracking of the reference value, we want the gain from reference to output to be close to one. Looking at the gain curve of the *complimentary transfer function* T we see that for $\omega < 1$, we have $T \approx 1$, resulting in good tracking of the reference value.

Additional comments: At the same time, we want to be insensitive to process noise and measurement noise, i.e. we want the gain to be as small as possible for these two signals.

The transfer function from process noise to output is S, while T is the transfer function of both reference values and measurement noise to the output. S and T can not be small at the same frequencies, due to the fact that

$$S(s) + T(s) = \frac{1}{1 + C(s)P(s)} + \frac{C(s)P(s)}{1 + C(s)P(s)} = 1$$

Thus, we need to think about the frequency character of these signals, and compare with the shapes of the transfer functions: Process noise and reference signals are often of low frequency, so we want to have $S \approx 0$ and $T \approx 1$ at low frequencies. Measurement noise is most often of high frequency, so we want to have $T \approx 0$ at high frequencies.

d. At $\omega > 1$ T is small, resulting in good attenuation of measurement noise. (Do you see how the "speed" of control relates to the impact of measurement noise?)

2.5

a. The sensitivity function is given by $S = \frac{1}{1+PC}$, so S is small at frequencies where PC is large. The stationary gain of P is finite. C_2 and C_3 both have integral action and infinite stationary gain. Thus, for these controllers, S will go to zero as $\omega \to 0$. C_1, being a pure P-controller, has a finite stationary gain. S will then also have a finite stationary gain.

C_2 and C_3 are PI-controllers, but C_3 has a delay which will introduce extra phase loss. This decreases the phase margin and therefore introduces a higher sensitivity peak. Thus, we have: $C_1 \to A$, $C_2 \to C$, and $C_3 \to B$.

b. Since controller C_1 does not have integral action, we will get a stationary error in the response to a constant load disturbance, d. The response using the delayed controller C_3 will be less damped than the response using the PI-controller because of the smaller phase margin, C_2. This gives: $C_1 \to II$, $C_2 \to I$, and $C_3 \to III$.

2.6

a.

$$y = \alpha h_2, \qquad f = \beta(h_1 - h_2)$$
$$\dot{h}_1 = \frac{1}{A_1}(u_1 - f), \qquad \dot{h}_2 = \frac{1}{A_2}(u_2 + f - y)$$

$$\dot{h} = \begin{pmatrix} -\frac{1}{A_1}\beta & \frac{1}{A_1}\beta \\ \frac{1}{A_2}\beta & -\frac{1}{A_2}(\beta + \alpha) \end{pmatrix} h + \begin{pmatrix} \frac{1}{A_1} & 0 \\ 0 & \frac{1}{A_2} \end{pmatrix} u$$
$$y = \begin{pmatrix} 0 & \alpha \end{pmatrix} h$$

b.

$$G(s) = \frac{1}{s^2 + (2\beta + \alpha)s + \alpha\beta} \begin{pmatrix} \alpha\beta & \alpha(s+\beta) \end{pmatrix}$$

c. The L_2-gain can be computed in Matlab as:

```
>> s = tf('s');
>> G = 1/(s^2+3*s+1)*[1 s+1];
>> P = norm(G, inf)
```

The L_2-gain is $\sqrt{2}$.

d. *Short explanation:* The gain definition uses 2-norm to measure signals. The error of thinking is a result of forgetting this and instead assuming that the size of the input is equal to the sum of the inflows. To really see the difference, consider e.g. the input $u_1 = 1, u_2 = -1$. Then the total inflow is 0, while the 2-norm of the input is $\sqrt{1^2 + (-1)^2} = \sqrt{2} \neq 0$.

Longer explanation: The L_2-gain is the largest output signal that can be achieved (in 2-norm) by an input signal which is not larger than one (in 2-norm). More formally, if we let the vector $u = \begin{pmatrix} u_1 & u_2 \end{pmatrix}^T$, the gain is given by

$$\|G(i\omega)\|_\infty = \sup_{u \neq 0} \frac{\|G(i\omega)u\|_2}{\|u\|_2} = \sup_{\|u\|_2 \leq 1} \|G(i\omega)u\|_2 = \sqrt{2}.$$

Obviously, the amount of water going into the tank is given by $u_1 + u_2$. But the gain definition does not mention the sum of the elements of u. It does however say that $1 \geq |u| = \sqrt{u_1^2 + u_2^2}$. If, e.g., we let $u_1 = u_2 = \sqrt{2}/2$, then the maximum output is achieved, while $|u| = 1$. But still, the output is equal to the sum of the inputs.

2.7

a. The closed-loop system is given by

$$Y(s) = \frac{1}{ms^2 + 2s}\left(1 + \frac{1}{s}\right)(R(s) - Y(s))$$

We have

$$Y(s) = \frac{s+1}{ms^3 + 2s^2 + s + 1} R(s)$$

The characteristic polynomial is

$$ms^3 + 2s^2 + s + 1 = (d+1)s^3 + 2s^2 + s + 1$$

A feedback loop between $Q(s)/P(s)$ and d gives the closed-loop system

$$\frac{d \cdot Q(s)/P(s)}{1 + d \cdot Q(s)/P(s)} = \frac{d \cdot Q(s)}{P(s) + d \cdot Q(s)}$$

To get the same characteristic polynomial, we choose $P(s) = s^3 + 2s^2 + s + 1$ and $Q(s) = s^3$.

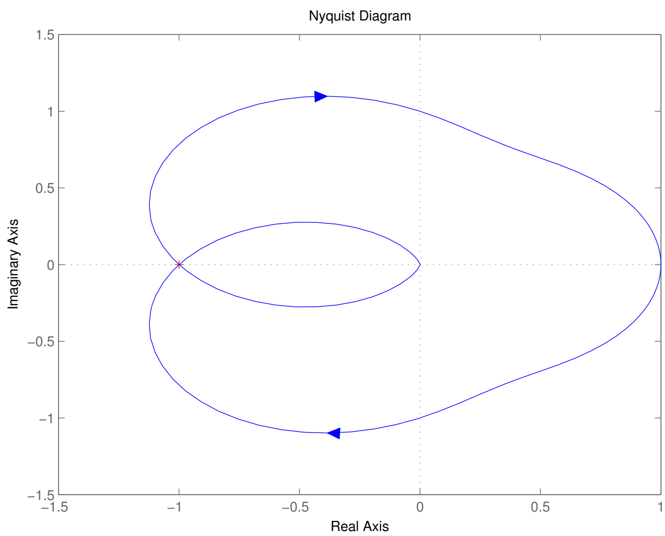


Figure 2.1 Nyquist diagram for open loop system $\frac{Q(s)}{P(s)}$ (with $d = 1 \Rightarrow m = 2$) in Problem 2.7(b).

b. Since $Q(s)/P(s)$ is stable, the closed-loop system is stable if and only if the Nyquist curve does not encircle -1.

Figure 2.1 shows the Nyquist curve for the open loop system $\frac{Q(s)}{P(s)}$ with $d = 1$ ($m = 2$). The closed-loop system is therefore stable for $d < 1 \Leftrightarrow m < 2$. If $d > 1$ the Nyquist curve would encircle -1 and the closed loop system would be unstable.

Matlab commands:

```
>> s = tf('s');
>> Q = s^3;
>> P = s^3 + 2*s^2 + s + 1;
>> d = 1;
>> nyquist(d*Q/P)
>> nyquist(0.9*Q/P)
>> nyquist(1.1*Q/P)
```

c. The open-loop transfer function (controller · process) is given by

$$G_{ol}(s) = \frac{1}{s^2 + 2s}\left(1 + \frac{1}{s}\right)e^{-s\tau}$$

The system will become unstable when $\arg G_{ol}(i\omega_c) = -180°$, i.e. when the phase margin is zero. The phase margin of the undelayed system can be found by plotting the Bode diagram (shown in Figure 2.2):

```
>> s = tf('s');
>> G = 1/(s^2+2*s)*(1+1/s);
>> margin(G)   % Plot Bode diagram and show phase and gain margins
```

From the plot we see that the cross-over frequency is $\omega_c = 0.767$ and the phase margin is $\phi_m = 16.5° = 0.288$ rad.

The delay $e^{-s\tau}$ gives a phase change of $\arg e^{-i\omega_c\tau} = -\omega_c\tau$ at the cross-over frequency. We will lose stability when

$$\omega_c\tau = \phi_m \quad \Rightarrow \quad \tau = 0.376$$

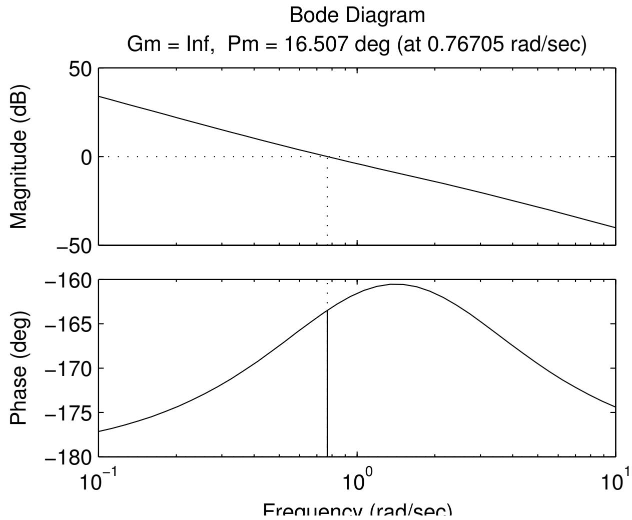


Figure 2.2 Bode diagram of undelayed system in Problem 2.7(c).

(In Matlab, you can also use the command `allmargin(G)` to find the delay margin directly.)

d. Block diagrams of the original and the rewritten closed-loop system are shown in Figure 2.3.

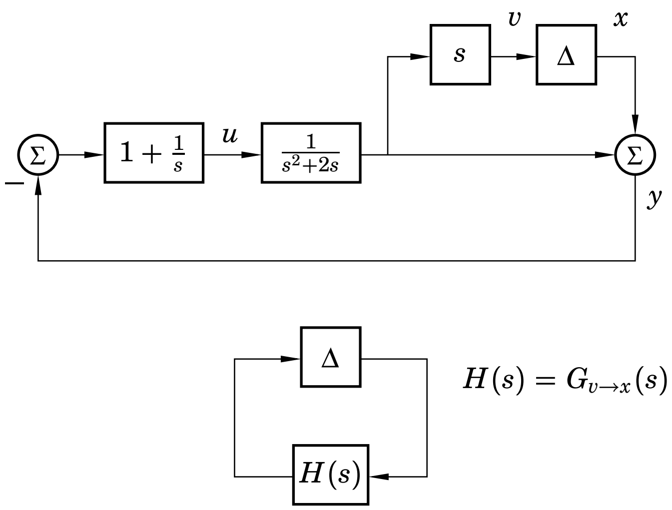


Figure 2.3 Block diagram of closed-loop system (top) and the rewritten closed-loop system (bottom) in Problem 2.7(d).

By inspection of the block diagram, we find that

$$H(s) = \frac{-\left(1+\frac{1}{s}\right)\frac{1}{s^2+2s}}{1+\left(1+\frac{1}{s}\right)\frac{1}{s^2+2s}} \cdot s = \frac{-s^2-s}{s^3+2s^2+s+1}$$

Since $H(s)$ is stable (see (a) and (b)), we can compute the gain

$$\|H\|_\infty = 2.68$$

using Matlab. The small gain theorem says that the closed-loop system is stable if

$$\|H\|_\infty \cdot \|\Delta\|_\infty < 1$$

so we must have

$$\|\Delta\|_\infty < 0.374$$

Matlab commands:

```
>> s = tf('s');
>> H = (-s^2-s)/(s^3+2*s^2+s+1);
>> norm(H, inf)
>> 1/ans
```

Solutions to Exercise 3. Disturbance Models and Robustness

3.1 Let

$$G(s) = \frac{1}{s+1}.$$

a. The closed loop is guaranteed to be stable accoring to the small gain theorem if $||G|| \cdot ||\Delta|| < 1$. We see that $||G|| = 1$ (use `norm(G,inf)` or look at the Bode Plot), so the system is guaranteed to be stable if $||\Delta|| < 1$.

b. The loop gain is given by

$$L = GK = \frac{K}{s+1},$$

and the sensitivity function by

$$S = \frac{1}{1+L} = \frac{1}{1+\frac{K}{s+1}} = \frac{s+1}{s+1+K}.$$

We see that there is one closed loop pole in $s = -(1+K)$, so the system is stable exactly when $K > -1$. We can compare this to the result in **a**, which guarantees that the system is stable when $|K| < 1$.

The different results arise from the fact that the small gain theorem is conservative in nature, i.e. it gives a *sufficient* condition on stability, but that condition may not be *necessary*. The main reason of such a conservatism, is that there is no *a priori* assumptions on Δ. Δ in **a** can be a transfer function of an arbitary order, not just an unknown scalar as in **b**. Looking at the closed loop poles, on the other hand, shows exactly when the system is stable.

3.2 **a.** Block diagrams of the original and the rewritten closed-loop system are shown in Figure 3.1. We have

$$C(s) = \frac{2s+2}{s} \qquad P(s) = \frac{1}{(s+1)^2} \qquad W(s) = \frac{s}{s+2}$$

$$G_{vn}(s) = -\frac{W(s)C(s)}{1+C(s)P(s)} = -\frac{2s^4+6s^3+6s^2+2s}{s^4+4s^3+7s^2+8s+4} = -\frac{2s^3+4s^2+2s}{s^3+3s^2+4s+4}$$

Matlab commands:

```
>> s = tf('s');
>> C = 2*(s+1)/s
>> P = 1/(s+1)^2
>> W = s/(s+2);
>> Gvn = W*feedback(C, P);
```

b. The L_2-gain of G_{vn} is equal to 2.63. This corresponds to the peak magnitude in the Bode Diagram of Figure 3.2.

c. The small gain theorem shows stability for all perturbations satisfying

$$||\Delta|| \cdot ||G_{vn}|| < 1$$

The closed loop system is therefore stable for all perturbations Δ with

$$||\Delta||_\infty < 1/||G_{vn}||_\infty = 0.38$$

Matlab commands:

```
>> norm(Gvn, inf)
>> 1/ans
```

d. Process models are often better (that is, they match the real process more closely) in the low frequency range. As frequency increases there is usually excitation of higher-order dynamics and non-linearities in the real process, which is not covered by the model.

Since we know that there is more uncertainty for high frequencies, this can be used to get some *structure* on the uncertainty block. This structure is given by the extra factor (such factors are usually called *weighting functions*), which effectively makes the uncertainty smaller for low frequencies (approximately when $\omega < 2$). Without this factor, the analysis would assume equal uncertainty for all frequencies, yielding a lower bound on the L_2-gain of Δ. In other words, the system would appear less robust.

3.3 **a.** Matlab commands:

```
>> s = tf('s');
>> P = 1/(s+2);
>> C = (0.81*s+3.6)/(0.225*s)
```

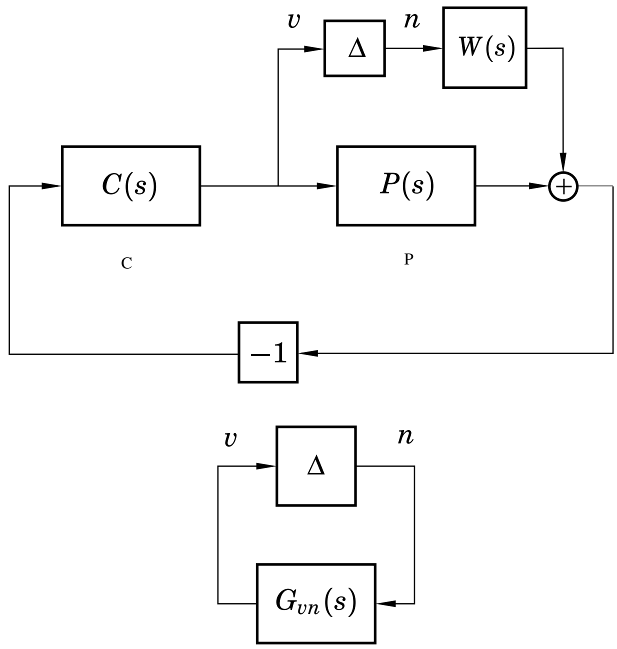


Figure 3.1 Systems for Problem 3.2.

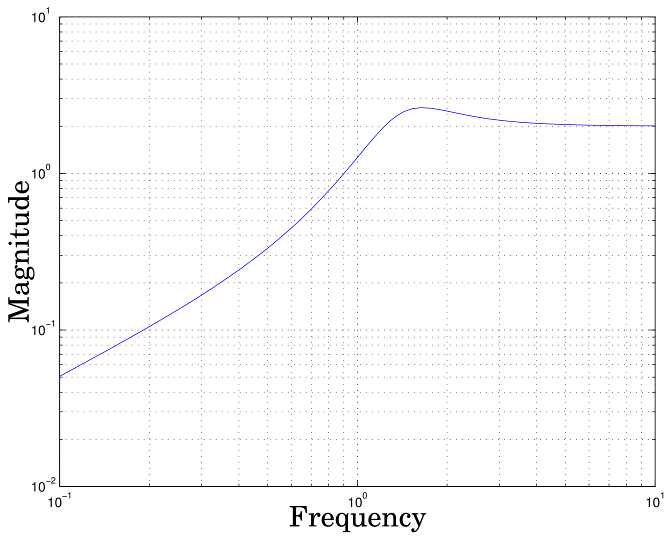


Figure 3.2 Bode magnitude diagram for $G_{vn}(s)$.

```
>> G = feedback(C*P,1);
>> pole(G)

ans =

  -2.8000 + 2.8566i
  -2.8000 - 2.8566i
```

b. The transfer functions are

$$Y(s) = \frac{1}{1+CP}V + \frac{P}{1+CP}D - \frac{CP}{1+CP}N + \frac{CP}{1+CP}R$$

$$Y(s) = SV(s) + PSD(s) + T(R(s) - N(s))$$

$$Y(s) = \begin{pmatrix} S(s) & P(s)S(s) & T(s) & -T(s) \end{pmatrix} \begin{pmatrix} V(s) \\ D(s) \\ R(s) \\ N(s) \end{pmatrix}$$

Matlab commands:

```
>> T = feedback(C*P,1);
>> S = 1-T;
>> bode(T)
>> hold on
>> bode(S)
>> S

Transfer function:
   s^2 + 2 s
----------------
s^2 + 5.6 s + 16
```

```
>> T

Transfer function:
   3.6 s + 16
----------------
s^2 + 5.6 s + 16
```

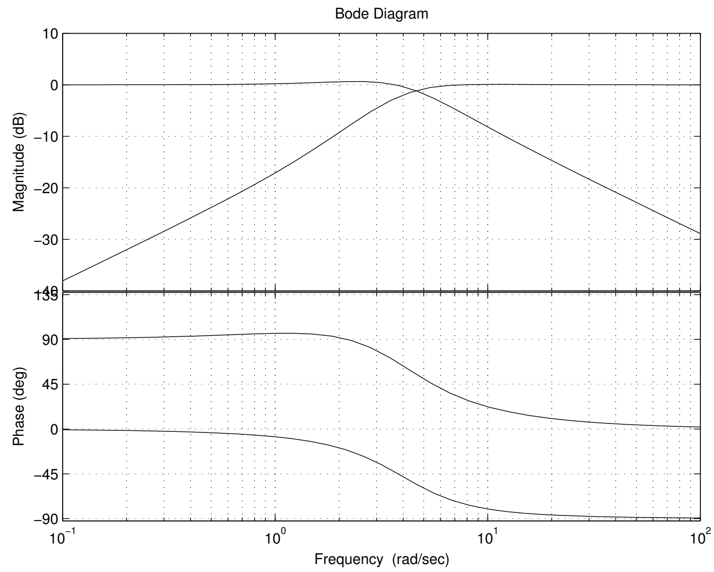


Figure 3.3 Bode diagrams of S and T in Problem 3.3.

c. In the bode plot of the sensitivity function, we see that $\|S(i0.5)\| = -23.8$ dB $= 10^{(-23.8/20)} = 0.0646$

Matlab commands:

```
>> abs(freqresp(S,0.5))

ans =

    0.0644
```

d. We convert $\omega = 2\pi 50$ Hz $= 314.16$ rad/s. In the bode plot of the complementary sensitivity function, we see that $\|T(i314.16)\| = -38.8$ dB $= 10^{(-38.8/20)} = 0.0115$

We have very good attenuation of both load disturbances and measurement noise.

3.4 a. The transfer function from n to v as seen in Figure 3.4 can be written as $H(s) = \frac{-WCP}{1+CP}$ according to the following Matlab commands:

```
>> s = tf('s');
>> W = s/(s+1);
>> P = 1/(s+2);
>> C = (0.81*s+3.6)/(0.225*s);
>> H = -W*feedback(C*P,1);
>> norm(H,inf)
 ans =
```

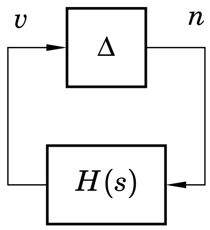


Figure 3.4 Rewritten closed-loop system for Problem 3.4(a).

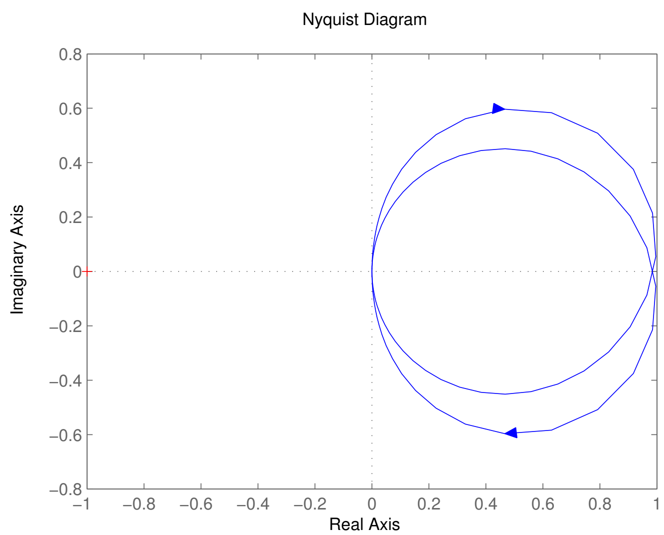


Figure 3.5 Nyquist plot of $-H(s)$ in problem 3.4(b).

```
     1.0072
>> lower_bound = 1/ans
 ans =
     0.9928
```

b. We know that $\Delta(s) = \delta$, a real number. Looking at Figure 3.4 we see that we can apply the Nyquist Criterion to analyze the closed loop stability.

From the Nyquist Plot of $-H(s)$ in Figure 3.5, we see that the closed loop is stable for all $\delta \geq 0$. For negative δ:s, the closed loop will become unstable once the bubble formed by the Nyquist Curve has grown so large that -1 is no longer on its outside. We find this value to be $-\delta = 1.0119$ from the gain margin of $-H$. Thus, the system is stable when $\delta > -1.0119$.

The small gain theorem is easy to use, but it can be conservative, since there is no prior assumptions on structure of uncertainty. With more information about the uncertainty, the bounds can be less conservative and we can allow all positive values of δ as well.

Matlab code:

```
>> nyquist(-H)
>> allmargin(H)
ans =
```

```
 GainMargin: 1.0119
GMFrequency: 2.3256
PhaseMargin: [-4.4628 -18.9141]
PMFrequency: [2.5211 3.1922]
DelayMargin: [2.4614 1.8649]
DMFrequency: [2.5211 3.1922]
     Stable: 1
```

c. In problem **3.2** the uncertainty is *added* to the process. This is called an *additive uncertainty*. Here, the uncertainty is multiplied to the output signal, giving a *multiplicative uncertainty*. In this type of model, the uncertainty is proportional to the process gain.

3.5 $\Phi_u(\omega)$ is an even, scalar, non-negative function. Thus we can divide it into

$$\Phi_u(\omega) = G(i\omega)G(-i\omega)\Phi_e(\omega)$$

where $G(s)$ has its poles and zeroes in the left half-plane and $\Phi_e = 1$ (white noise).

a.

$$\Phi_u(\omega) = \frac{a^2}{\omega^2 + a^2}\Phi_e(\omega) = \frac{a}{i\omega + |a|} \cdot \frac{a}{-i\omega + |a|}$$

So the linear filter is

$$G(s) = \frac{a}{s + |a|}, \qquad a \neq 0.$$

b. In the same way, we get

$$\begin{aligned}\Phi_u(\omega) = \frac{a^2b^2}{(\omega^2 + a^2)(\omega^2 + b^2)}\Phi_e(\omega) \\ = \frac{ab}{(i\omega + |a|)(i\omega + |b|)} \cdot \frac{ab}{(-i\omega + |a|)(-i\omega + |b|)}\end{aligned}$$

$$\Rightarrow \quad G(s) = \frac{ab}{(s + |a|)(s + |b|)}$$

3.6 a. To make a state-space description, we let $x_1 = z$, $x_2 = \dot{z} \quad \Longrightarrow$

$$\begin{aligned}\dot{x}_1 &= x_2, \\ \dot{x}_2 &= \frac{1}{m}(u - k_1 x_2 - v).\end{aligned}$$

In matrix form:

$$\dot{x} = \begin{pmatrix} 0 & 1 \\ 0 & -\frac{k_1}{m} \end{pmatrix} x + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} u + \begin{pmatrix} 0 \\ -\frac{1}{m} \end{pmatrix} v,$$

$$z = \begin{pmatrix} 1 & 0 \end{pmatrix} x.$$

b. We want to find a filter H such that

$$\Phi_v(\omega) = |H(i\omega)|^2 \Phi_e(\omega)$$

Thus $H(s) = \frac{\sqrt{k_0}}{s+|a|}$, which is equivalent to $\dot{v} + |a|v = \sqrt{k_0}\, e$.
Adding a new state $x_3 = v$ to the state-space description, gives

$$\dot{x}_3 = -|a|x_3 + e$$

and

$$\dot{x} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & -\frac{k_1}{m} & -\frac{1}{m} \\ 0 & 0 & -|a| \end{pmatrix} x + \begin{pmatrix} 0 \\ \frac{1}{m} \\ 0 \end{pmatrix} u + \begin{pmatrix} 0 \\ 0 \\ \sqrt{k_0} \end{pmatrix} e$$

$$z = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} x$$

The input-output relation is now given by

$$(p^2 + \frac{k_1 p}{m})\, z = \frac{1}{m}\left(u - \frac{\sqrt{k_0}}{p+|a|}\, e\right)$$

3.7

a. With $\{A, B, C, N\}$ according to the solution of problem 3.6, we have

$$\dot{x} = Ax + Bu + Ne$$
$$y = Cx + n$$

where n has spectral density $\Phi_n \equiv 0.1$.

b. A noise signal with the specified spectral density is given by the output of a linear system with white noise input that has spectral density $\Phi_{w_n} = 0.1$. The transfer function of the system is

$$G_n(s) = \frac{s}{s+|b|} = \frac{s+|b|-|b|}{s+|b|} = 1 - \frac{|b|}{s+|b|}$$

In state-space form, this becomes

$$\dot{x}_4 = -|b|x_4 + |b|w_n$$
$$n = -x_4 + w_n$$

Combining the noise model with our original system gives the expanded state-space description:

$$\dot{x} = \begin{pmatrix} A & 0 \\ 0 & -|b| \end{pmatrix} x + \begin{pmatrix} B \\ 0 \end{pmatrix} u + \begin{pmatrix} N & 0 \\ 0 & |b| \end{pmatrix} \begin{pmatrix} e \\ w_n \end{pmatrix}$$

$$y = \begin{pmatrix} C & -1 \end{pmatrix} x + w_n$$

Note that the disturbance can be described using a transfer function and white noise of any spectral density. For instance, it is often convenient to assume white noise with a spectral density of 1. In this case, the transfer function of the system would be

$$G_n(s) = \frac{\sqrt{0.1}s}{s + |b|}$$

The expanded state space description would then need to be adjusted to account for this.

c. Now, the transfer function of the noise model is $G_n(s) = \frac{1}{s+|b|}$. In state-space form, this is

$$\dot{x}_4 + |b|x_4 = w_n.$$

The expanded system becomes

$$\begin{aligned} \dot{x} &= \begin{pmatrix} A & 0 \\ 0 & -|b| \end{pmatrix} x + \begin{pmatrix} B \\ 0 \end{pmatrix} u + \begin{pmatrix} N & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} e \\ w_n \end{pmatrix} \\ y &= \begin{pmatrix} C & 1 \end{pmatrix} x \end{aligned}$$

As in subproblem b, the disturbance can be described using a transfer function and white noise of any spectral density. Assuming white noise with a spectral density of 1, the transfer function of the system would be

$$G_n(s) = \frac{\sqrt{0.1}}{s + |b|}$$

3.8 w: We want a model that produces step changes given white noise input. The easiest choice is to take an integrator with impulse inputs (impulses have constant spectrum, so they are considered white noise). Introduce the state x_w, with $\dot{x}_w = v$.

n: Since n is periodic with 2 Hz, we need to have a transfer function with a strong resonance peak at the frequency $\omega_0 = 2\pi \cdot 2 = 4\pi$ rad/s. The system needs to have at least order two to have a resonance peak. We can represent the system as

$$N(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2} E(s)$$

with e being white noise input.

To get a distinct resonance peak, we need to have a poorly damped system, so we let $\zeta = 0.01$

Introduce the states $x_{n1} = n$ and $x_{n2} = \dot{n}$

$$\dot{x}_n = \begin{pmatrix} \dot{x}_{n1} \\ \dot{x}_{n2} \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 1 \\ -\omega_0^2 & -2\xi\omega_0 \end{pmatrix}}_{A_n} x_n + \underbrace{\begin{pmatrix} 0 \\ \omega_0^2 \end{pmatrix}}_{B_n} e$$

The expanded model is then

$$\dot{x}_u = \begin{pmatrix} \dot{x} \\ \dot{x}_w \\ \dot{x}_n \end{pmatrix} = \begin{pmatrix} A & N & 0 \\ 0 & 0 & 0 \\ 0 & 0 & A_n \end{pmatrix} x_u + \begin{pmatrix} B \\ 0 \\ 0 \end{pmatrix} u + \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & B_n \end{pmatrix} \begin{pmatrix} v \\ e \end{pmatrix}$$

3.9

a. The spectrum of the wind is of low-pass character with cut-off frequency α. When α is increased, $v(t)$ becomes more similar to white noise, i.e. there is more high-frequency content in the signal. Thus, higher α means more wind gusts.

Alternatively, one could look at the covariance function:

$$R_v(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \Phi_v(\omega) e^{i\omega\tau} d\omega = e^{-\alpha|\tau|}, \quad \alpha > 0.$$

The covariance function has a sharper peak when α is large. That is, the correlation between $v(t)$ and $v(t+\tau)$ is small, meaning that the wind changes more often.

b. Using spectral factorization, the influence of wind can be described as white noise $e(t)$ with intensity 1 filtered through a linear system with transfer function

$$H(s) = \frac{\sqrt{2/\alpha}}{1 + s/\alpha}$$

Thus $Y(s) = G(s)\, H(s) E(s)$, where

$$G(s)\, H(s) = \frac{K\sqrt{2\alpha}}{(\alpha + s)(s^2 + s + 1)} = \frac{K\sqrt{2\alpha}}{s^3 + (1+\alpha)s^2 + (1+\alpha)s + \alpha}.$$

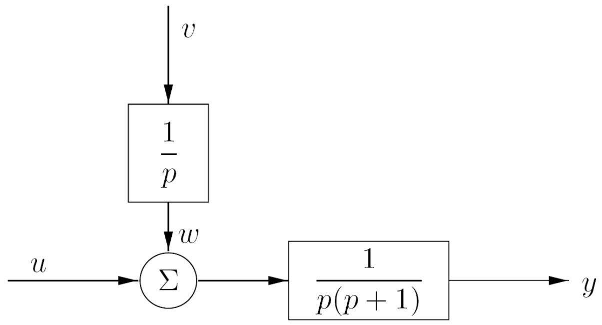
The variance of the output is

$$\begin{aligned} \operatorname{Var}(y) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} |G(i\omega) H(i\omega)|^2 d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left| \frac{K\sqrt{2\alpha}}{(i\omega)^3 + (1+\alpha)(i\omega)^2 + (1+\alpha)i\omega + \alpha} \right|^2 d\omega \\ &= \frac{K^2(1+\alpha)}{1 + \alpha + \alpha^2}. \end{aligned}$$

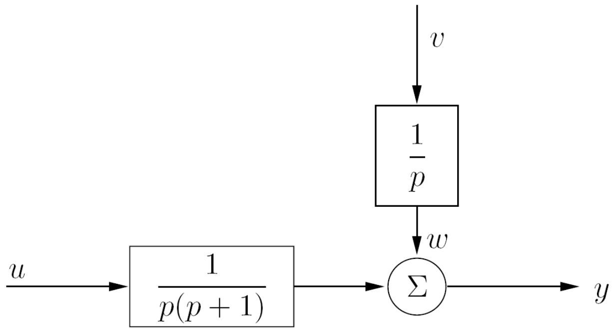
Apparently, the variance increases with wind strength, which is no surprise. However, the variance decreases with the amount of wind gusts. The reason is that a low amount of gusts means that there are longer periods of almost constant wind force, where the swing is displaced far from the origin. A lot of gusts, on the other hand, results in the wind force changing sign frequently, more or less cancelling its own effect a lot of the time.

3.10

a. (i)



(ii)



$v(t)$ is a unit disturbance

b. (i)

$$\dot{x} = \overbrace{\begin{pmatrix} 0 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{pmatrix}}^{A} x + \overbrace{\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}}^{B} u + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v$$
$$y = \underbrace{\begin{pmatrix} 1 & 0 & 0 \end{pmatrix}}_{C} x.$$

(ii)

$$\dot{x} = \overbrace{\begin{pmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}}^{A} x + \overbrace{\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}}^{B} u + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v$$
$$y = \underbrace{\begin{pmatrix} 1 & 0 & 1 \end{pmatrix}}_{C} x.$$

c. (i) $w(t)$ could be an offset current on the input to the motor, and/or a step disturbance in the load.

(ii) In this case $w(t)$ is a measurement disturbance, i.e. an additive error (constant) in the angle measurement.

Solutions to Exercise 4. Loop Shaping

4.1

a. Since we cannot change the phase of the system using a P-controller, higher gain will lead to lower phase margin (as the phase approaches -180 for high frequencies).

Higher gain will also decrease stationary errors, but increase the maximum peak in the sensitivity function (making the system very sensitive to measurement noise).

```
>> figure(1)
>> step(P/(1+0.1*P),P/(1+1*P),P/(1+5*P),P/(1+10*P));
>> title('Step responses')
>> figure(2)
>> bode(P/(1+0.1*P),P/(1+1*P),P/(1+5*P),P/(1+10*P));
>> title('Transfer functions from load disturbance');
>> figure(3)
>> bode(1/(1+0.1*P),1/(1+1*P),1/(1+5*P),1/(1+10*P));
>> title('Sensitivity functions');
```

b. It is not possible to achieve good behavior with a PI-controller, but try to get it as good as possible:

```
>> figure(1)
>> K= ... ; Ti = ... ;
>> C = tf(K*[1 1/Ti],[1 0]);
>> step(P/(1+C*P);
```

4.2 **a.** From the basic course: We calculate the gain $\|C(i\omega)\| = 1/\sqrt{\omega^2/a^2+1}$ and use log scale. Then

$$\log|C(i\omega)| = -0.5\log(\omega^2/a^2+1) \approx \begin{cases} 0 & \omega << a \\ \log(a)-\log(\omega) & \omega >> a \end{cases}$$

and the two lines meet where $\omega = a$ (the breakpoint). Also, the phase is at -45° at $\omega = a$, starts at 0° and ends at -90°.

We can add a pole to the controller if we want to decrease gain for higher frequencies, e.g. to limit the cut-off frequency ω_c. It is often the case that we want to increase the gain at low frequencies, but keep it low at high frequencies. We can then use a controller of the type $C(s) = K/(s/a+1)$ with a pole to limit high frequency gain and a static gain larger than one to increase the low frequency gain.

```
>> C01 = tf([1],[1/0.1 1]);
>> C1 = tf([1],[1/1 1]);
>> C5 = tf([1],[1/5 1]);
>> bode(C01, C1, C5);
```

b. The same as in (a), except that a zero breaks the gain **up** at b.

$$\log|C(i\omega)| = 0.5\log(\omega^2/b^2+1) \approx \begin{cases} 0 & \omega << b \\ \log(\omega)-\log(b) & \omega >> b \end{cases}$$

We can add a zero to the controller to increase gain at high frequencies in order to increase the cut-off frequency ω_c. Also, since the phase of the zero goes to $+90°$, we increase the phase margin by adding a zero.

4.3 The following Matlab code shows some relevant plots for a design:

```
>> s = tf('s');
>> C = ...;        % Make up your own design

>> figure(1)
>> margin(C*P)     % Plot open loop frequency response

>> figure(2)
>> nichols(C*P)    % Check stability margins in Nichols plot
>> ngrid

>> figure(3)
>> % Plot load step response and control signal
>> subplot(2,1,1)
>> step(P/(1+P*C), P/(1+10*P)); % Compare to P-controller
>> title('Load step response');
>> subplot(2,1,2)
>> step(P*C/(1+P*C), 10*P/(1+10*P))
>> title('Control signal for load step');
```

4.4

a. The ideal frequency response is $G_{yr} \equiv 1$. Then we would always have $y = r$. However, achieving something close to this would require very aggressive control, so that is not a good idea. (The controller would need to invert the process dynamics, resulting in second-order derivative action on the control error).

b. We want to shape $F(s)$ so that the constraints on the control signal are respected, for a step change in the reference. This may be achieved by reducing the bandwidth.

4.5 Plot the Bode diagram for $G_o(s)$ in Matlab or use the command

```
>> [Gm,Pm,Wcg,Wcp] = margin(G_o)
```

to calculate the cut-off frequency $\omega_c = 0.73$ and the phase margin $\phi_m = 20.7°$. To reach the aim of a $\phi_{m,desired} = 50°$, the controller has to increase the phase at the cut-off frequency with approx 30°. We use the lead compensation given by

$$G_k(s) = KN\frac{s+b}{s+bN}$$

with the phase

$$\phi = \arctan\left(\frac{s}{b}\right) - \arctan\left(\frac{s}{bN}\right)$$

The maximum of the phase compensation for the compensator is at the frequency $b\sqrt{N}$, which preferably should coincide with ω_c, hence $N = (\omega_c/b)^2$.

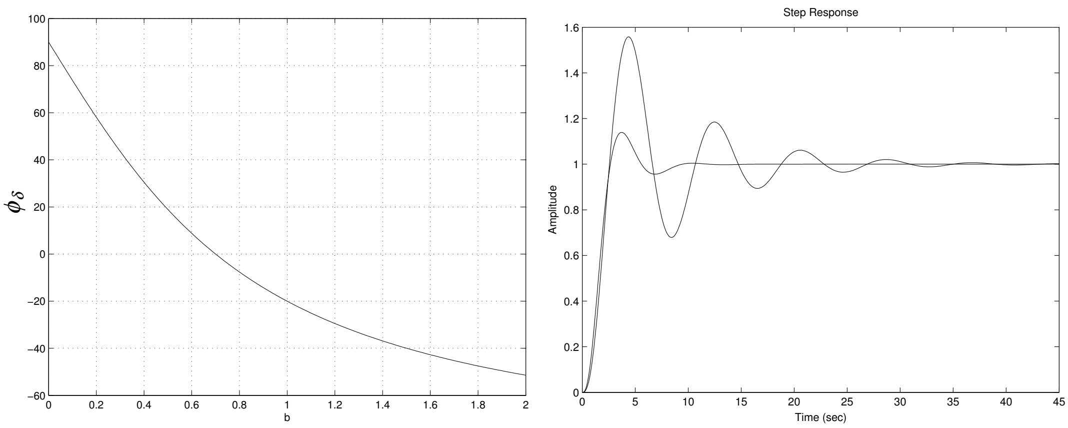


Figure 4.1 To the left: Plot of ϕ_δ against b. To the right: Step response from the original system and the compensated system in Problem 4.5.

Plot the phase addition of the compensator given by

$$\phi_\delta = \arctan\left(\frac{\omega_c}{b}\right) - \arctan\left(\frac{b}{\omega_c}\right)$$

and determine that the factor $b \approx 0.4$ for $\phi_\delta = 30°$ (see Figure 4.1). To keep the cut-off frequency invariant the gain of the compensator has to be calculated from $|G_k(i\omega_c)G_o(i\omega_c)| = K\sqrt{N}\cdot 1$ gives $K = \frac{1}{\sqrt{N}} = 0.55$. Plot the step response by the commands:

```
>> G_l=tf(K*N*[1 b],[1 b*N])
>> step(G_o*G_l/(1+(G_o*G_l))
```

The stationary error:

$$E(s) = \frac{1}{1+G_kG_o}U(s) = \frac{s(s+0.5)(s+3)(s+bN)}{s(s+0.5)(s+3)(s+bN)+2KN(s+b)}U(s)$$

The Laplace transform of a ramp function is $U(s) = 1/s^2$ and the error is

$$\lim_{s\to 0} sE(s) = \frac{1.5}{2K} = 1.37$$

which fulfills the specification.

4.6 a. The transfer function from d to y is given by

$$G_{yd}(s) = \frac{P}{1+PC}$$

For frequencies $\omega \leq 0.5$ (approximately), it can be seen in the Bode diagram that both $|P(i\omega)| \gg 1$ and $|P(i\omega)C(i\omega)| \gg 1$. Therefore $G_{yd}(s) \approx \frac{1}{C}$, and $|C(i\omega)|$ becomes larger than 1 for frequencies $\omega \leq 0.02$.

The magnitude of $G_{yd}(s)$ is thus smaller than 1 in a frequency range of approximately $[0, 0.02]$, thus $\omega_b = 0.02$ rad/s.

This can also be seen as the frequency point where $|PC|$ becomes larger than $|P|$ in the bode diagram.

b. To increase ω_b, we would like to increase the gain of $C(i\omega)$ for frequencies $\omega > 0.02$. This is done by moving the zero in $C(s)$ (the break-point in the Bode diagram) from 0.02 to some higher frequency.

Choose e.g. $r = 0.1$. Motivation:

- As $G_{yd}(s) \approx \frac{1}{C}$, and $|C(i\omega)|$ now becomes larger than 1 for frequencies $\omega \leq 0.1$, ω_b has been increased to about 0.1.
- The cut-off frequency for $r = 0.02$ is $\omega_c \approx 0.8$. As this frequency is higher than the new break-point 0.1, $C(i\omega_c) \approx 1$ still holds $\Rightarrow$ the cut-off frequency stays the same.

Solutions to Exercise 5. Multivariable Zeros, Singular Values and Controllability/Observability

5.1 **a.** In Matlab, we may derive the controllability- and observability matrices using

```
>> Wc = ctrb(A,B)

Wc =

     1    -1     1
     1    -2     4
     0     0     0

>> rank(Wc)

ans =

     2

>> Wo = obsv(A,C)

Wo =

     1     0     1
    -1     0    -3
     1     0     9

>> rank(Wo)

ans =

     2
```

Since the system is on diagonal form we can see, using theorem 3.1 in the course book (Glad&Ljung), that the uncontrollable mode corresponds to the third state (as that row in the B matrix is 0). By theorem 3.2 in the course book, the unobservable mode is determined to be the second state in a similar fashion (the column of C equal to 0).

The system is illustrated in the block diagram in Figure 5.1. We can see that the state x_2 will not influence y, and is therefore not observable. We can also see that the control signal u will not affect the state x_3, and therefore this state is not controllable.

b. The transfer function is simply

$$G(s) = C(sI - A)^{-1}B = \frac{1}{s+1}$$

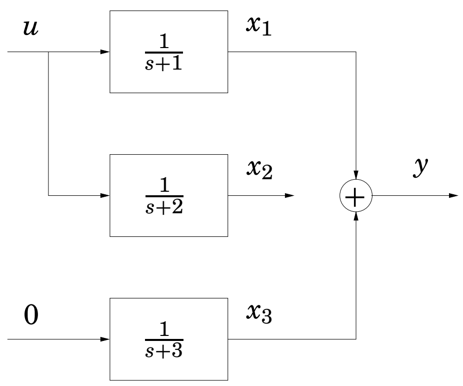


Figure 5.1

and the system can thus be represented as a minimal realization in state space form of order 1. Note that this corresponds to the first subsystem in Figure 5.1 which is both observable and controllable.

If we are only interested in the relationship between u and y, we can use the resulting first order transfer function $G(s)$. However, the original third order state space model contains additional information, as seen in Figure 5.1. The second and third subsystems in this model may represent physical entities of the plant that must be taken into account. If we need to influence x_3 or monitor x_2, additional sensors or actuators are needed.

5.2

a. First of all, define the system in Matlab

```
>> A = [-0.21 0.2;0.2 -0.21];
>> B = 0.01*eye(2);
>> C = eye(2);
>> D = 0;
>> sys = ss(A,B,C,D);
```

The controllability Gramian is calculated using

```
>> W = gram(sys,'c')

W =

    0.0026    0.0024
    0.0024    0.0026
```

b. Recall the formula:

$$\int_0^{\infty} u^2(s)ds \geq x^T(\infty)W^{-1}x(\infty)$$

Therefore to identify the hardest to control state direction, we need to calculate the eigenvalues and the eigenvectors of W^{-1} :

```
[T L] = eig(inv(W))

T =

   -0.7071   -0.7071
   -0.7071    0.7071

L =

   1.0e+03 *

    0.2000         0
         0    8.2000
```

Apparently one eigenvalue of the inverse of the Gramian is almost 40 times larger than the other. Hence one state direction is poorly controllable.

Inspection of the corresponding eigenvectors shows that the small eigenvalue correponds to a state direction where both temperatures move in the same way, while the poorly controllable state direction correponds to temperatures moving in opposite directions.

5.3 a. Continuing the code we get

```
>> syms c1 c2 c3 c4 c5
>> C = [c1 c2 c3 c4 c5];
>> Wo = [C;C*A;C*A^2;C*A^3;C*A^4];
>> det(Wo)

ans =

0

>> rank(Wo)

ans =

4
```

Since the system does not have full rank (5) we see that no matter how we choose C (when it's a vector), the system can never be made observable. This means that we need information from more than just one signal to make the system observable.

b. Determine the eigenvectors of the system

```
>> [V,D]=eig(A)

V =

         0    0.3333    0.4286   -0.0261    0.0206
         0    0.6667   -0.8571    0.7973   -0.3916
    1.0000    0.6667    0.2857   -0.0399    0.0196
         0         0         0    0.6017         0
```

```
             0         0         0         0    0.9197
...
```

Rewrite the system on diagonal form using the variable change $x(t) = Vz(t)$

$$\begin{aligned}
\dot{x}(t) &= V\dot{z}(t) = AVz(t) + Bu(t) \Rightarrow \\
\dot{z}(t) &= V^{-1}AVz(t) + V^{-1}Bu(t) = \Lambda z(t) + V^{-1}Bu(t) \\
y(t) &= CVz(t)
\end{aligned}$$

where Λ is a diagonal matrix with the eigenvalues in the diagonal. Now that we have the system on the wanted form, we can determine if there are any columns in CV that are zero

```
>> C*V

ans =

             0    0.3333    0.4286   -0.0261    0.0206
             0    0.6667   -0.8571    0.7973   -0.3916
```

The first state in z therefore corresponds to the unobservable mode. In the original variables this is the third state:

```
>> V*[1;0;0;0;0]

ans =

     0
     0
     1
     0
     0
```

So, the third state is the unobservable mode.

5.4 Figure 5.2 depicts the observable system. Obviously the problem is in the pole $p_0 = -3$. We control directly the plant P_1, and we observe the output of plant P_2. It means that we observe the effect of the pole $p_0 = -3$, but due to pole-zero cancellation, we cannot control it.

Similarly for system in Figure 5.3, we control the plant P_1, and the pole $p_0 = -3$ is controllable, but the effect of that pole is cancelled by the zero and we do not observe it. Hence the whole system is not observable.

Figure 5.2 Block diagram for problem 5.4.

5.5 **a.** The biggest subdeterminant of the transfer function matrix is

$$\frac{(s+1)}{(s+2)^2} + \frac{1}{(s+2)^2} = \frac{1}{(s+2)}$$

Figure 5.3 Block diagram for problem 5.4.

Furthermore, the matrix elements in themselves are subdeterminants. The pole polynomial, i.e. the least common denominator of all subdeterminants, is then

$$p(s) = (s + 2)$$

This means that the system has a pole in $s = -2$. The system can thus be realized on state space form of order 1.

The largest possible subdeterminant was

$$\frac{1}{(s+2)}$$

The zero polynomial is thus just a constant and the system does therefore not have any zeros.

Note, that we basically calculated the determinant of the transfer matrix $\det(G(s))$ and took its denominator as the pole polynomial and numerator as the zero polynomial.

b.

$$G(s) = \begin{pmatrix} \frac{1}{s+2} & -\frac{1}{s+2} \\ \frac{1}{s+2} & \frac{s+1}{s+2} \end{pmatrix} = \begin{pmatrix} \frac{1}{s+2} & -\frac{1}{s+2} \\ \frac{1}{s+2} & 1 - \frac{1}{s+2} \end{pmatrix} = \frac{1}{s+2}\begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$
$$= \frac{1}{s+2}\begin{pmatrix} 1 \\ 1 \end{pmatrix}\begin{pmatrix} 1 & -1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

The state-space realization can now be written as

$$\frac{dx}{dt} = -2x + \begin{pmatrix} 1 & -1 \end{pmatrix} u$$
$$y = \begin{pmatrix} 1 \\ 1 \end{pmatrix} x + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} u$$

c. The singular value plot (see Figure 5.4) is drawn using the command `sigma`. The $\mathcal{L}_2$ gain $||G||_\infty$ is the largest singular value of $G(s)$ across all frequencies ω, from the figure we can see that $||G||_\infty = 1$ in this case. We also see that the largest gain of this system is achieved at high frequencies.

5.6 a. To determine the frequency response at a certain frequency ω, it's handy to use the Matlab command `freqresp`. To calculate the singular values together with the U and V matrices, use the function `svd`. The Matlab code can look like this:

```
>> s = tf('s');
>> G = [1 1/s];
>> [U,S,V] = svd(freqresp(G,1))
```

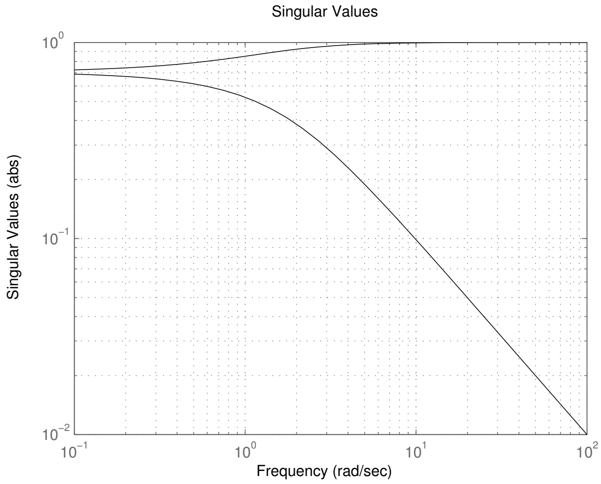


Figure 5.4

```
U =

     1


S =

    1.4142         0


V =

   0.7071          0 + 0.7071i
   0 + 0.7071i    0.7071
```

The system has one singular value, given by the diagonal element of the matrix S. The maximum gain, corresponding to the highest singular value, is $\overline{\sigma} = 1.4142$. The first column of V, $v_1 = (0.7071 \quad 0.7071i)^T$, corresponds to the input direction that gives the maximum gain $\overline{\sigma}$. Since the system has two inputs and only one output, there will always be an input direction that gives zero output (where the inputs cancel each other). The second column of V gives us this direction, $v_2 = (0.7071i \quad 07071)^T$.

b. If the input signal is a sinusoid with frequency $\omega = 1$ rad/s, the complex numbers will correspond to a phase shift of this sinusoid. The input direction giving the highest gain is $v_1 = [0.7071 \quad 0.7071i]^T$, meaning that the second input has 90° of phase lead before the first.

The first input comes through the system unchanged; the second goes through an integrator, causing a phase lag of 90°. Thus the input direction $v_1 = [0.7071 \quad 0.7071i]^T$ will cause the two sinusoids that sum up at the output to be in phase; resulting in maximal gain.

If we instead use the lowest gain input direction $v_2 = [0.7071i \quad 0.7071]^T$, the second input will have a phase lag of 90°, causing a 180° phase lag at the output. The two signals will cancel at the output, resulting in minimal gain.

5.7 **a.**
```
>> s = tf('s');
>> G = 1/(75*s+1)*[87.8 -86.4;108.2 -109.6];
>> sigma(G)
>> grid
```
See Figure 5.5 for the Matlab plot.

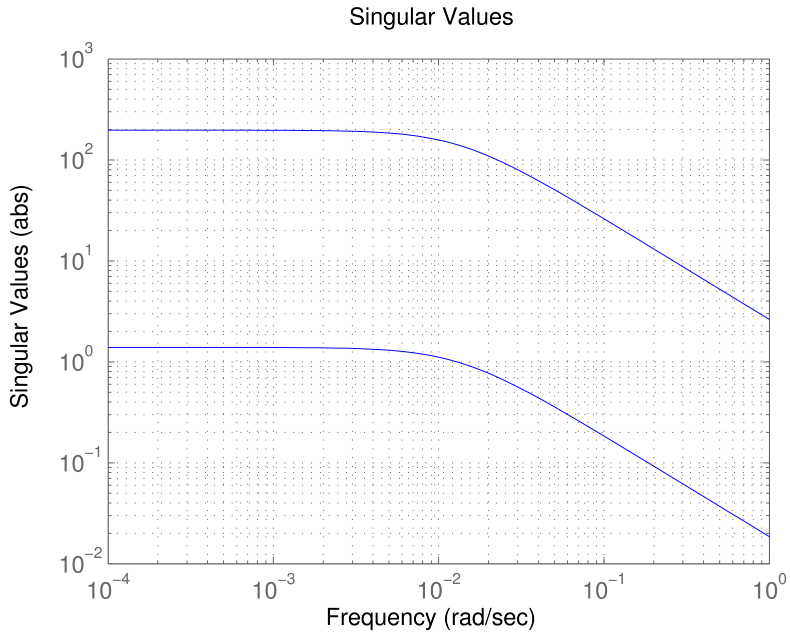


Figure 5.5 Singular value plot for Problem 5.7

b. Calculate the frequency responses at the given frequencies

```
>> Gfr1 = freqresp(G,0)

Gfr1 =

   87.8000  -86.4000
  108.2000 -109.6000

>> Gfr2 = freqresp(G,0.1)

Gfr2 =

   1.5336-11.5022i  -1.5092+11.3188i
   1.8900-14.1747i  -1.9144+14.3581i
```

The gain of a transfer matrix at a particular frequency is computed as $\sup_{d\neq 0} \frac{\|G(i\omega)d\|_2}{\|d\|_2}$. Note that $\|\cdot\|_2$ is a matrix 2-norm, not an L_2 norm, and d

is a constant vector (or a direction). If we choose a particular direction d_0 then the supremum disappears and the gain is given by $\dfrac{\|G(i\omega)d_0\|_2}{\|d_0\|_2}$.

Thus the gains are given by

$$\frac{\|G(0)d_1\|_2}{\|d_1\|_2} = \frac{\|\,[-5.1 \quad -8.6\,]^T\,\|_2}{\|\,[\,0.6713 \quad 0.7412\,]^T\,\|_2} = \frac{\sqrt{(-5.1)^2 + (-8.6)^2}}{\sqrt{(0.6713)^2 + (0.7412)^2}} = \frac{10.0}{1}$$

$$\frac{\|G(0)d_2\|_2}{\|d_2\|_2} = 139.3$$

$$\frac{\|G(0.1i)d_1\|_2}{\|d_1\|_2} = 1.3$$

$$\frac{\|G(0.1i)d_2\|_2}{\|d_2\|_2} = 18.4$$

They can also be calculated in Matlab using

```
>> d1 = [0.6713;0.7412];
>> d2 = [1;0];
>> norm(Gfr1*d1),norm(Gfr1*d2),norm(Gfr2*d1),norm(Gfr2*d2)

ans =

    9.9990

ans =

  139.3416

ans =

    1.3215

ans =

   18.4159
```

c. Using Matlab:

```
>> [U,S,V] = svd(Gfr1)

U =

   -0.6246   -0.7809
   -0.7809    0.6246

S =

  197.2087         0
         0    1.3914

V =
```

```
-0.7066   -0.7077
 0.7077   -0.7066
```

The maximum gain is $\overline{\sigma} = 197.2$ and the minimum gain is $\underline{\sigma} = 1.39$. The input direction associated with the maximum gain is $v_1 = [-0.7066 \quad 0.7077]^T$. The input direction giving the least gain is $v_2 = [-0.7077 \quad -0.7066]^T$. These directions are constant for all frequencies. The reason is that the denominators of all matrix elements are the same, which gives

$$G(i\omega) = \frac{1}{75i\omega + 1} G(0).$$

Let $G(0) = U\Sigma V^*$. We then have $G(i\omega) = U\left(\frac{1}{75i\omega+1}\Sigma\right)V^*$, and we can see that ω will only change the singular value matrix Σ, not the direction matrices U and V.

Solutions to Exercise 6. Fundamental Limitations

6.1 **a.** The transfer function of the process $P(s)$ is given by

$$P(s) = \frac{s}{s^2 + 2s + 1}$$

and the zero is located in the origin.

b. The sensitivity function is given by $S(s) = \dfrac{1}{1 + C(s)P(s)}$ and it will be one at low frequencies since $P(0) = 0$.

c. The error $e(t)$ is given by $r(t) - \theta(t)$ and the static error is then given by the final value theorem which can be used if $sE(s)$ is asymptotically stable (all poles have a negative real-part).

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s)$$

Here the transfer function from r to e is given by:

$$G_{re}(s) = \frac{1}{1 + C(s)P(s)}$$

The following result is obtained if $r(t)$ is assumed to be a step with the gain a.

$$\lim_{s\to 0} sE(s) = \lim_{s\to 0} sG_{re}(s)R(s) = a$$

since $P(0) = 0$ (and thereby $G_{re}(0) = 1$) at low frequencies and $R(s) = a/s$. This means that the ball will not follow a reference trajectory that changes step-wise; there will be a static error equal to a. Hence, no matter the reference value, the ball will end up in the bottom of the cylinder.

An alternative explanation is that the sensitivity function S is 1 at low frequencies, therefore

$$T(s) = \frac{C(s)P(s)}{1 + C(s)P(s)} = 1 - S$$

must be small and then $y(t)$ does not follow $r(t)$.

d. The transfer function for the open loop with a PI controller is given by:

$$P(s)C(s) = \frac{s}{s^2 + 2s + 1} K \frac{s + 1/T_i}{s} = K \frac{s + 1/T_i}{s^2 + 2s + 1}$$

Here the process zero is canceled by the controller. The stationary gain of the open respectively closed system are given by:

$$PC(0) = \frac{K}{T_i}$$

and

$$G_{re}(0) = \frac{1}{1 + PC(0)} = \frac{T_i}{K + T_i}$$

The static error

$$\lim_{t\to\infty} e(t)$$

is given by:

$$\lim_{s\to 0} sG_{re}(s)R(s) = \frac{aT_i}{K + T_i}$$

if $R(s)$ is assumed to be $R(s) = a/s$ and $sE(s)$ is asymptotically stable. Thus this PI controller does not remove the static error, although it can be decreased by increasing the controller gain. Also note that even if this means that the error will be static, that is not the case for the control signal, which will grow linearly. This follows from the physics of the process, since holding the ball at some constant angle requires a constant force to oppose gravity. This constant force translates to a constant cylinder acceleration.

6.2 The sensitivity function is given by:

$$S(s) = \frac{1}{1 + C(s)P(s)}$$

From this it follows that

$$S(i\omega) = \frac{1}{1 + C(i\omega)P(i\omega)}$$

a. In the case of a purely imaginary process *pole* in $i\omega_p$ we have

$$P(i\omega_p) = \infty$$

and consequently

$$S(i\omega_p) = 0$$

b. A measurement disturbance n with frequency ω_p will have a vanishing effect on y and e, since

$$S(i\omega_p) = 0$$

c. No stabilizing controller can change the fact that $S(i\omega_p) = 0$. Cancellations of unstable poles should always be avoided.

d. In the case of a purely imaginary process *zero* in $i\omega_z$ we have

$$P(i\omega_z) = 0$$

and consequently

$$S(i\omega_z) = 1$$

e. The transfer function from n to x is given by $-T(s)$, where T is the complementary sensitivity function. Since $S(i\omega_z) = 1$ and $S + T = 1$ it must hold that $T(i\omega_z) = 0$, i.e. an output disturbance with frequency ω_z will have no effect on x and hence appear directly in y regardless of the controller $C(s)$.

6.3 **a.** The transfer function from n to x is given by

$$G_{n\to x}(s) = -\frac{P(s)C(s)}{1+P(s)C(s)}$$

We want to determine $C(s)$ such that $G_{n\to x}(s) = 5/(s+5)$. This gives the equation

$$-\frac{P(s)C(s)}{1+P(s)C(s)} = \frac{5}{s+5} \Longrightarrow C(s) = -\frac{\frac{5}{s+5}}{P(s)\cdot(1+\frac{5}{s+5})}$$

Inserting $P(s) = (3-s)/(s+1)^2$, we obtain

$$C(s) = -\frac{5\cdot(s+1)^2}{(3-s)(s+10)}$$

However, this is not a stabilizing controller. For example, the transfer function from n to u, $G_{n\to u} = -\frac{C(s)}{1+P(s)C(s)}$, will be unstable because of the cancellation of the unstable zero in $P(s)$.

b. The specification

$$|S(i\omega)| \le \frac{2\omega}{\sqrt{\omega^2+36}} \qquad \omega \in I\!R$$

is obviously equivalent to

$$\sup_{\omega}\left|\frac{\sqrt{\omega^2+36}}{2\omega}S(i\omega)\right| \le 1$$

However, $W_a(i\omega) = \frac{i\omega+a}{2i\omega}$ gives $|W_a(iw)| = \frac{\sqrt{\omega^2+36}}{2\omega}$ so the specification can equivalently be written

$$\sup_{\omega}|W_a(i\omega)S(i\omega)| \le 1$$

There is a right half plane zero in $z = 3$. According to Theorem 7.4 in [Glad&Ljung] this makes the specification impossible to satisfy unless $|W_a(z)| \le 1$. We see here that $|W_a(3)| = |\frac{3+6}{2\cdot 3}| = \frac{3}{2} > 1$, so the specification is impossible to satisfy for $a = 6$.

c. The Bode plot of $P(s)$ is given in Figure 6.1 and the sensitivity function when $C(s) = 1$ is given in Figure 6.2 together with the specification. Since the specification $\frac{2\omega}{\sqrt{\omega^2+1}} = 0$ when $\omega = 0$ the controller $C(s)$ must contain an integrator. To avoid instability we must also lift the phase curve through adding a zero and decrease the gain in the open-loop, $P(s)C(s)$. A controller on the form

$$C(s) = K\cdot\frac{s/b+1}{s}$$

with e.g. $K = 0.17$, $b = 0.5$ will do the job.

Hint: To plot the specification on top of the Bode plot of S the following Matlab commands can be used:

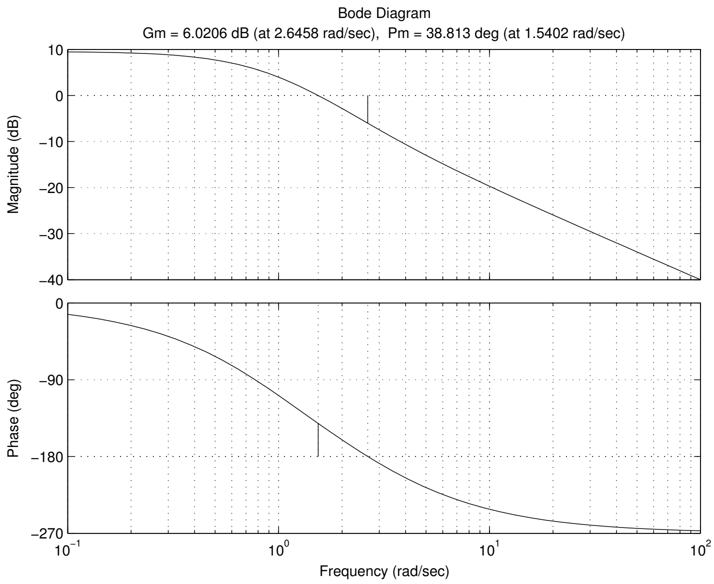


Figure 6.1 Bode plot of $P(s)$ in Problem 6.3.

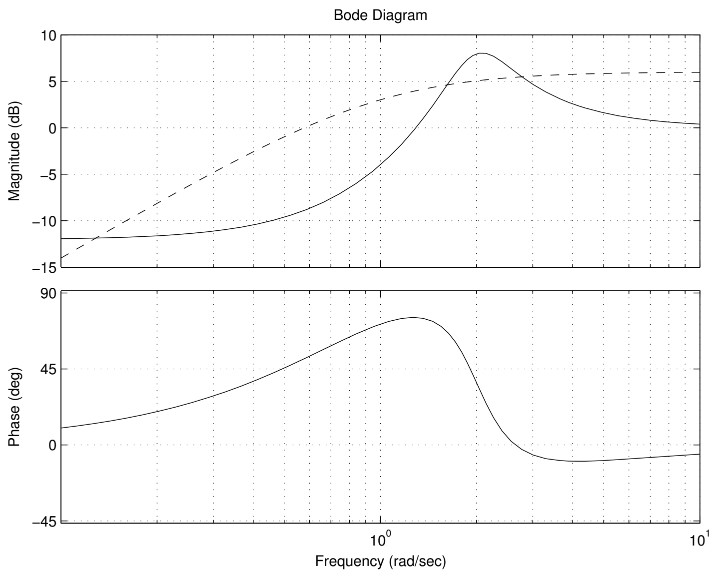


Figure 6.2 Sensitivity function when $C(s) = 1$ and the specification (dashed) in Problem 6.3.

```
>> [mag,fas,w] = bode(S);
>> bode(S)
>> loglog(w,20*log10(2.*w./sqrt(w.*w+1)))
(20*log10 converts to dB)
```

6.4 The first case is impossible, because there is a time-delay of 2 seconds in the plant, so the control signal will affect the output with this delay. Thus, the controller would need to be non-causal to achieve the specification.

The second specification in the figure says that the gain should be below 2 (actually the requirement is closer to ≈ 1.6).

From the lecture notes (lecture 7) we see that if there is an unstable pole p and an unstable zero z, we have the following fundamental limitation:

$$||S||_\infty \geq \left|\frac{z+p}{z-p}\right|$$

We know that $S + T = 1$, so in this case, we have

$$||T||_\infty = ||S-1||_\infty \geq ||S||_\infty - 1 \geq \left|\frac{20+10}{20-10}\right| - 1 = 2$$

where the first inequality follows from the reversed triangle inequality

$$|||x|| - ||y||| \leq ||x-y||$$

It is also possible to get to the same conclusion without using the unstable zero, since the existence of a fast unstable pole is enough the make the specification impossible to achieve.

The third case is possible. If proportional control, $C(s) = K$ is used the closed loop transfer function becomes $G(s) = \frac{K}{s-3+K}$. For stability it is required that $K > 3$. The static gain is given by $\frac{K}{K-3}$. Since it is a first order system there will be no overshoot in the step response, which means that a P-controller with $K > 6$ will fulfill the specification.

6.5 **a.** Assume $\sup_\omega |W_S(i\omega)S(i\omega)| \leq 1$ and $\sup_\omega |W_T(i\omega)T(i\omega)| \leq 1$ are satisfied.

We know that $1 = |S(s_0) + T(s_0)| \leq |S(s_0)| + |T(s_0)|$ (triangle inequality).

If $|W_S(s_0)| > 2$ for some right half place s_0, then $|S(s_0)| < 1/2$, since

$$\sup_\omega |W_S(i\omega)S(i\omega)| = \sup_{Re(s)\geq 0} |W_S(s)S(s)| \leq 1$$

Analogously we get $|T(s_0)| < 1/2$. Then

$$1 = |S(s_0) + T(s_0)| \leq |S(s_0)| + |T(s_0)| < 1$$

and we arrive to contradiction. Hence either $|W_T(s)T(s)| > 1$ or $|W_S(s)S(s)| > 1$ and the corresponding specification must fail.

b. We have

$$W_S(1) = \left(\frac{1+0.1}{1}\right)^n = \left(\frac{1+10}{10}\right)^n = W_T(1)$$

and the value is larger than 2 for $n \geq 8$. Hence, the statement in **a** shows that the specifications are incompatible.

6.6

a. The requirements on $|S(i\omega)| = \bar{\sigma}(S(i\omega))$ and $|T(i\omega)| = \bar{\sigma}(T(i\omega))$ may be formulated as

$$|S(i\omega)| \leq \tfrac{1}{10}, \quad \omega \leq 0.1, \qquad |T(i\omega)| \leq \tfrac{1}{10}, \quad \omega \geq 2$$

$$|S(0)| \leq \tfrac{1}{100}$$

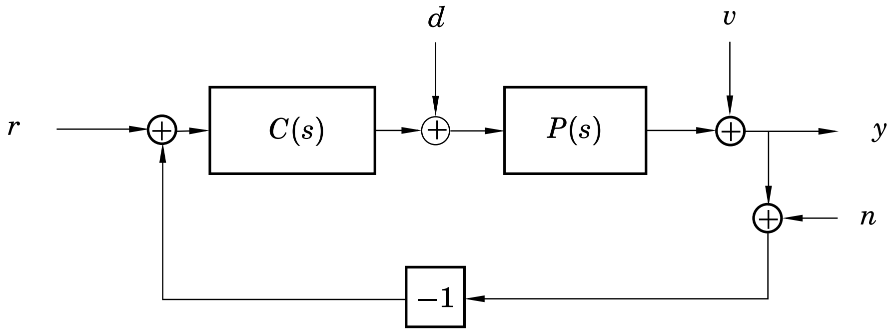


Figure 6.3 System in Problem 6.6

b. The specifications in **a** can be formulated with weighting functions W_S and W_T as

$$|S(i\omega)| \le |W_S^{-1}(i\omega)|, \quad \forall\omega$$
$$|T(i\omega)| \le |W_T^{-1}(i\omega)|, \quad \forall\omega$$

If e.g. W_S^{-1} and W_T^{-1} are chosen to be first order functions, according to

$$W_S^{-1}(s) = a_1\left(1+\frac{s}{b_1}\right), \qquad W_T^{-1}(s) = \frac{a_2}{s}\left(1+\frac{s}{b_2}\right)$$

we get

$$W_S^{-1}(s) = \frac{1}{100}(1+100s), \qquad W_T^{-1}(s) = \frac{0.14}{s}\left(1+\frac{s}{2}\right)$$

c. The corresponding conditions on the open-loop gain $L(i\omega) = P(i\omega)C(i\omega)$ are

$$\bar\sigma(S(0)) = \bar\sigma((I+L(0))^{-1}) = (\underline{\sigma}(I+L(0)))^{-1} \le \frac{1}{100}$$
$$\Rightarrow \underline{\sigma}(I+L(0)) \ge 100$$
$$\Rightarrow \underline{\sigma}(I+L(0)) \ge \underline{\sigma}(L(0)) - \underline{\sigma}(I) = \underline{\sigma}(L(0)) - 1 \ge 100$$
$$\Rightarrow \underline{\sigma}(L(0)) \ge 101$$

The condition that $|S(i\omega)| \le \frac{1}{10}, \quad \omega \le 0.1$ is derived in an analogous way to

$$\Rightarrow \underline{\sigma}(L(i\omega)) \ge 11, \quad \omega \le 0.1$$

The last specification, $|T(i\omega)| \le \frac{1}{10}, \quad \omega \ge 2$, gives (it is assumed that

$\underline{\sigma}(L(i\omega)) \leq \overline{\sigma}(L(i\omega)) < 1$ in the considered frequency range)

$$\overline{\sigma}(T(i\omega)) = \overline{\sigma}\left((I + L(i\omega))^{-1}L(i\omega)\right) \leq \frac{1}{10}$$

$$\begin{aligned}\overline{\sigma}\left((I + L(i\omega))^{-1}L(i\omega)\right) &\leq \overline{\sigma}\left((I + L(i\omega))^{-1}\right)\overline{\sigma}(L(i\omega)) = \\ &= (\underline{\sigma}(I + L(i\omega)))^{-1}\,\overline{\sigma}(L(i\omega))\end{aligned}$$

$$\begin{aligned}\underline{\sigma}(I + L(i\omega)) \geq \underline{\sigma}(I) - \underline{\sigma}(L(i\omega)) = 1 - \underline{\sigma}(L(i\omega)) &\geq 1 - \overline{\sigma}(L(i\omega)) \\ \Rightarrow (\underline{\sigma}(I + L(i\omega)))^{-1} &\leq \frac{1}{1 - \overline{\sigma}(L(i\omega))}\end{aligned}$$

$$\begin{aligned}(\underline{\sigma}(I + L(i\omega)))^{-1}\,\overline{\sigma}(L(i\omega)) \leq \frac{\overline{\sigma}(L(i\omega))}{1 - \overline{\sigma}(L(i\omega))} &\leq \frac{1}{10} \\ \Rightarrow \overline{\sigma}(L(i\omega)) &\leq \frac{1}{11}\end{aligned}$$

The above specification should be valid when $\omega \geq 2$.

d. The minimal slope (roll-off rate) of the open-loop gain in the interval [0.1, 2] is approximately given by the straight line that intersects the edges (corners) of the "forbidden areas" in **c**.

$$\text{Bode diagram slope:} \quad \frac{\log(1/11) - \log 11}{\log 2 - \log 0.1} \approx -1.61$$

This gives

$$\frac{\log 1 - \log 11}{\log \omega_c - \log 0.1} = -1.61 \quad \Rightarrow \quad \omega_c = 0.44 \text{ rad/s}$$

From Bode's relations, we get the relation

$$\arg L \lessapprox -1.61 \cdot \frac{\pi}{2} = -145^\circ$$

which gives a phase margin of $\varphi_m \approx 35^\circ$.
Lower bound on $\|T\|_\infty$?

$$L(i\omega_c) = 1 \cdot e^{-i \cdot 145^\circ} = -0.818 + 0.575i$$

$$|T(i\omega_c)| = \left|\frac{L(i\omega_c)}{1 + L(i\omega_c)}\right| \approx 1.66$$

$$\|T\|_\infty = \sup_\omega |T(i\omega)| \quad \Rightarrow \quad \|T\|_\infty \geq |T(i\omega)|, \quad \forall\omega$$

$$\Rightarrow \quad \|T\|_\infty \geq 1.66$$

6.7 The specification

$$|S(i\omega)| \leq \frac{2\omega}{\sqrt{\omega^2 + c^2}} \qquad \omega \in \mathbb{R}$$

is equal to

$$\sup_{\omega} \left| \frac{\sqrt{\omega^2 + c^2}}{2\omega} S(i\omega) \right| \leq 1$$

Since

$$W_S(i\omega) = \frac{i\omega + c}{2i\omega}$$

gives

$$|W_S(i\omega)| = \frac{\sqrt{\omega^2 + c^2}}{2\omega}$$

the specification can be written

$$\sup_{\omega} |W_S(i\omega) S(i\omega)| \leq 1$$

According to Theorem 7.2, this specification is impossible to meet when the process has a RHP zero in $s = z$, unless $|W_S(z)| \leq 1$. Here we have a zero in $z = 6$, so we must have

$$\frac{6 + c}{12} \leq 1 \Rightarrow c \leq 6 = a$$

Solutions to Exercise 7. Controller Structures and Preparations for Laboratory Exercise 2

Note: Exercises 7.1-7.3 serve as preparation for Laboratory Excercise 2.

7.1 **a.** The relative gain array for a complex valued matrix is given by

$$\mathrm{RGA}\,(G) = G. * (G^{\dagger})^T$$

where † denotes the pseudo-inverse of G, and $.*$ denotes element wise multiplication. For a process $G(s)$ the RGA is usually computed for the DC-gain ($\omega = 0$) and the cut-off frequency ($\omega = \omega_c$). By inspecting the elements in the RGA-matrix, we can often decide what output should be controlled with what input. We should choose a pairing that gives the diagonal elements close to 1 and avoid pairings that give negative diagonal elements.

b.

$$\mathrm{RGA}(G(0)) = G(0) \,.*\, G^{-T}(0) = \begin{pmatrix} -\frac{5}{7} & \frac{12}{7} \\ \frac{12}{7} & -\frac{5}{7} \end{pmatrix}$$

c. Since we should avoid negative diagonal elements and keep the diagonal elements close to 1, we should choose the pairing $y_1 \leftrightarrow u_2$ and $y_2 \leftrightarrow u_1$.

7.2 We have

$$P(0) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.01 & 0.1 \\ 0.1 & 1 & 0 \end{pmatrix}$$

and

$$P(0)^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -0.1 & 0 & 1 \\ 0.01 & 10 & -0.1 \end{pmatrix}$$

$$\mathrm{RGA}(P(0)) = P(0) \,.*\, (P(0)^{-1})^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

The RGA suggests that we should control output 1 with input 1, output 2 with input 3, and output 3 with input 2.

7.3 **a.** We see from the flow equation

$$A_i \frac{d\Delta h_i}{dt} = -a_i \sqrt{\frac{g}{2h_i^0}}\, \Delta h_i + \Delta q_{in} \tag{7.1}$$

that the outflow from tank i is

$$q_{out} = a_i \sqrt{\frac{g}{2h_i^0}}\, \Delta h_i.$$

The inflows into the tanks are found as the sum of the outflow from the tank above and the flow from the pumps into the respective tanks. Writing down equation (7.1) for each of the four tanks now gives the dynamics.

Substituting the time constants T_i into the dynamics, and arranging them into matrix form then gives the state space form.

b. The transfer matrix is given by

$$P(s) = C(sI - A)^{-1}B =$$

$$= \begin{pmatrix} k_c & 0 & 0 & 0 \\ 0 & k_c & 0 & 0 \end{pmatrix} \begin{pmatrix} s+\frac{1}{T_1} & 0 & -\frac{A_3}{A_1T_3} & 0 \\ 0 & s+\frac{1}{T_2} & 0 & -\frac{A_4}{A_2T_4} \\ 0 & 0 & s+\frac{1}{T_3} & 0 \\ 0 & 0 & 0 & s+\frac{1}{T_4} \end{pmatrix}^{-1} \begin{pmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2)k_2}{A_3} \\ \frac{(1-\gamma_1)k_1}{A_4} & 0 \end{pmatrix}$$

$$= \begin{pmatrix} \frac{\gamma_1 c_1}{1+sT_1} & \frac{k_2}{k_1}\cdot\frac{(1-\gamma_2)c_1}{(1+sT_1)(1+sT_3)} \\ \frac{k_1}{k_2}\cdot\frac{(1-\gamma_1)c_2}{(1+sT_2)(1+sT_4)} & \frac{\gamma_2 c_2}{1+sT_2} \end{pmatrix}$$

c. The zeros are given by the equation

$$T_3T_4s^2 + (T_3+T_4)s + 1 - \frac{(1-\gamma_1)(1-\gamma_2)}{\gamma_1\gamma_2} = 0$$

The two first coefficients are always positive, since $T_3, T_4 > 0$. The last coefficient is positive (and both zeros are thus stable) iff

$$\frac{(1-\gamma_1)(1-\gamma_2)}{\gamma_1\gamma_2} < 1 \quad\Leftrightarrow\quad \gamma_1 + \gamma_2 > 1$$

In the case $\gamma_1 = \gamma_2 = 0.7$ we get a minimum-phase system which should be easier to control than the non-minimum-phase system we get in the case $\gamma_1 = \gamma_2 = 0.3$.

d. We have

$$P(0) = \begin{pmatrix} \gamma_1 c_1 & \frac{k_2}{k_1}(1-\gamma_2)c_1 \\ \frac{k_1}{k_2}(1-\gamma_1)c_2 & \gamma_2 c_2 \end{pmatrix}$$

and

$$P(0)^{-1} = \frac{1}{c_1c_2(\gamma_1+\gamma_2-1)} \begin{pmatrix} \gamma_2 c_2 & -\frac{k_2}{k_1}(1-\gamma_2)c_1 \\ -\frac{k_1}{k_2}(1-\gamma_1)c_2 & \gamma_1 c_1 \end{pmatrix}$$

$$\text{RGA}(P(0)) = P(0)\,.\!*\,(P(0)^{-1})^T =$$
$$= \frac{1}{c_1c_2(\gamma_1+\gamma_2-1)} \begin{pmatrix} \gamma_1c_1\gamma_2c_2 & -(1-\gamma_2)c_1(1-\gamma_1)c_2 \\ -(1-\gamma_2)c_1(1-\gamma_1)c_2 & \gamma_2c_2\gamma_1c_1 \end{pmatrix}$$
$$= \begin{pmatrix} \dfrac{\gamma_1\gamma_2}{\gamma_1+\gamma_2-1} & 1-\dfrac{\gamma_1\gamma_2}{\gamma_1+\gamma_2-1} \\ 1-\dfrac{\gamma_1\gamma_2}{\gamma_1+\gamma_2-1} & \dfrac{\gamma_1\gamma_2}{\gamma_1+\gamma_2-1} \end{pmatrix} = \begin{pmatrix} \lambda & 1-\lambda \\ 1-\lambda & \lambda \end{pmatrix}$$

In the case $\gamma_1 = \gamma_2 = 0.7$ we get

$$\text{RGA}(P(0)) = \begin{pmatrix} 1.225 & -0.225 \\ -0.225 & 1.225 \end{pmatrix}$$

The RGA suggests we should control output 1 with input 1 and output 2 with input 2.

In the case $\gamma_1 = \gamma_2 = 0.3$ we get

$$\text{RGA}(P(0)) = \begin{pmatrix} -0.225 & 1.225 \\ 1.225 & -0.225 \end{pmatrix}$$

The RGA suggests that in this case we should control output 1 with input 2 and output 2 with input 1.

7.4 **a.** We compute the RGA for $\omega = 0$ rad/s.

$$\text{RGA}(G(s)) = \begin{pmatrix} \frac{s-1}{s+1} & \frac{2}{s+1} \\ \frac{2}{s+1} & \frac{s-1}{s+1} \end{pmatrix}$$

gives

$$\text{RGA}(G(0)) = \begin{pmatrix} -1 & 2 \\ 2 & -1 \end{pmatrix}.$$

Since you should avoid pairing that gives negative diagonal elements we choose $y_1 \leftrightarrow u_2$ and $y_2 \leftrightarrow u_1$.

b. We have that

$$G(0) = \begin{pmatrix} 1 & -2 \\ 1 & -1 \end{pmatrix}$$

Using a decoupled controller structure with $W_1 = G^{-1}(0)$ and $W_2 = I$ we get a decoupled system in stationarity. (See Glad&Ljung ch. 8.3.) The controller is

$$F(s) = W_1F^{\text{diag}}(s)W_2 = \begin{pmatrix} -F_{11}(s) & 2F_{22}(s) \\ -F_{11}(s) & F_{22}(s) \end{pmatrix}.$$

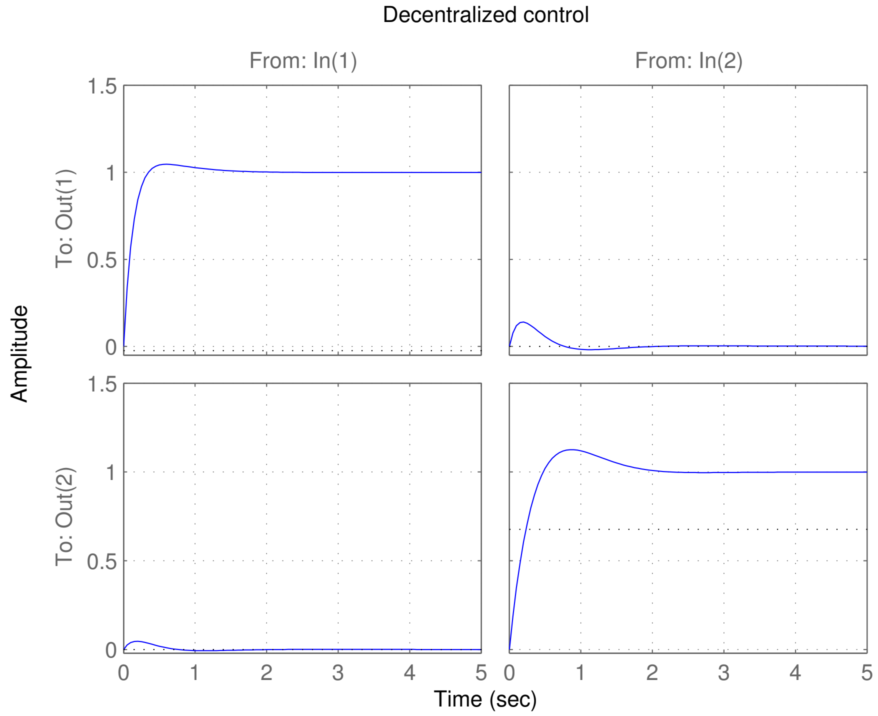


Figure 7.1 Decentralized control

7.5

1. Decentralized control. First we calculate the RGA of the process,

$$\mathrm{RGA}(G(0)) = G(0)\,.*\,G^{-T}(0) = \begin{pmatrix} 1.2308 & -0.2308 \\ -0.2308 & 1.2308 \end{pmatrix}.$$

We see that we should choose $y_1 \leftrightarrow u_1$ and $y_2 \leftrightarrow u_2$. A resonable tuning, either by pole placement or hand tuning, gives PI-controllers with parameters close to

$$F(s) = \begin{pmatrix} 2(1+\frac{1}{0.2s}) & 0 \\ 0 & 2(1+\frac{1}{0.2s}) \end{pmatrix}.$$

See figure 7.1 for step responses.

2. Decoupled control. The inverse of the static gain matrix is given by

$$G^{-1}(0) = \begin{pmatrix} 4 & 3 \\ 1 & 4 \end{pmatrix}^{-1}$$

Thus, for decoupling, we use $W_1 = G^{-1}(0)$ and $W_2 = I$. Hand-tuning of the PI-controllers gives

$$F(s) = \begin{pmatrix} 3(1+\frac{1}{0.2s}) & 0 \\ 0 & 6(1+\frac{1}{0.3s}) \end{pmatrix}.$$

See figure 7.2 for step responses.

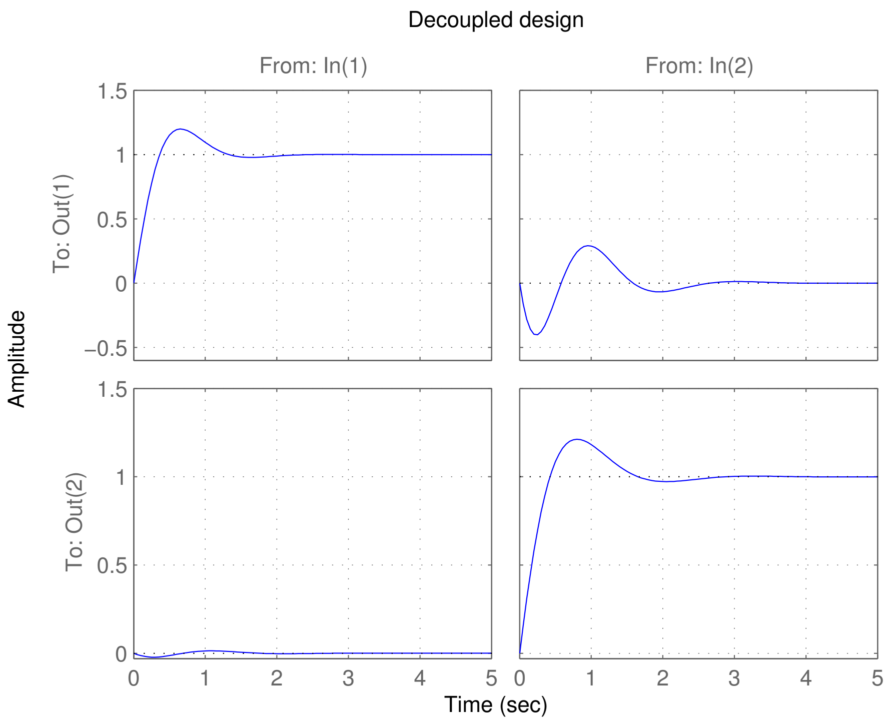


Figure 7.2 Decoupled control

```
close all
clear all
s = tf('s');

G = [4/(s+1) 3/(3*s+1); 1/(3*s+1) 2/(s+0.5)];

%Decentralized control
RGA = dcgain(G).*(inv(dcgain(G)))'
F = [2*(1+1/(0.5*s)) 0;0 2*(1+1/(0.5*s))];
figure(1)
step(feedback(G*F, eye(2)),5)
title('Decentralized control');grid

% Decoupled design
Go= dcgain(G)
F = [3*(1+1/(0.2*s)) 0;0 6*(1+1/(0.3*s))];
figure(2);
step(feedback(G*inv(Go)*F, eye(2)),5);
title('Decoupled design');grid
```

Solutions to Exercise 8. Linear Quadratic Optimal Control

8.1

a. The Riccati equation becomes ($A = a,\ B = 1,\ M = 1,\ Q_1 = 1,\ Q_2 = R$)

$$2Sa + 1 - SR^{-1}S = 0$$

This gives

$$S = aR + \sqrt{(aR)^2 + R}$$

($S = aR - \sqrt{(aR)^2 + R}$ is not a solution since S has to be positive definite.) Thus the optimal control is given by

$$L = \frac{S}{R} = a + \sqrt{a^2 + \frac{1}{R}}.$$

The closed loop system is hence, using $u(t) = -Lx(t) + L_r r(t)$

$$\dot{x}(t) = -\sqrt{a^2 + \frac{1}{R}}x(t) + L_r r(t)$$
$$y(t) = x(t)$$

L_r has to be chosen so that we get a stationary gain of 1 from the reference to the output, i.e. $G_{r\to y}(0) = C(-A + BL)^{-1}BL_r + D = 1$.
We get $L_r = (L - a) = \sqrt{a^2 + \frac{1}{R}}$.

b. See Matlab code below and Figure 8.1. Conclusion: Less weight on u gives a faster system since we are allowed to move the control signal more, and vice versa.

```
clear all
close all

A= 1;
B = 1;
C = 1;

P = ss(A, B, C, 0);

Q = 1;
Rvec=0.001:0.001:0.5;

for i=1:length(Rvec)
    R = Rvec(i);
    [L, S, E] = lqr(P,Q, R);
    Evec(i) = E;
end
```

```
plot(Rvec, Evec)
xlabel('Control signal weight')
ylabel('Closed loop pole')
grid
```

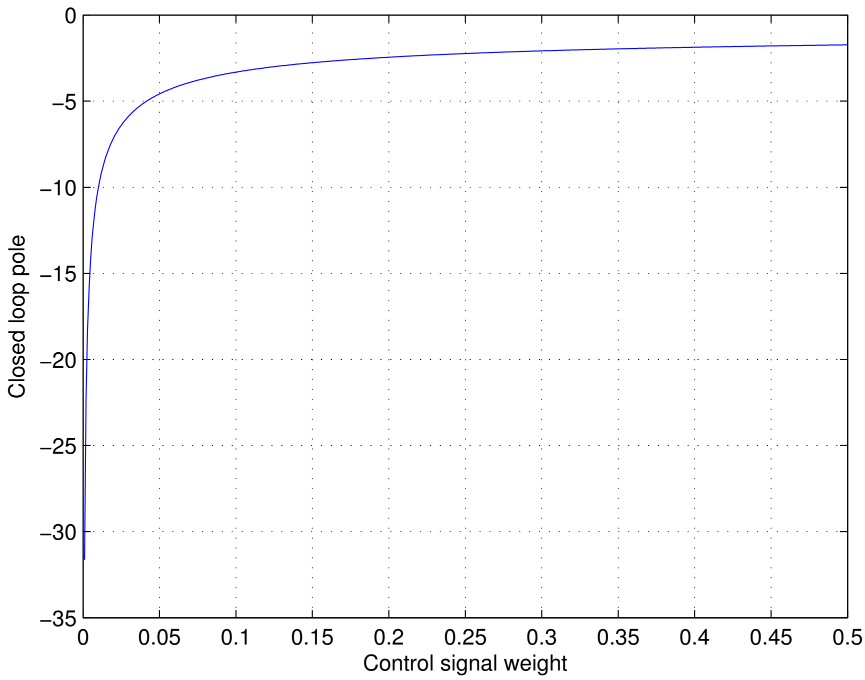


Figure 8.1 Control signal weight versus closed loop pole

8.2 See Figure 8.2 and Matlab code below

```
clear all
close all

A = [1 0;1 0];
B = [1 0]';
C = [1 1];


%Solving the Riccati equation
Q = C'*C;
R = 1;
S = zeros(2,1);
E = eye(2);

[X, K, G]=care(A, B, Q, R, S, E);
L1 = R\B'*X
eig(A-B*L1)

%using lqry
```

```
sys = ss(A, B, C, 0);
[L2, S, E]=lqry(sys, R,R)
eig(A-B*L2)

%simulate the system with initial conditions
sys = ss(A-B*L2, B, C, 0);
x0 = [1 1];
initial(sys,x0);grid
```

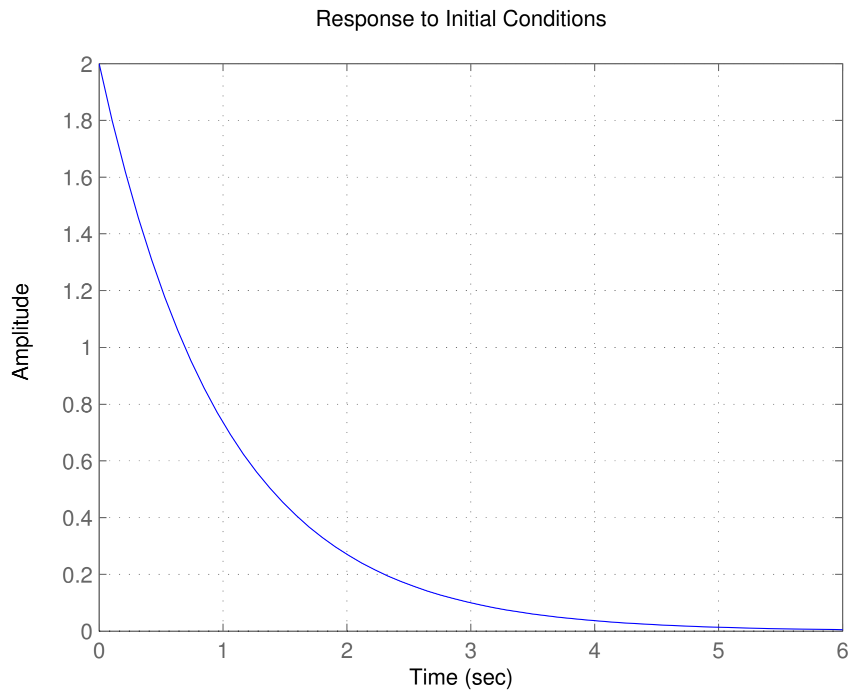


Figure 8.2 Response to initial conditions

8.3 The loop gain is

$$L(sI - A)^{-1}B = \frac{18}{(s-1)(s+2)}$$

Thus, the Nyquist curve will approach the origin with a phase of $-180°$. LQ-optimal loop gain always has an asympototic phase of $-90°$. Therefore, it can not be an LQ-optimal state feed back vector.

8.4 The system has two unstable poles in 2 and 3. If the cost function should be less than ∞ then the system must be stabilizable, i.e. all unstable poles must be controllable. The controllability matrix is given by

$$W_c = (BAB) = \begin{pmatrix} -4 & -12 \\ 8 & 24 \end{pmatrix}$$

which is a rank 1 matrix. Thus, both modes are not controllable, and hence, we can not make the cost function less than ∞.

8.5

a. The weighting matrices are $Q_1 = \text{diag}([1\ 2])$ and $Q_2 = 0.01,\ 10,\ 1000$.

b. See Figure 8.3 for step responses, and Matlab code below.

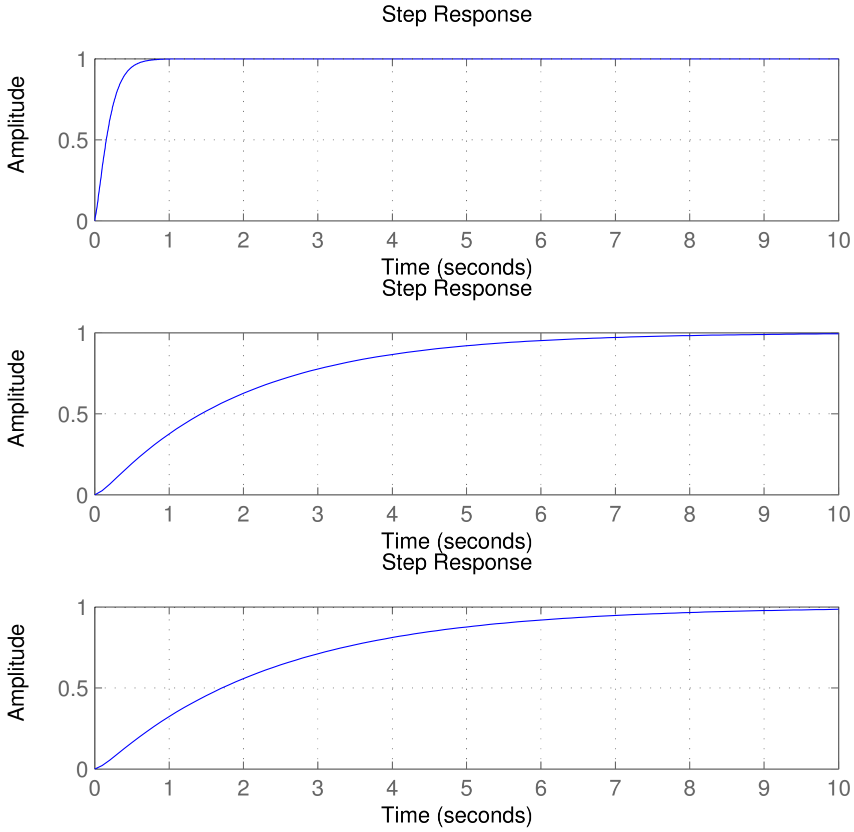


Figure 8.3 Step responses for different weight on control signal.

```
clear all
close all


A = [1 3;4 8];
B = [1 0.1]';
M = [0 1]
P = ss(A, B, M, 0);

Q1 = diag([1 2]);

%Calculate the LQ-controller, display the closed loop eigenvalues, and do
%stepresponses for the different values of Q2
figure(1)

Q2 = 0.01;
```

```
[L, S, E] = lqr(P,Q1,Q2);
Lr = 1/(M*inv(B*L-A)*B);
E
subplot(311)
step(ss(A-B*L, B*Lr, M, 0),10);grid

Q2 = 10;
[L, S, E] = LQR(P,Q1,Q2);
E
Lr = 1/(M*inv(B*L-A)*B);
subplot(312)
step(ss(A-B*L, B*Lr, M, 0),10);grid

Q2 = 1000;
[L, S, E] = LQR(P,Q1,Q2);
E
Lr = 1/(M*inv(B*L-A)*B);
subplot(313)
step(ss(A-B*L, B*Lr, M, 0),10);grid
```

8.6

a. Put

$$S = \begin{pmatrix} s_1 & s_2 \\ s_2 & s_3 \end{pmatrix}$$

and solve the Ricatti equation

$$Q_1 + A^T S + SA - SBQ_2^{-1}B^T S = 0.$$

This gives

$$\begin{pmatrix} 0 & 0 \\ s_1 & s_2 \end{pmatrix} + \begin{pmatrix} 0 & s_1 \\ 0 & s_2 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} - \frac{1}{0.1} \cdot \begin{pmatrix} s_2^2 & s_2 s_3 \\ s_2 s_3 & s_3^2 \end{pmatrix} = 0,$$

with the solution

$$\begin{aligned} s_1 &= \sqrt{2} \cdot 10^{-1/4}, \\ s_2 &= 10^{-1/2}, \\ s_3 &= \sqrt{2} \cdot 10^{-3/4}. \end{aligned}$$

The optimal controller is given by

$$L = Q_2^{-1} B^T S = (\sqrt{10} \quad \sqrt{2} \cdot 10^{1/4}).$$

To get $y = r$ in stationarity:

$$1 = G(0) = M(-A + BL)^{-1} B L_r \quad \Rightarrow L_r = \sqrt{10}.$$

b. Both x_1 and x_2 must be measured, e.g.

$$C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

c. 3. is the only case with a cost on the velocity x_2. This makes the controller try to avoid oscillations, so we get $3. - D)$, the only step response without overshoot. The weight, Q_2, on the control signal determines the speed of the system. A low weight on the control signal gives a faster system since we are allowed to use more control signal. This gives $1. - C)$, $2. - A)$, $4. - B)$.

8.7

a. Weighting matrices $Q_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ och $Q_2 = \eta$. The Riccati equation to be solved with respect to S is

$$A^T S + SA + Q_1 - SBQ_2^{-1}B^T S = 0$$

Put

$$S = \begin{pmatrix} s_1 & s_2 \\ s_2 & s_3 \end{pmatrix},$$

which gives

$$\begin{pmatrix} 0 & 0 \\ s_1 & s_2 \end{pmatrix} + \begin{pmatrix} 0 & s_1 \\ 0 & s_2 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} - \frac{1}{\eta} \cdot \begin{pmatrix} s_2^2 & s_2 s_3 \\ s_2 s_3 & s_3^2 \end{pmatrix} = 0$$

We see, by insertion, that

$$\begin{aligned} s_1 &= \sqrt{2} \cdot \eta^{1/4} \\ s_2 &= \eta^{1/2} \\ s_3 &= \sqrt{2} \cdot \eta^{3/4} \end{aligned}$$

solves the Riccati equation.

b. The optimal state feed back is

$$\begin{aligned} L = Q_2^{-1} B^T S &= \frac{1}{\eta} \cdot (0 \quad 1) \begin{pmatrix} \sqrt{2}\eta^{1/4} & \eta^{1/2} \\ \eta^{1/2} & \sqrt{2} \cdot \eta^{-3/4} \end{pmatrix} \\ &= \frac{1}{\eta} \cdot (\eta^{1/2} \quad \sqrt{2}\eta^{3/4}) = (\eta^{-1/2} \quad \sqrt{2} \cdot \eta^{-1/4}) \end{aligned}$$

The poles are the eigenvalues to $A - BL$. Put $\mu = \eta^{-1/4} \Rightarrow L = (\mu^2 \quad \sqrt{2} \cdot \mu)$. This gives

$$0 = \det \begin{pmatrix} s & -1 \\ \mu^2 & s + \sqrt{2} \cdot \mu \end{pmatrix} = s^2 + \sqrt{2}\mu s + \mu^2,$$

that is

$$\begin{aligned} s &= -\frac{\mu}{\sqrt{2}} \pm \sqrt{\frac{\mu^2}{2} - \mu^2} = -\frac{\mu}{\sqrt{2}} \pm i \cdot \frac{\mu}{\sqrt{2}} = \\ &= -\frac{\mu}{\sqrt{2}} \cdot (1 \pm i) = -\frac{1}{\sqrt{2} \cdot \eta^{1/4}} \cdot (1 \pm i) \end{aligned}$$

If η is reduced, the distance between the poles and the origin will increase. This means that $u(t)$ will increase. Check the criterion!

Solutions to Exercise 9. Kalman Filtering/LQG

9.1 **a.** We have that $A = B = C = N = M = 1$. The Riccati equation thus reduces to

$$2P + R_1 - \frac{P^2}{R_2} = 0,$$

which has the positive semi-definite solution $P = R_2 + R_2\sqrt{1 + \frac{R1}{R2}}$. Thus, the Kalman filter gain is

$$K = \frac{1}{R_2}P = 1 + \sqrt{1 + \frac{R_1}{R_2}} = 1 + \sqrt{1+\beta}.$$

b. The Kalman filter dynamics are given by

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K(y(t) - C\hat{x}(t))$$

where $y(t) = Cx(t) + v_2(t)$. Using the values $A = B = C = N = M = 1$ we have the error dynamics

$$\dot{e}(t) = (A - KC)e(t) - Kv_2(t) + v_1(t) = -\sqrt{1+\beta}e(t) - (1 + \sqrt{1+\beta})v_2(t) + v_1(t)$$

c. The Kalman filter pole depends on the ratio as $-\sqrt{1+\beta}$. We can see that if $\beta \to \infty$, the pole of the Kalmanfilter $\to -\infty$. Hence, the estimation error dynamics are fast, we believe very much in our measurents. On the other hand, if $\beta \to 0$, the Kalman filter pole tends to -1, that is, as fast as the process pole. Now, we trust the model more than the measurements.

9.2 See Matlab code below.

```
clear all
close all

A = [0 1;1 0];
C = [1 0];
N = [1 1]';
```

a.

```
%Using care
Q = N*N';
R = 1;
S = zeros(2,1);
E = eye(2);

[X, L, T]=care(A', C', Q, R, S, E);

K1 = X*C'

eig(A-K1*C)
```

b.
```
%Using lqe
[K2, P, E]=lqe(A,N, C, 1,1,0)
eig(A-K2*C)
```

9.3 **a.** The noise model has the following state space realization,

$$\begin{aligned}\dot{x}_w(t) &= -\,\delta x_w(t) + n(t)\\ w(t) &= x_w(t)\end{aligned}$$

Extending the state space model of the process with the noise model gives,

$$\begin{aligned}\dot{x}_e(t) &= \begin{pmatrix} -1 & 0 \\ 0 & -\delta \end{pmatrix} x_e(t) + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} v_{1e}(t)\\ y(t) &= \begin{pmatrix} 1 & 1 \end{pmatrix} x_e(t) + v_2(t)\\ z(t) &= \begin{pmatrix} 1 & 1 \end{pmatrix} x_e(t)\end{aligned}$$

Note here that $z(t)$ contains the noise state $x_w(t)$, so that, if we design an LQG controller we will try to minimize the disturbance state effect also.

b. See Matlab code in (**d**). Why do we need a small weight on $u(t)$? Since integral action requires the control signal gain to be large at low frequencies we have to let the control signal be large, otherwise the low frequency gain will be limited independent of noise model.

c. If we change the cut-off frequency of the noise filter, we change the cut-off frequency of the low frequency gain of the controller, this is shown in Figure 9.1.

If we on the other hand change the noise intensity, we indirectly change the gain of the noise filter. Hence, we will increase the controller gain for all frequencies, see Figure 9.2.

d. Response to constant load disturbances will *always* have a static error since we do not have infinite gain at low frequencies. That is, we do not have pure integral action, only approximative.

See below for Matlab code,

```
clear all
close all

B = [1; 0];
C = [1 1];
D = 0;
N = eye(2);
H = [0 0];


%Different values of cut-off frequency of noise filter

%Note that the values in the diagonal of the disturbance
%filter inputintensity matrix [1 0; 0 100] are arbitrary;
%their relation will later be varied
```

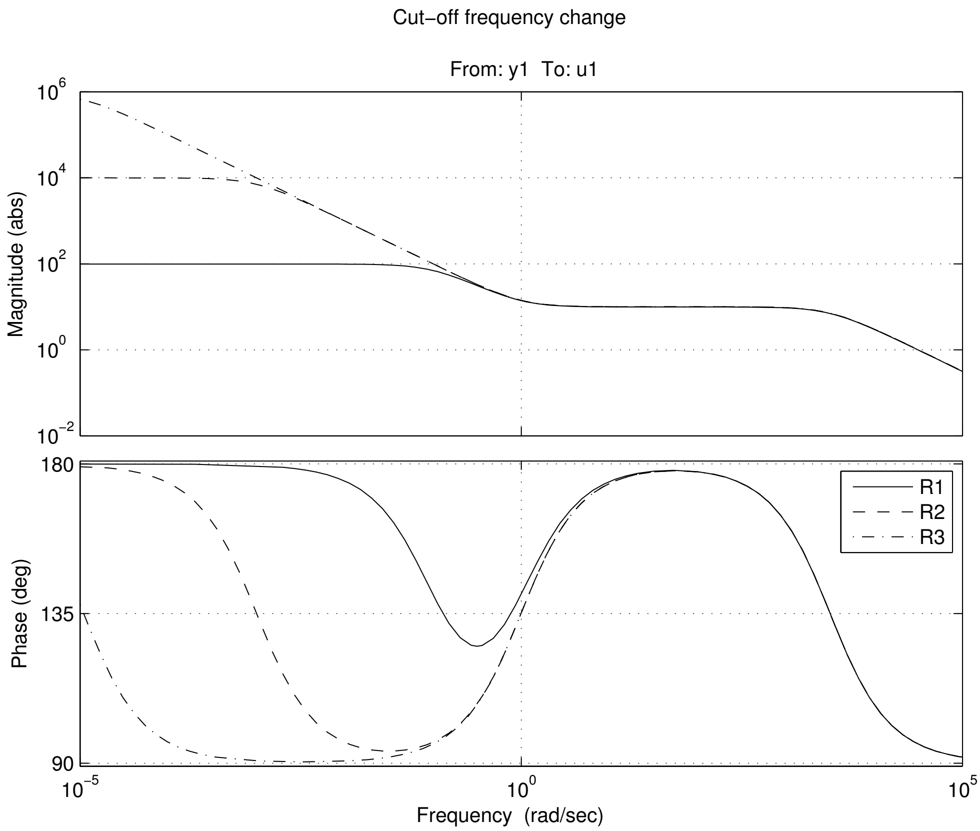


Figure 9.1 Change of cut-off frequency

```
A = [-1 0;0 -0.1];
sys = ss(A, [B N], C, [D H]);
[Kest, L, P]=kalman(sys, [1 0; 0 100], 1);
P = ss(A, B, C, D);
[K,S,E] = lqry(P,1,0.0000001,0);
R1 = lqgreg(Kest,K);

A = [-1 0;0 -0.001];
sys = ss(A, [B N], C, [D H]);
[Kest, L, P]=kalman(sys, [1 0; 0 100], 1);
P = ss(A, B, C, D);
[K,S,E] = lqry(P,1,0.0000001,0);
R2 = lqgreg(Kest,K);


A = [-1 0;0 -0.00001];
sys = ss(A, [B N], C, [D H]);
[Kest, L, P]=kalman(sys, [1 0; 0 100], 1);
P = ss(A, B, C, D);
[K,S,E] = lqry(P,1,0.0000001,0);
R3 = lqgreg(Kest,K);
figure(1)
```

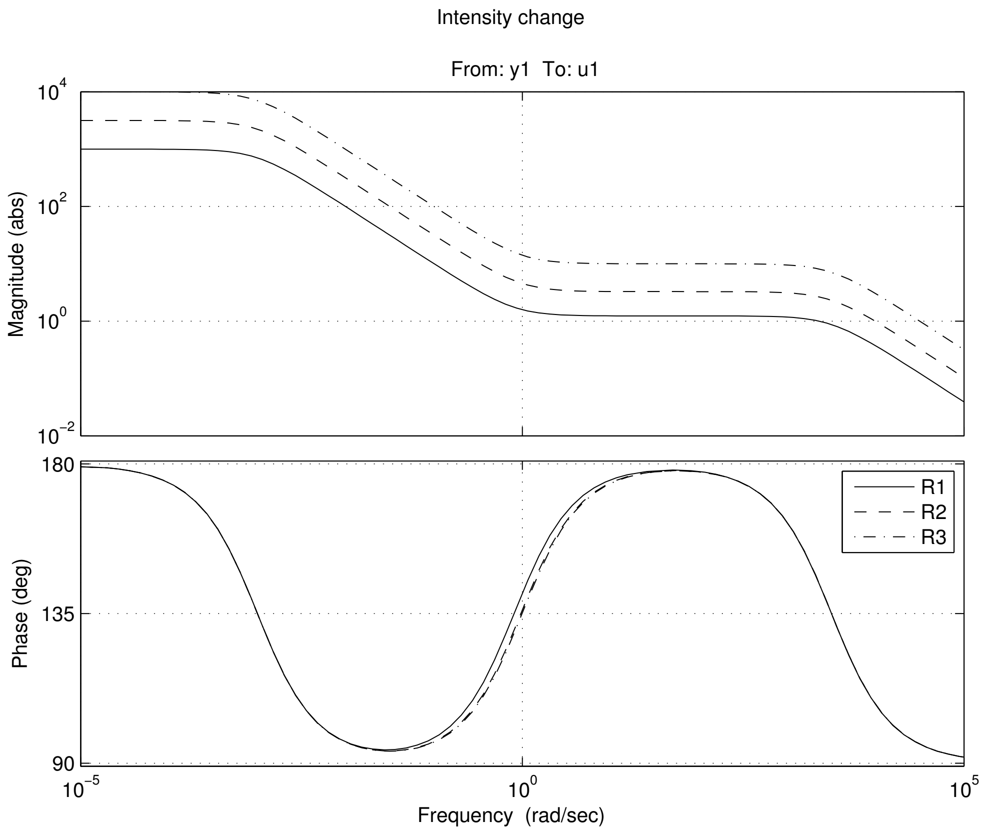


Figure 9.2 Change of noise intensity

```
bode(R1, R2,'--', R3,'-.');grid
legend('R1', 'R2', 'R3')
title('Cut-off frequency change')

%Different values of the disturbance filter input intensity

A = [-1 0;0 -0.001];
sys = ss(A, [B N], C, [D H]);
[Kest, L, P]=kalman(sys, [1 0; 0 1], 1);
P = ss(A, B, C, D);
[K,S,E] = lqry(P,1,0.0000001,0);
R1 = lqgreg(Kest,K);

A = [-1 0;0 -0.001];
sys = ss(A, [B N], C, [D H]);
[Kest, L, P]=kalman(sys, [1 0; 0 10], 1);
P = ss(A, B, C, D);
[K,S,E] = lqry(P,1,0.0000001,0);
R2 = lqgreg(Kest,K);

A = [-1 0;0 -0.001];
sys = ss(A, [B N], C, [D H]);
[Kest, L, P]=kalman(sys, [1 0; 0 100], 1);
```

```
P = ss(A, B, C, D);
[K,S,E] = lqry(P,1,0.0000001,0);
R3 = lqgreg(Kest,K);

figure(2)
bode(R1, R2,'--', R3,'-.');grid
legend('R1', 'R2', 'R3')
title('Intensity change')
```

9.4 a. To get a small complementary sensitivity at the oscillation frequency, we need the LQG controller to have a low gain at this frequency; effectively ignoring corresponding oscillations in the output y. This can be achieved by modelling the influence of the oscillatory system as a disturbance w on y according to

$$\begin{aligned} \dot{x} &= Ax + Bu + Nv_1 \\ y &= Cx + w + v_2 \end{aligned}$$

To model the oscillatory characteristics of w, we can consider w to be generated by passing white noise n through a second-order filter with a resonance peak at $\omega_0 = 0.5$ rad/s and a zero at $s = 0$, with transfer function

$$H(s) = \frac{K_v s}{s^2 + 2\zeta\omega_0 s + \omega_0^2}.$$

The zero at $s = 0$ is placed there to avoid an increased gain at low frequencies, which would otherwise follow. It is not necessary unless it is important to avoid this phenomenon and the exercise can be solved without it, which will then yield a slightly different solution to the one below.

The parameter ζ determines the magnitude of the resonance peak, and we can choose e.g. $\zeta = 0.02$.

On state-space form, the filter is given by

$$\begin{aligned} \dot{x}_v(t) &= \begin{pmatrix} -0.02 & -0.25 \\ 1 & 0 \end{pmatrix} x_v(t) + \begin{pmatrix} 1 \\ 0 \end{pmatrix} n(t) \\ w(t) &= \begin{pmatrix} K_v & 0 \end{pmatrix} x_v(t) \end{aligned}$$

Extend the original state space form with the noise model

$$\begin{aligned} \dot{x}(t) &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -0.02 & -0.25 \\ 0 & 0 & 1 & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} 0.1 & 0 \\ 0.1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} v_1(t) \\ n(t) \end{pmatrix} \\ y(t) &= \begin{pmatrix} 1 & 0 & K_v & 0 \end{pmatrix} x(t) + v_2(t) \\ z(t) &= \begin{pmatrix} 1 & 0 & 0 & 0 \end{pmatrix} x(t) \end{aligned}$$

If this model is used to compute K in the Kalman filter, for an appropriate value of K_v, we get supression of the resonance frequency. The intensity of

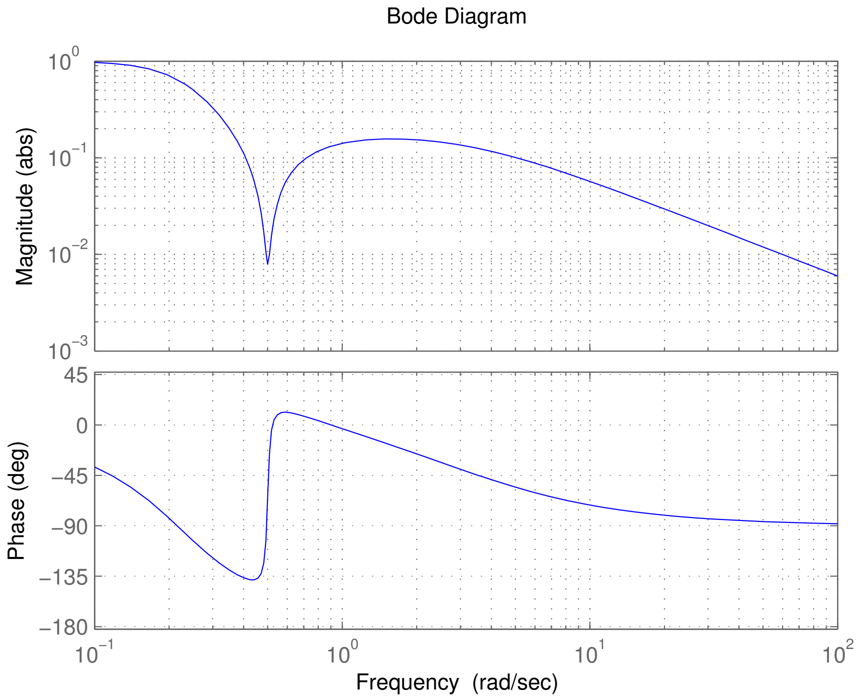


Figure 9.3 Attenuation of oscillative disturbance

the added noise input can e.g. be set to 1 since we can control the amplitude of the disturbance by changing K_v. Thus, we have the intensity matrices $R_1 = \text{diag}(1,\ 1)$, $R_2 = 0.1$.

Note that $z(t)$ do not depend on the x_v-states, i.e., if we are about to design an LQG controller, we have no weight on the added noise. The added noise is only used for specifying at what frequencies our measurements are uncertain.

b. See figure 9.3 for the Bode plot of the transfer function from measurement $y(t)$ to estimated output $\hat{y}(t)$ using $K_v = 1$. We see a large attenuation of frequencies at $\omega = 0.5$ rad/s.

Matlab code

```
clear all
close all
s = tf('s');

A = [0 1 0 0;0 -1 0 0;0 0 -0.02 -0.2501;0 0 1 0];
B = [0 1 0 0]'
C = [1 0 1 0];
N = [0.1 0; 0.1 0; 0 1; 0 0];

[K, P, E] = lqe(A,N,C,blkdiag(1,1),0.1);

Cnom = [1 0 0 0];
tf(ss(A-K*C, K, Cnom, 0))
```

```
bode(ss(A-K*C, K, Cnom, 0),{0.1, 100})
grid
```

9.5 a. We have the state-space representation

$$\dot{x}(t) = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v_1(t)$$
$$y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} + v_2(t)$$

(If a different state-space representation is chosen, the solution will look different although the steps will be similar.)

The Riccati-equation

$$AP + PA^T + NR_1N^T - PC^TR_2^{-1}CP = 0$$

is solved by letting $P = \begin{pmatrix} p_1 & p_2 \\ p_2 & p_3 \end{pmatrix}$. The equations become,

$$2p_2 - p_1^2 = 0$$
$$p_3 - p_1p_2 = 0$$
$$1 - p_2^2 = 0$$

The solution is thus

$$P = \begin{pmatrix} \sqrt{2} & 1 \\ 1 & \sqrt{2} \end{pmatrix}$$

with the optimal gain

$$K = PC^T = \begin{pmatrix} \sqrt{2} & 1 \end{pmatrix}^T$$

b. The poles of the Kalman filter are the eigenvalues of $A - KC$,

$$A - KC = \begin{pmatrix} -\sqrt{2} & 1 \\ -1 & 0 \end{pmatrix}$$

with the eigenvalues $\lambda_j = \dfrac{1}{\sqrt{2}}(-1 \pm i)$.

Solutions to Exercise 10. LQG and Preparations for Laboratory Exercise 3

10.1

a. See **c**.

b. Using the state vector $x_e = \begin{pmatrix} x^T & \hat{x}^T \end{pmatrix}^T$ and the obvious notation A, B, C, we get the system

$$\begin{aligned} \dot{x}_e &= \begin{pmatrix} A & -BL \\ KC & A - BL - KC \end{pmatrix} x_e + \begin{pmatrix} I \\ 0 \end{pmatrix} v_1 + \begin{pmatrix} 0 \\ K \end{pmatrix} v_2 \\ z &= \begin{pmatrix} C & 0 \end{pmatrix} x_e \end{aligned}$$

c. With less measurement noise the estimated states converge faster to the actual states, and the output z converge faster to zero. See Figures 10.2-10.1 and Matlab code below.

As shown in exercise 9.1, only the relation between process noise and measurement noise matters. More process noise will therefore have the same effect as less measurement noise.

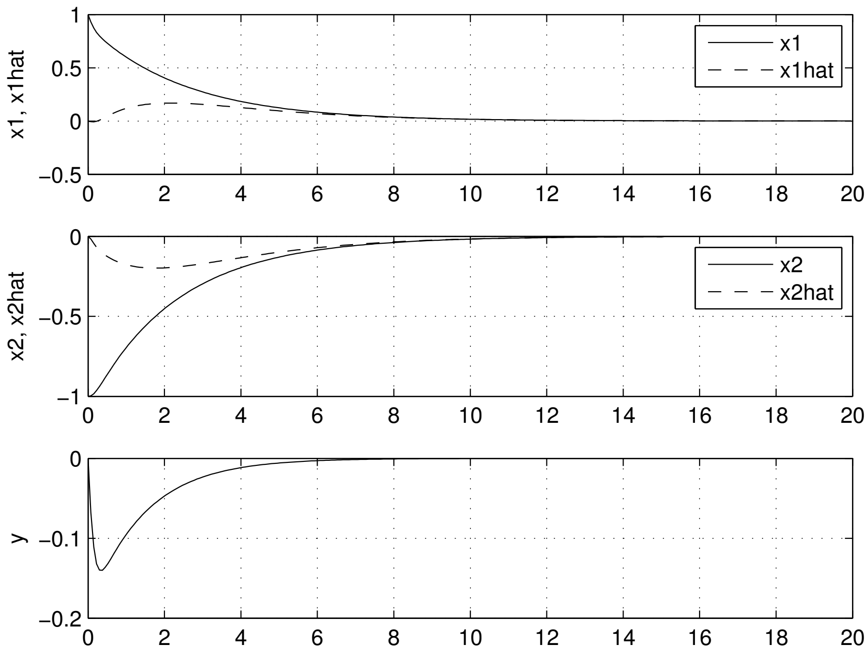


Figure 10.1 Initial response if little measurement noise.

```
clear all
close all
clc
A = [0 1;0 0];
```

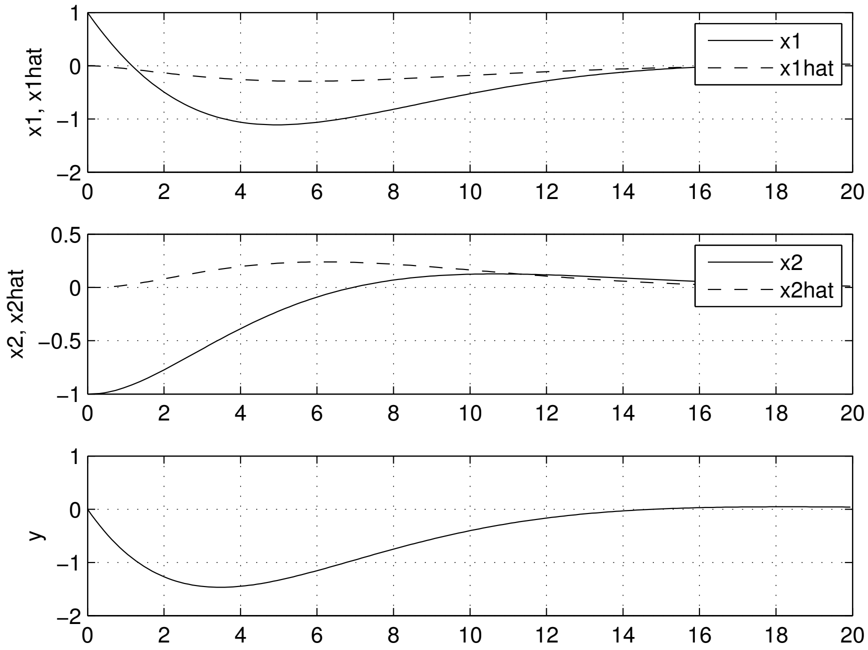


Figure 10.2 Initial response if much measurement noise.

```
B = [1 6;0 4];
C = [1 1];
D = zeros(1,2);

%State feed-back design
process = ss(A, B, C, D);
Q1 = 1;
Q2 = eye(2);
[L, S, E] = lqry(process,Q1,Q2);

%Kalman filter design
G = eye(2);
H =zeros(1,2);
syskalman = ss(A, [B G], C, [D H]);
R1 = eye(2);
R2 = 1;
[Kest, K, E] = kalman(syskalman,R1,R2);

%Construct closed loop
reg = lqgreg(Kest, L);
closed_loop = feedback(process, -reg);

%Plot response
[Y, T, X] = initial(closed_loop, [1 -1 0 0],0:0.01:20);
figure(1)
subplot(311)
plot(T, X(:,1));hold on;plot(T, X(:,3),'--');grid
```

```
legend('x1','x1hat');ylabel('x1, x1hat')
subplot(312)
plot(T, X(:,2));hold on;plot(T, X(:,4),'--');grid
legend('x2','x2hat');ylabel('x2, x2hat')
subplot(313)
plot(T,Y); grid; ylabel('y');
```

10.2 First of all, we see that we can not measure the states we want to control, so we need a Kalman filter. We start by setting up the problem in the standard form

$$\begin{aligned} \dot{x} &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u + v_1 \\ y &= \begin{pmatrix} 1 & 0 \end{pmatrix} x + v_2 \\ z &= x \end{aligned}$$

where v_2 is white noise with intensity 1. The cost function is

$$J = \int_0^\infty \left(z^T Q_1 z + u Q_2 u \right) dt$$

with $Q_1 = I_2$ and $Q_2 = 1$.
For the state feed-back gain, we have to solve the Riccati equation

$$A^T S + SA + Q_1 - SBQ_2^{-1} B^T S = 0$$

This gives the following equations,

$$\begin{aligned} 1 - s_2^2 &= 0 \\ s_1 - s_2 s_3 &= 0 \\ 2s_2 + 1 - s_3^2 &= 0 \end{aligned}$$

with the solution $s_1 = s_3 = \sqrt{3}$, $s_2 = 1$. This gives the state feed-back vector $L = B^T S = \begin{pmatrix} 1 & \sqrt{3} \end{pmatrix}$.
For the Kalman filter we must solve the Riccati equation

$$AP + PA^T + R_1 - PC^T CP^T = 0$$

with $R_1 = I_2$, which gives

$$\begin{aligned} 2p_2 + 1 - p_1^2 &= 0 \\ p_3 - p_1 p_2 &= 0 \\ 1 - p_2^2 &= 0 \end{aligned}$$

Using the solution for S we have that $p_1 = p_3 = \sqrt{3}$ and $p_2 = 1$ and $K = PC^T = \begin{pmatrix} \sqrt{3} \\ 1 \end{pmatrix}$
The controller is given by

$$\begin{aligned}\dot{\hat{x}} &= (A - BL - KC)\hat{x} + Ky \\ u &= -L\hat{x}\end{aligned}$$

and we have that

$$A - BL - KC = \begin{pmatrix} -\sqrt{3} & 1 \\ -2 & -\sqrt{3} \end{pmatrix}$$

10.3 No solution provided.

Solutions to Exercise 11. Youla Parametrization and Dead Time Compensation

11.1 a. We can divide P even further into smaller parts such that

$$P_{zw} = \begin{pmatrix} P_{xd} & P_{xn} \\ P_{vd} & P_{vn} \end{pmatrix}, \quad P_{zu} = \begin{pmatrix} P_{xu} \\ P_{vu} \end{pmatrix}, \quad P_{yw} = \begin{pmatrix} P_{yd} & P_{yn} \end{pmatrix}$$

Looking at the block diagram of the closed-loop system, we see that

$$\begin{aligned} &P_{xd} = P_0, \quad P_{xn} = 0, \quad P_{vd} = 1, \quad P_{vn} = 0 \\ &P_{xu} = P_0, \quad P_{vu} = 1 \\ &P_{yd} = P_0, \quad P_{yn} = 1 \\ &P_{yu} = P_0 \end{aligned}$$

Note that you have to determine the open-loop transfer functions, as if $C = 0$. The results gives us the following transfer matrix P:

$$P = \begin{pmatrix} P_0 & 0 & P_0 \\ 1 & 0 & 1 \\ P_0 & 1 & P_0 \end{pmatrix},$$

where

$$P_{zw} = \begin{pmatrix} P_0 & 0 \\ 1 & 0 \end{pmatrix}, \quad P_{zu} = \begin{pmatrix} P_0 \\ 1 \end{pmatrix}, \quad P_{yw} = \begin{pmatrix} P_0 & 1 \end{pmatrix}, \quad P_{yu} = P_0$$

b. Using the formula, we get

$$\begin{aligned} H &= P_{zw} + P_{zu}C(1 - P_{yu}C)^{-1}P_{yw} \\ &= \begin{pmatrix} P_0 & 0 \\ 1 & 0 \end{pmatrix} + \begin{pmatrix} P_0 \\ 1 \end{pmatrix} C(1 - P_0 C)^{-1} \begin{pmatrix} P_0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} P_0 & 0 \\ 1 & 0 \end{pmatrix} + \frac{C}{1 - P_0 C} \begin{pmatrix} P_0^2 & P_0 \\ P_0 & 1 \end{pmatrix} \\ &= \frac{1}{1 - P_0 C} \begin{pmatrix} (P_0 - P_0^2 C) + P_0^2 C & P_0 C \\ (1 - P_0 C) + P_0 C & C \end{pmatrix} = \begin{pmatrix} P_0 S & -T \\ S & CS \end{pmatrix} \end{aligned}$$

This means that the closed loop transfer function H consists of the gang of four. Note that

$$T = 1 - S = -\frac{P_0 C}{1 - P_0 C}$$

in this case where we have no explicit minus sign in the feedback loop.

c. Go back to the formula $H = P_{zw} + P_{zu}C(1 - P_{yu}C)^{-1}P_{yw}$, but replace $C(1 - P_{yu}C)^{-1}$ with Q. This gives

$$H = P_{zw} + P_{zu}QP_{yw} = \begin{pmatrix} P_0 & 0 \\ 1 & 0 \end{pmatrix} + Q \begin{pmatrix} P_0^2 & P_0 \\ P_0 & 1 \end{pmatrix} = \begin{pmatrix} P_0 + P_0^2 Q & P_0 Q \\ 1 + P_0 Q & Q \end{pmatrix}$$

and each element of H is linear in Q.

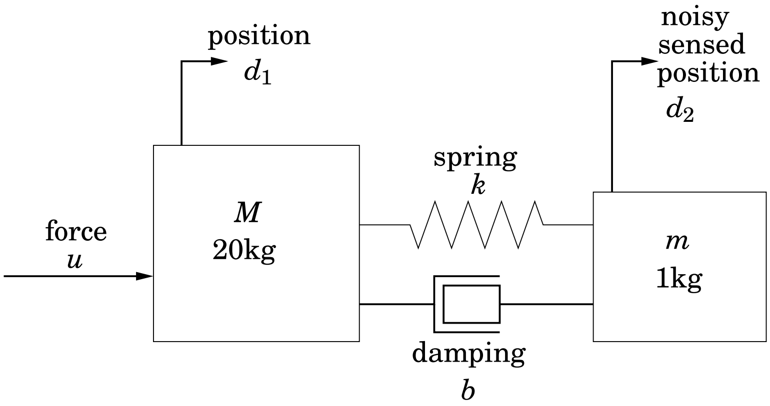


Figure 11.1 Mass spring system in Exercise 11.2.

d. Rearranging the Q formula a bit, we get

$$Q = C(1-P_0C)^{-1} \Rightarrow Q - QP_0C = C \Rightarrow (1+QP_0)C = Q, \Rightarrow C = (1+QP_0)^{-1}Q$$

where the last equality gives us the same controller as (8.18) in Glad&Ljung with the only difference that the feedback sign is a part of the controller here. P_0 is assumed to be the model we have of the system.

11.2 a. From the equation for the plant

$$\begin{aligned} \dot{x} &= Ax + Bu \\ d_1 &= C_1x \\ d_2 &= C_2x \end{aligned}$$

and the block diagram of the closed-loop system, we can see that

$$\begin{aligned} \dot{x} &= Ax + B(u+u_i) = Ax + \begin{pmatrix} 0 & 0 & B \end{pmatrix} \begin{pmatrix} r \\ n \\ u_i \end{pmatrix} + Bu \\ &= Ax + B_w w + Bu \\ z &= \begin{pmatrix} d_1 \\ u_o \\ e \end{pmatrix} = \begin{pmatrix} C_1x \\ u+u_i \\ r-d_1 \end{pmatrix} = \begin{pmatrix} C_1x \\ u+u_i \\ r-C_1x \end{pmatrix} \\ &= \begin{pmatrix} C_1 \\ 0 \\ -C_1 \end{pmatrix} x + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} r \\ n \\ u_i \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} u \\ &= C_z x + D_{zw} w + D_{zu} u \\ y &= \begin{pmatrix} r \\ d_2 + n \end{pmatrix} = \begin{pmatrix} r \\ C_2x + n \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ C_2 \end{pmatrix} x + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} r \\ n \\ u_i \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} u \\ &= C_y x + D_{yw} w + D_{yu} u \end{aligned}$$

b. The constraint on the maximum control signal, $\|u(t)\| \leq u_{max}$, will correspond to the closed loop transfer matrix $H_{u_o r}$, with index $(2,1)$. In problem 11.1 we saw that the transfer function $H_{u_o u_i}$ will correspond to the sensitivity function S. The M_s constraint will therefore correspond to the index $(2,3)$. The objective function will be related to two indices, namely those associated with H_{er} and $H_{u_o r}$, $(3,1)$ and $(2,1)$.

c. We have the formula $H = P_{zw} + P_{zu} Q P_{yw}$. Since P_{zu} is a 3×1 system and P_{yw} is a 2×3 system, Q must be 1×2. Therefore we have that $Q = [Q_1 \quad Q_2]$.

11.3 The system is non-minimum phase. There are many ways to choose the Q filter for IMC, but we have to respect some fundamental limitations. Here we will use a simple choice of Q. We try to cancel the process dynamics with $Q(s)$, but use the stable counterpart $6+3s$ of the zero instead. We also need to add a pole to $Q(s)$ to make it proper, which we place in $s = \lambda^{-1}$. We get

$$Q(s) = \frac{s^2 + 5s + 6}{(6+3s)(\lambda s + 1)}.$$

The controller becomes

$$C(s) = \frac{s^2 + 5s + 6}{s(3\lambda s + 6(\lambda + 1))},$$

which can be rewritten as

$$C(s) = \frac{5}{6(1+\lambda)} \left(1 + \frac{6}{5s} + \frac{s}{5}\right) \frac{1}{\frac{3\lambda}{6(\lambda+1)} s + 1}.$$

This corresponds to a PID controller in series with a lowpass filter.

11.4 As before, there are many ways to apply IMC. Here we try the two approaches to deal with time delays described in Glad&Ljung section 8.3.

1. Choose to ignore the time delay when the $Q(s)$ transfer function is calculated, but not when $F_y(s)$ is calculated.
   Thus, choosing $\lambda = 3$ we get

$$Q(s) = \frac{(P(s)e^{4s})^{-1}}{\lambda s + 1} = \frac{s+1}{3s+1}$$

   Hence, the controller is given by

$$F_y(s) = \frac{Q(s)}{1 - Q(s)P(s)} = \frac{s+1}{3s + 1 - e^{-4s}}$$

2. Approximate the time delay with a first order Padé approximation,

$$G(s) \approx \frac{1}{s+1} \frac{1-2s}{1+2s}.$$

   When we calculate the $Q(s)$-transfer function, we exclude $1-2s$. Thus, we now have, using $\lambda = 3$

$$Q(s) = \frac{(s+1)(2s+1)}{(3s+1)^2}.$$

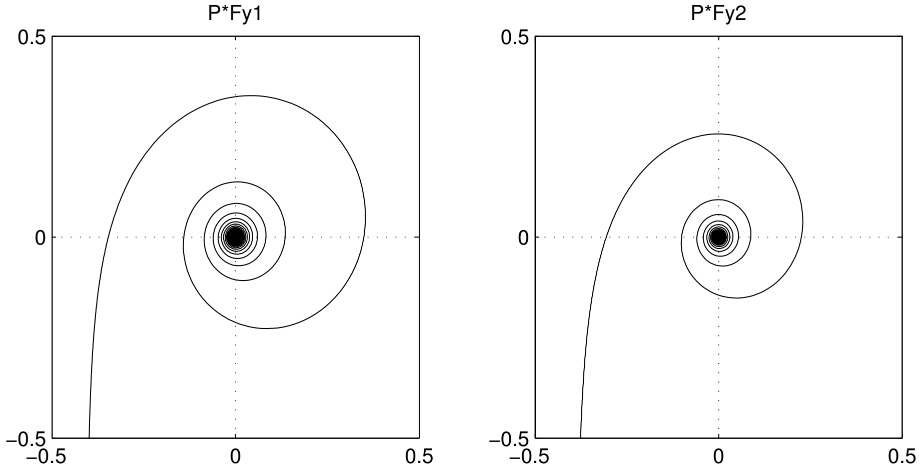


Figure 11.2 Nyquist plots of the loop transfer functions in Problem 11.4. The left plot shows the first alternative and the right plot shows the second.

Hence we have the controller

$$F_y(s) = \frac{Q(s)}{1 - Q(s)P(s)} = \frac{(s+1)(2s+1)}{(3s+1)^2 - (1+2s)e^{-4s}}$$

The Nyquist plots can be generated in Matlab, using the following lines of code

```
>> s = tf('s');
>> w = logspace(-2,2,1000);
>> P = 1./(1+i*w).*exp(-4*i*w);
>> Fy1 = (i*w+1)./(3*i*w+1-exp(-4*i*w));
>> Fy2 = (i*w+1).*(1+2i*w)./((3*i*w+1).*(3*i*w+1)-(1+2*i*w).*exp(-4*i*w));
>> figure
>> plot(P.*Fy1)
>> grid
>> figure
>> plot(P.*Fy2)
>> grid
```

From the plots (Figure 11.2) we see that neither encircles -1 and the closed loop systems are stable in both cases.

Solutions to Exercise 12. Synthesis by Convex Optimization

12.1

a.
```
Cz = [0 1 0 0;
      0 0 0 0;
      0 -1 0 0];
Cy = [0 0 0 0;
      0 0 0 1];

Dzw = [0 0 0;
       0 0 1;
       1 0 0];
Dzu = [0;
       1;
       0];
Dyw = [1 0 0;
       0 1 0];
Dyu = zeros(2,1);
Bw = zeros(4,3);
Bw(:,3) = B;
...
n_q1 = 10;
n_q2 = 10;
...
os = 1.02;
t1 = 3.6;
t2 = 19;
p = 0.8;
...
umax = 6;
...
M_s = 1.4;
...
weight_e = 0.5;
weight_u = 1.0;
```

b. The green curves in the plots correspond to the nominal controller. See the plots in Figures 12.1-12.3 and read the figure texts. From the plots we can also see that the only inactive constraint is the one posed on the control signal. Also see figure 12.4 for a plot of the open loop Nyquist curve together with the M_s-circle.

c. The bode diagram is displayed in Figure 12.5. The shape of the magnitude plot is very similar to that of a PD controller with a first order filter. Since the D-part of a PID controller acts to damp out oscillations, it seems rather logic that we have this kind of similarity.

The system has its resonance frequency at 5.8 rad/s, which is almost exactly at the same frequency as the deepest dip in the controller magnitude

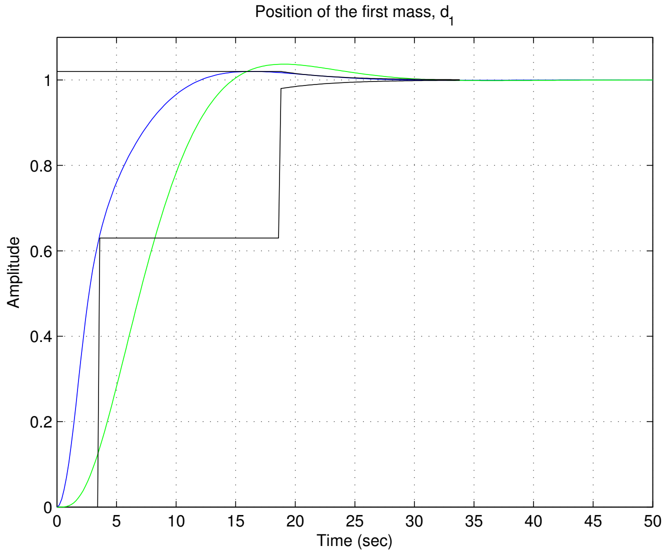


Figure 12.1 Step responses from reference r to mass position d_1. The response violating the constraints we have posed is the one given with the nominal control and the other one comes from using our optimized controller.

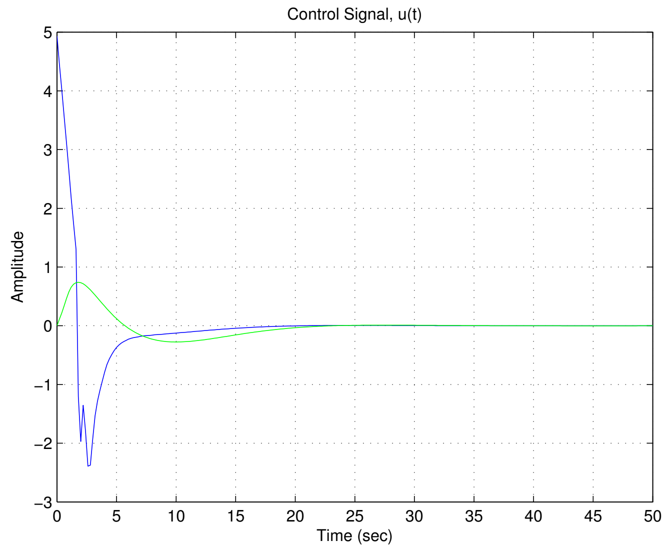


Figure 12.2 The nominal controller gives the control signal with the least energy of the two.

plot. The logic behind the dip is that we do not want to augment signals at this frequency. A PD controller does not have this kind of flexibility in its structure to damp out a certain frequency and is therefore not so well suited for highly oscillative systems like this one.

d. The least value on `NQ`, for which our problem is feasible, is 7. The constraints that fail are those posed on d_1 during a step change in the reference. See Figure 12.6.

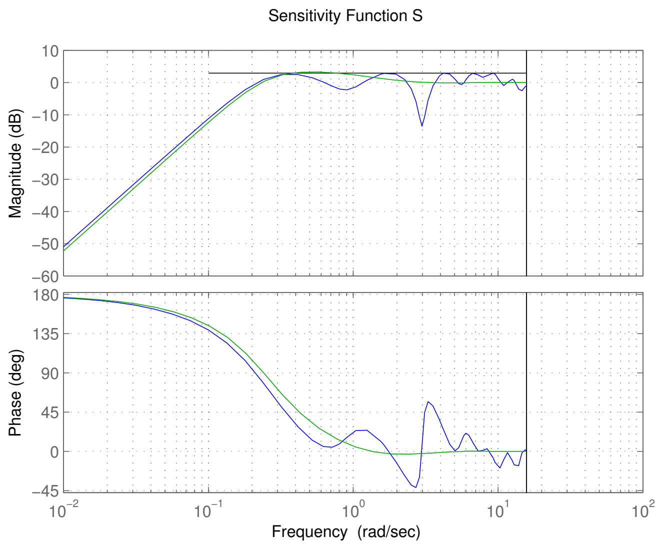


Figure 12.3 The sensitivity function of the system. The plot that violates the constraint correspond to the nominal control.

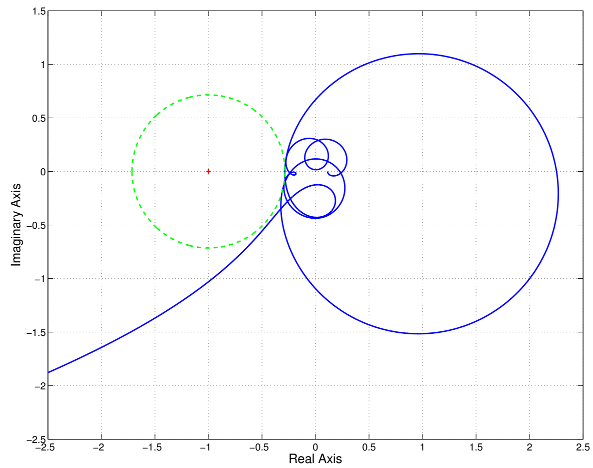


Figure 12.4 The open loop Nyquist curve and the M_s-circle when the optimal controller is used.

e. See Figure 12.7 for a plot of the cost function value versus the order of the Q-filters. When `NQ` reaches around 20, the control will gain very little from increased complexity of the Q-filter. We can then say that we have a good estimate of the *Limit of Performance*, i.e. lowest cost that linear controller can achieve given the problem setup.

The maximum value of the control signal will decrease as the order of the Q-filters goes up. For `NQ`$= [7, 10, 15, 20]$, we get $u_{max} = [6.0, 4.9, 3.8, 3.7]$. This means that the more complex the controller becomes, the more freedom it will have to choose its control signal. As it is good to have a control signal that is low on energy (due to the cost function), it is also likely that

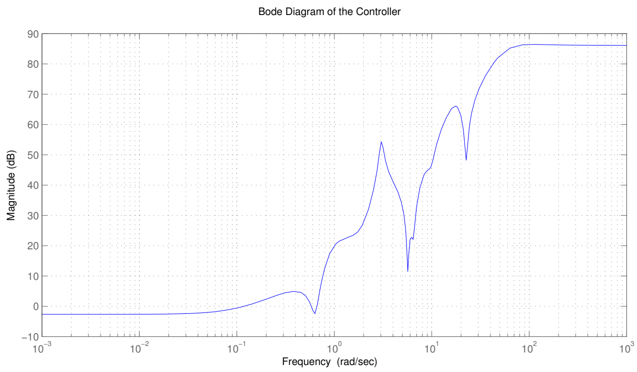


Figure 12.5 Bode diagram of the controller.

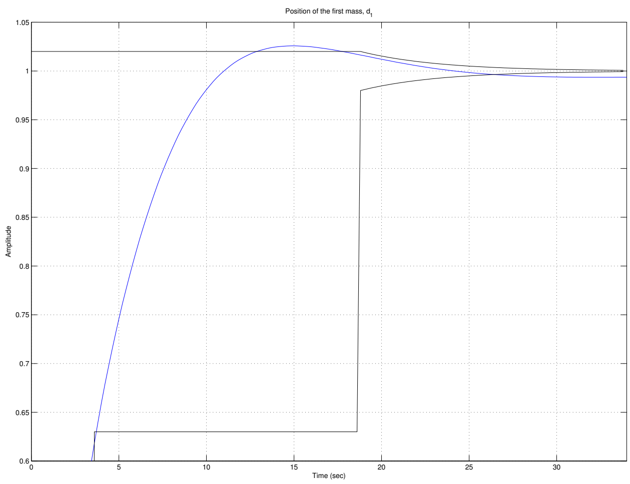


Figure 12.6 The problem is infeasible with `NQ=6`.

it goes down in magnitude if it has the possibility.

f. If we start out by setting $\rho = 0$ (i.e. `weight_u = 0.0`), then we do not punish the control signal energy at all, which means that we may get very agressive and poorly damped control. The constraint on u_{max} will to some extent prevent this, but if u_{max} is made arbitrarily large then we can get a step responses like the one in Figure 12.8. If instead $\gamma = 0$ while ρ remains 1, then the solution will remain fairly unchanged. The reason is that both the constraints on rise time and the cost of having a large e will force u to be quite active still. If the step response constraints are made inactive, the solution will be fairly close to the one of the nominal LQG controller.

g. The solution will still be feasible when the overshoot is constrained to 0.4% (`os=1.004`). In order to make the least possible overshoot even smaller we can for instance set:

- $u_{max} = 7$, which will give the control signal a bigger span to work with
- $M_s = 1.6$, gives the controller a higher gain (see Figure 12.4 and imag-

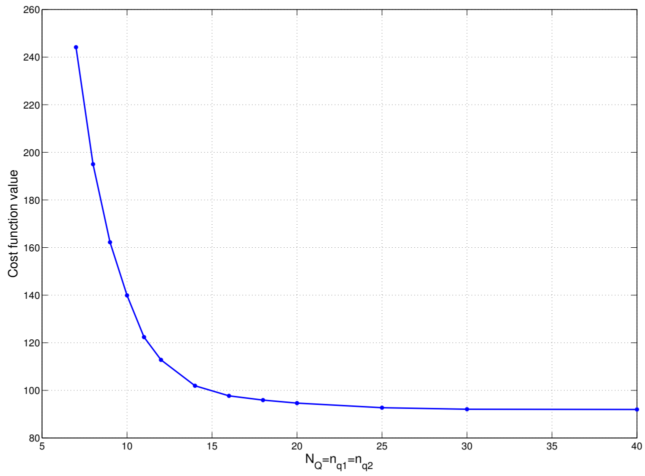


Figure 12.7 Cost function value plotted against the complexity of the Q-filters.

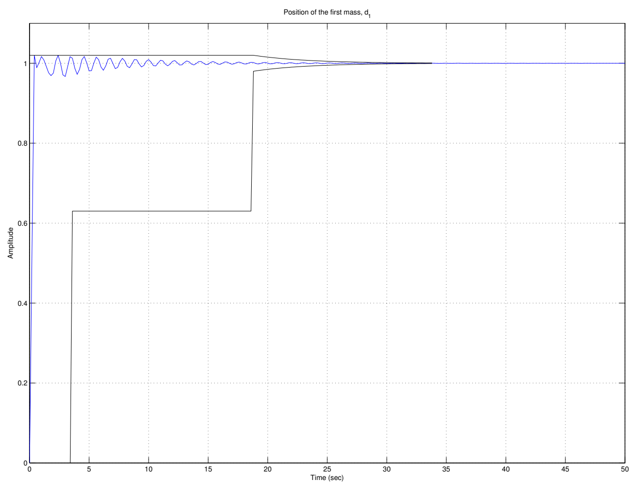


Figure 12.8 d_1 due to a unit step in the reference when $\rho = 0$. The control is very agressive.

ine how the open loop system can change if the M_s-circle is made smaller), making it be able take assume greater values, within the boundaries set by u_{max}.

- $N_Q = 30$, which will give the controller more freedom to shape the control signal optimally.

All these 3 changes will be able to handle an overshoot of 0.2% respectively. $N_Q = 30$ will also be able to shape the step response such that there is no overshoot (see Figure 12.9). Note that t_2 had to be set to 34 (maximum time for which we have constraints) in order for the lower step constraints to not make the problem too difficult. Try with a lower value of t_2 to see the effect.

h. In `plot_result.m` we find the following line of code

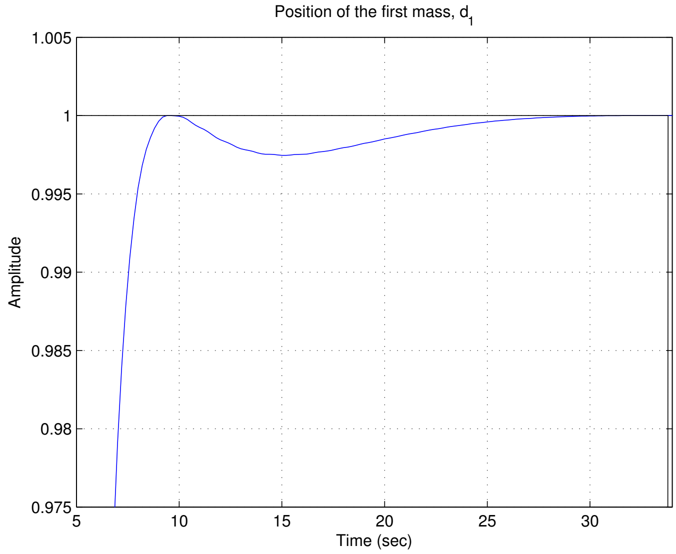


Figure 12.9 d_1 due to a unit step in the reference when no overshoot is allowed.

```
cl = prob.T11 + prob.T12*[prob.Q{1} prob.Q{2}]*prob.T21;
```

which is used before the plots are created. This line corresponds to the formula stated in the previous exercise session, namely $H = P_{zw} + P_{zu}QP_{yw}$, where P includes the nominal controller. Copying this line of code into the Matlab prompt, followed by `bodemag(cl)` will give us the 9 desired plots. Looking at the middle one, corresponding to $H_{u_o n}$, we can see that it does not have high frequency roll-off. A high frequency measurement noise, n, may therefore lead to a very noise control signal. This shows that it is very important to take all signals in a system into consideration and that a solution, even though optimal, may need to be reconsidered. If we were to modify the problem, a good idea may therefore be to put constraints on this closed loop transfer function as well.

i. No solution given.

Solutions to Exercise 13. Controller simplification

13.1

a. Inspection of the locations of the poles and zeros gives us the transfer function

$$G(s) = 1.04 \frac{s/1.3 + 1}{(s/1.2 + 1)(s^2 + 0.4s + 1.04)}$$

b. The closeness of the pole-zero pair on the real axis suggests that a model reduction might be possible.

c. A balanced realization and the Hankel singular values for the system can be calculated using the Matlab command

```
>>> s = tf('s');
>>> G = 1.04*(s/1.3+1)/((s/1.2+1)*(s^2+0.4*s+1.04));
>>> [balr,g] = balreal(G);
```

which gives the following Hankel singular values:

$$g = \begin{pmatrix} 1.5105 \\ 1.0196 \\ 0.0091 \end{pmatrix}$$

Elimination of the state in the balanced realization corresponding to the smallest Hankel singular value is done in Matlab by

```
>>> modsys = modred(balr,g<0.01)
>>> modsysG = tf(modsys)
```

This gives the following transfer function for the reduced order system:

$$G_{red}(s) = 0.0181 \frac{s^2 - 2.412s + 57.49}{s^2 + 0.4086s + 1.043}$$

A Bode magnitude plot of the original system and the reduced system is shown in figure 13.1.

13.2 a. With

$$S = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \quad A = \begin{pmatrix} -1 & 0 \\ -1 & -0.5 \end{pmatrix}, \quad B = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

we have

$$AS + SA^T + BB^T = \begin{pmatrix} -2 & 0 \\ -2 & -0.5 \end{pmatrix} + \begin{pmatrix} -2 & -2 \\ 0 & -0.5 \end{pmatrix} + \begin{pmatrix} 2 \\ 1 \end{pmatrix} \begin{pmatrix} 2 \\ 1 \end{pmatrix}^T = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

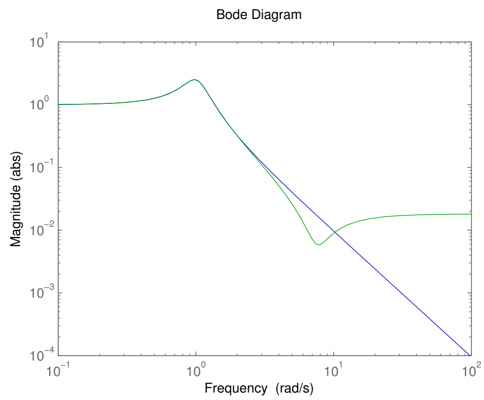


Figure 13.1 Bode diagram of the original and reduced system in problem 13.1

so S is the controllability gramian. Similarly, with

$$O = \begin{pmatrix} 0.5 & 0 \\ 0 & 1 \end{pmatrix}$$

$$OA + A^T O + C^T C = \begin{pmatrix} -0.5 & 0 \\ -1 & -0.5 \end{pmatrix} + \begin{pmatrix} -0.5 & -1 \\ 0 & -0.5 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

so O is the observability gramian.

b. The Hankel singular values are the eigenvalues of

$$SO = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

so they are both 1.

c. The coordinate change $\xi = Tx$ yields the new gramians $S_\xi = TST^T$ and $O_\xi = T^{-T} O T^{-1}$. If T is chosen as

$$T = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 \\ 0 & 1 \end{pmatrix}$$

the gramians become

$$S_\xi = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad O_\xi = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Hence, a balanced realization is

$$\dot{\xi} = \hat{A}\xi + \hat{B}u$$
$$y = \hat{C}\xi + \hat{D}u$$

where

$$\hat{A} = TAT^{-1} = \begin{pmatrix} -1 & 0 \\ -\sqrt{2} & -0.5 \end{pmatrix} \quad \hat{B} = TB = \begin{pmatrix} \sqrt{2} \\ 1 \end{pmatrix}$$
$$\hat{C} = CT^{-1} = \begin{pmatrix} \sqrt{2} & 1 1 \end{pmatrix} \quad \hat{D} = D$$

d. In this case, the Hankel singular values have the same size, therefore either could be removed. (However, this means that it is probably not a good idea to do any truncation at all!) If the second state is removed by letting $\dot{\xi}_2 = 0$, ξ_2 can be expressed in terms of ξ_1 through $0 = \hat{A}_{21}\xi_1 + \hat{A}_{22}\xi_2 + \hat{B}_2 u$. The reduced realization then becomes

$$\begin{aligned}\dot{\xi}_1 &= (\hat{A}_{11} - \hat{A}_{12}\hat{A}_{22}^{-1}\hat{A}_{21})\xi + (\hat{B}_1 - \hat{A}_{12}\hat{A}_{22}^{-1}\hat{B}_2)u \\ y_r &= (\hat{C}_1 - \hat{C}_2\hat{A}_{22}^{-1}\hat{A}_{21})\xi + (\hat{D} - \hat{C}_2\hat{A}_{22}^{-1}\hat{B}_2)u\end{aligned}$$

where for example $\hat{A}_{21}$ is the element in the second row and first column in $\hat{A}$.

$$\begin{aligned}\dot{\xi}_1 &= -\xi_1 + \sqrt{2}u \\ y_r &= -\sqrt{2}\xi_1 + 12u\end{aligned}$$

The transfer function is obtained through the Laplace transform

$$G_1(s) = 12 - \frac{2}{s+1}$$

13.3 a. The Matlab command `tf(ss(A,B,C,D))` gives

$$G(s) = \frac{10s^2 + 18s + 5}{s^2 + 1.5s + 0.5}$$

b. Plotting the Bode diagram for $G(s)-G_1(s)$ through the command `bodemag(G-G1)` gives 2 as the maximal error, obtained at large frequencies. The error bound, twice the sum of the truncated singular values, also gives 2. In this case the error bound is tight.

c. Truncating both states gives

$$G_2 = \hat{D} - \hat{C}\hat{A}^{-1}\hat{B} = 10$$

d. Plotting `bodemag(G-Gr)` gives 2 as the maximal error, near $\omega = 1$. The error bound $2(1+1) = 4$ is conservative.

13.4 Through partial fractions one can write

$$\frac{2s^2 + 2.99s + 1}{s(s+1)^2} = \frac{1}{s} + \frac{s + 0.99}{(s+1)^2}$$

The Matlab command
`[G3bal,sig]=balreal(tf([1 .99],[1 2 1]))` gives

$$sig = \begin{pmatrix} 0.4950 \\ 0.00001 \end{pmatrix}$$

so one state can be removed right away.
`G3red=modred(G3b,(sig<0.1))` yields

$$\frac{1}{s + 1.01}$$

Adding the integrator back gives the reduced system

$$\frac{1}{s} + \frac{1}{s+1.01} = \frac{2s+1.01}{s(s+1.01)}$$

Note that it is not necessary to do the separation into a stable and an unstable system by hand, the Matlab command `stabsep` can be used to do this. Also, the commands `balreal` and `modred` can actually be used for systems with unstable poles since they do the separation automatically.

Solutions to Exercise 14. Old Exam

14.1 Partial fraction expansion gives

$$G(s) = \begin{bmatrix} \frac{1}{(s+1)(s+2)} & \frac{s+3}{(s+1)(s^2+6s+8)} \end{bmatrix}$$
$$= \frac{1}{s+1}\begin{bmatrix} 1 & \frac{2}{3} \end{bmatrix} + \frac{1}{s+2}\begin{bmatrix} -1 & -\frac{1}{2} \end{bmatrix} + \frac{1}{s+4}\begin{bmatrix} 0 & -\frac{1}{6} \end{bmatrix}$$

so a realization on diagonal form can be written as

$$\dot{x} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -4 \end{bmatrix} x + \begin{bmatrix} 1 & \frac{2}{3} \\ -1 & -\frac{1}{2} \\ 0 & -\frac{1}{6} \end{bmatrix} u$$
$$y = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} x$$

14.2 Denote the output by z. The spectral density of z is then

$$\Phi_z = |G(i\omega)|^2 \Phi_n(\omega) = \frac{1}{i\omega + a}\frac{1}{-i\omega + a} =$$
$$= \frac{1}{a^2 + \omega^2}$$

14.3

a. The bad damping in the disturbance response is a symptom of low phase margin, which is approximately 20° at $\omega_c = 1$ (as seen in the bode diagram). The lead filter improves the phase margin, but the phase peak is located between the zero and pole at

$$\omega_p = \sqrt{1.79 \cdot 8.94} = 4\,\text{rad/s},$$

which is far from ω_c!

b. One way to improve the control is to move the phase peak to $\omega_c = 1$ by dividing the pole and zero by 4. The new controller is

$$C'(s) = K\left(1 + \frac{1}{s}\right)\frac{s/0.45 + 1}{s/2.24 + 1}.$$

The gain K should be chosen so that the cutoff frequency is preserved, that is $|C'(i\omega_c)| = |C(i\omega_c)|$, which gives $K = 0.45$.

The new controller gives an increase in phase margin of 19°. The high frequency gain

$$\lim_{s\to\infty} C(s)$$

actually decreases from 4.39 to 2.24.

14.4

a. Block scheme calculations gives

$$G_{y_2,m} = \frac{B_1(s)(s^2 - \omega_0^2)}{s[A_1(s)(s^2 - \omega_0^2) + B_1(s)\omega_0^2]}$$

b. A zero at $z = \omega_0$ imposes a constraint on the achievable bandwidth of the closed loop system – it is not possible to achieve a bandwidth significantly larger than ω_0.

c. Plot D shows too high bandwidth of the closed loop to be feasible. Plot B and C does not fulfill the $|S + T| = 1$ constraint. Plot A shows a bandwidth of about 1rad/s which is reasonable - hence plot A.

14.5

a. The determinant of the system is

$$det(G_2(s)) = \frac{1}{(s+1)(s+2)}$$

such that the least common denominator of all submatrices is $p(s) = (s + 1)(s + 2)$. Thus, the system have poles in -1, -2 and no zeros. Since the system has poles in the open left halplane and no non-minimum phase zeros, there are no fundamental limitations on the system bandwidth.

b.

$$RGA = G_2(0). * (G_2^T(0))^{-1} = \begin{pmatrix} -0.5 & 0 \\ -3 & -1 \end{pmatrix} .* \begin{pmatrix} -2 & 6 \\ 0 & -1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Since the RGA is the identity matrix we can expect the system to be easily controlled with decentralized control at low frequencies. The identity matrix also gives us that it is suitable to pair input 1 with output 1 and input 2 with output 2.

14.6 Using the IMC method, we set $Q(s)$ to

$$Q(s) = \frac{1}{(\lambda s + 1)^{n-m}} G^{-1}(s)$$

giving us the controller

$$C(s) = (1 - Q(s)G(s))^{-1}Q(s) = \left(1 - \frac{1}{(\lambda s + 1)^{n-1}}\right)^{-1} \frac{(s+b)^n}{(s+a)(\lambda s + 1)^{n-1}}$$

$$= \frac{(s+b)^n}{(s+a)((\lambda s + 1)^{n-1} - 1)}$$

To match the structure of the PID controller

$$\frac{K}{sT_i} \frac{(T_i T_d s^2 + T_i s + 1)}{(s\frac{T_d}{N} + 1)}$$

we see that we will need to choose $n = 2$. This leaves us with

$$C(s) = \frac{b^2}{s\lambda a}\frac{(\frac{1}{b^2}s^2 + \frac{2}{b}s + 1)}{(\frac{s}{a} + 1)},$$

such that we can now determine the PID parameters one by one

$$T_i = \frac{2}{b}, \quad T_d = \frac{1}{T_i b^2} = \frac{1}{2b}, \quad K = \frac{b^2 T_i}{\lambda a} = \frac{2b}{\lambda a}, \quad N = T_d a = \frac{a}{2b}.$$

Since only K depends on λ, this is the only PID parameter that we have the possibility to tune ourselves.

14.7 P can be said to consist of several submatrices

$$P = \begin{pmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{pmatrix},$$

where

$$P_{zw} = \begin{pmatrix} P_{e_1r_1} & P_{e_1r_2} \\ P_{e_2r_1} & P_{e_2r_2} \end{pmatrix}, \quad P_{zu} = \begin{pmatrix} P_{e_1u_1} & P_{e_1u_2} \\ P_{e_2u_1} & P_{e_2u_2} \end{pmatrix},$$

$$P_{yw} = \begin{pmatrix} P_{y_1r_1} & P_{y_1r_2} \\ P_{y_2r_1} & P_{y_2r_2} \\ P_{r_1r_1} & P_{r_1r_2} \\ P_{r_2r_1} & P_{r_2r_2} \end{pmatrix}, \quad P_{yu} = \begin{pmatrix} P_{y_1u_1} & P_{y_1u_2} \\ P_{y_2u_1} & P_{y_2u_2} \\ P_{r_1u_1} & P_{r_1u_2} \\ P_{r_2u_1} & P_{r_2u_2} \end{pmatrix}$$

We can now determine all transfer functions that make up P:

$$P_{zw} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I, \quad P_{zu} = \begin{pmatrix} -P^0_{11} & -P^0_{12} \\ -P^0_{21} & -P^0_{22} \end{pmatrix} = -P^0,$$

$$P_{yw} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 \\ I \end{pmatrix}, \quad P_{yu} = \begin{pmatrix} P^0_{11} & P^0_{12} \\ P^0_{21} & P^0_{22} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} P^0 \\ 0 \end{pmatrix}$$

14.8

a. Zeros of a multivariate process are defined as the points where the transfer matrix looses rank. For a quadratic matrix, losing rank is equivalent to the determinant being zero. The zeros are given by the equation:

$$\det P(s) = \frac{2s + 5}{(s + 4)(s + 2)(s + 1)} = 0$$

Thus s = -2.5

b. Here RGA(0) is calculated.

$$G(0) = \begin{bmatrix} 0.5 & -0.5 \\ 1 & 0.25 \end{bmatrix}$$

Then RGA(0) becomes

$$RGA(0) = G(0). * (G(0)^{-1})^T = \begin{bmatrix} 0.2 & 0.8 \\ 0.8 & 0.2 \end{bmatrix}$$

and thus output 1 should pair with input 2 and output 2 with input 1.

14.9 We see that A and C will give the same controller since we have just scaled the weights by 100, so A and C will correspond to Step response 1 and 2. Notice that the system is very oscillative. D has much larger weight on the control signal, thus we will not be able to get a fast system that dampens the oscillative system, hence D must correspond to Step response 3. B will correspond to Step response 4. We have very small weight on the control signal compared to output, which will give a fast system.