Exercise 7 - SISO Performance limitations

Problem 7.1: High-gain feedback

Consider the three single input single output systems

$$G_1(s) = \frac{-s+1}{s+1}, \quad G_2(s) = \frac{s+1}{-s+1} \quad \text{and} \quad G_3(s) = \frac{-s+1}{-s+2}.$$

Use root locus analysis to determine for which values of a proportional controller K the closed-loop systems with negative feedback can be stabilised. Note that the controller gain is located in the feedback path which yields the characteristic equations

$$1 + KG_i(s) = 0 \qquad \text{for } i \in \{1, 2, 3\}.$$

Solution:
It is possible to plot the root loci manually for the three systems $G_1(s)$, $G_2(s)$ and $G_3(s)$. Mark the open-loop poles and zeros in the complex plane. As we have just one pole in the open-loop transfer functions, each locus has only one branch. The locus starting points ($K = 0$) are the open-loop poles and the ending points ($K = \infty$) are the open-loop zeros. The loci will lie entirely on the real axis.

It is also possible to plot the root loci with MATLAB:

```
%% Root loci for G1, G2 and G3

% System definitions
s  = tf( 's' );
G1 = (-s + 1) / (s + 1);
G2 = (s + 1) / (-s + 1);
G3 = (-s + 1) / (-s + 2 );

% Plot root loci
rlocus( G1 ), title( 'G1' );
figure, rlocus( G2 ), title( 'G2' );
figure, rlocus( G3 ), title( 'G3' );
```

System $G_1(s)$:
The characteristic equation of system 1 can be rearranged to

$$\begin{aligned} & 0 = 1 + K\frac{-s+1}{s+1} \\ \Leftrightarrow \quad & 0 = (s+1) + K(-s+1) \\ \Leftrightarrow \quad & s = \frac{K+1}{K-1}. \end{aligned}$$

We obtain the root locus by varying the controller gain K of the closed-loop system. The closed-loop poles move from the open-loop poles to the open-loop zeros. For $0 < K \leq 1$, the locus moves from $s = -1$ to $s = -\infty$. For $K > 1$, the locus moves from $s = \infty$ to

$s = 1$. In this case a P-controller with a gain of more than 1 will destabilise the system by introducing a closed-loop pole in the RHP.

System $G_2(s)$:
The characteristic equation of system 2 can be rearranged to

$$\begin{aligned} & 0 = 1 + K\,\frac{s+1}{-s+1} \\ \Leftrightarrow \quad & 0 = (-s+1) + K(s+1) \\ \Leftrightarrow \quad & s = \frac{K+1}{1-K}. \end{aligned}$$

For $0 < K \leq 1$, the locus moves from $s = 1$ to $s = \infty$. For $K > 1$, the locus moves from $s = -\infty$ to $s = -1$. In this case, a P-controller with a gain of less than 1 will destabilise the system by introducing a closed loop pole in the RHP. In other words, a controller gain of more than 1 will stabilise the unstable plant.

System $G_3(s)$:
The characteristic equation of system 3 can be rearranged to

$$\begin{aligned} & 0 = 1 + K\,\frac{-s+1}{-s+2} \\ \Leftrightarrow \quad & 0 = (-s+2) + K(-s+1) \\ \Leftrightarrow \quad & s = \frac{K+2}{K+1}. \end{aligned}$$

The trajectory of the closed-loop poles lies in between $s = 2$ and $s = 1$, when the controller gain is increased from $K = 0$ to $K = \infty$. In this case, it is impossible to stabilise the system with a P-controller.

Problem 7.2: Bode sensitivity integral

Check if the first "waterbed formula" holds for the unstable system with open-loop transfer function

$$L(s) = \frac{5}{s^2 + s - 2}$$

by evaluating BODE's sensitivity integral and the corresponding residual.

Solution:
The first "waterbed formula" is defined as

$$\int_0^\infty \ln |S(j\omega)|\, \mathrm{d}\omega = \pi \sum_{i=1}^{N_p} \mathrm{Re}(p_i).$$

We evaluate the left-hand side of the equation first. The sensitivity function

$$S = \frac{1}{1+L} = \frac{1}{1 + \frac{5}{s^2+s-2}} = \frac{s^2+s-2}{s^2+s+3}.$$

The absolute value of the sensitivity function evaluated at $s = j\omega$

$$|S(j\omega)| = \frac{\sqrt{(2+\omega^2)^2 + \omega^2}}{\sqrt{(3-\omega^2)^2 + \omega^2}}$$

which yields the integration

$$\int_0^\infty \ln|S(j\omega)|\,\mathrm{d}\omega = \int_0^\infty \frac{1}{2}\ln\left((2+\omega^2)^2+\omega^2\right) - \frac{1}{2}\ln\left((3-\omega^2)^2+\omega^2\right)\mathrm{d}\omega = \pi.$$

The result can be calculated by hand using a relationship from the book of BRONSTEIN or the like. Alternatively, the solution can be found symbolically using MATLAB:

```
%% Symbolical integration of Bode's sensitivitiy integral
syms w;
int( 0.5 * log( (2 + w^2)^2 + w^2) ...
   - 0.5 * log( (3 - w^2)^2 + w^2), 0, inf )
```

For the right-hand side of the equation we evaluate the RHP-poles of the open-loop transfer function

$$L(s) = \frac{5}{s^2+s-2} = \frac{5}{(s+2)(s-1)}.$$

It has one unstable pole at $p_1 = 1$, hence $N_p = 1$ and

$$\pi\sum_{i=1}^{N_p}\mathrm{Re}(p_i) = \pi\sum_{i=1}^{1}\mathrm{Re}(1) = \pi.$$

The first "waterbed formula" holds.

Problem 7.3: Poisson sensitivity integral

Analyse the proper rational plant given by the transfer function

$$G(s) = \frac{s-4}{(s-3)(s+1)}.$$

(a) Stabilise the plant with a YOULA parameterisation using coprime factorisation in MATLAB.

(b) Check if the second "waterbed formula" holds for the stabilised system by evaluating POISSON's sensitivity integral and the corresponding residual.

Solution:

(a) A state space representation of the plant is given by

$$\mathbf{G}(s) \overset{s}{=} \left[\begin{array}{c|c}\mathbf{A} & \mathbf{B}\\ \hline \mathbf{C} & \mathbf{D}\end{array}\right].$$

With a state-feedback matrix $\mathbf{F}$ and a observer-gain matrix $\mathbf{H}$ such that $\mathbf{A}+\mathbf{BF}$ and $\mathbf{A}+\mathbf{HC}$ are both stable, a coprime factorisation can be constructed from

$$\begin{bmatrix}\mathbf{M} & \mathbf{U}\\ \mathbf{N} & \mathbf{V}\end{bmatrix} \overset{s}{=} \left[\begin{array}{c|c}\mathbf{A}+\mathbf{BF} & \begin{bmatrix}\mathbf{B} & -\mathbf{H}\end{bmatrix}\\ \hline \begin{bmatrix}\mathbf{F}\\ \mathbf{C}+\mathbf{DF}\end{bmatrix} & \begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{D} & \mathbf{I}\end{bmatrix}\end{array}\right].$$

The coprime factorisation is obtained with a combined state-feedback/observer structure, for which positive feedback is assumed. Hence, the factors satisfy a modified BEZOUT identity

$$\mathbf{VM} - \mathbf{UN} = \mathbf{I}.$$

A stabilising controller for $\mathbf{Q} = \mathbf{0}$ is given by

$$\mathbf{K} = -\mathbf{V}^{-1}\mathbf{U}.$$

The stabilising controller can be calculated with MATLAB:

```
%% Design a Youla-based stabilising controller

% System definiton
s    = tf( 's' );
G    = (s - 4) / ( (s - 3) * (s + 1) );

% Transformation to state space
Gss = ss( G );
A    = Gss.a;
B    = Gss.b;
C    = Gss.c;
D    = Gss.d;

% Calculate state-feedback and observer-gain matrices
F = -place( A, B, [ -2 -5 ] );
H = -transpose( place( A', C', [ -2 -5 ] ) );

% Coprime factorisation
Gcf  = tf( ss( A + B * F, [ B, -H ], ...
              [ F; C + D * F ], [ 1, 0; D, 1 ] ) );
M    = Gcf(1,1);
N    = Gcf(2,1);
U    = Gcf(1,2);
V    = Gcf(2,2);

% Check if Bezout's identity holds
tf( minreal( V * M - U * N ) )

% Design controller
K = -minreal( U / V );
step( feedback( G * K, 1 ) )
```

(b) The second "waterbed formula" is defined as

$$\int_0^\infty \ln |S(j\omega)|\, w(z,\omega)\, \mathrm{d}\omega = \pi \ln \prod_{i=1}^{N_p} \left| \frac{p_i + z}{\bar{p}_i - z} \right|$$

where if the zero is real

$$w(z,\omega) = \frac{2z}{z^2 + \omega^2}.$$

We evaluate the left-hand side of the equation first. The open-loop transfer function of the stabilised system

$$L = GK = \frac{-399.2\,(s-4)(s+1.008)}{(s-3)(s+1)(s^2+16s+503.2)}$$

which yields the sensitivity function

$$S = \frac{1}{1+L} = \frac{(s-3)(s+1)(s^2+16s+503.2)}{(s+5)^2(s+2)^2}.$$

The open-loop transfer function $L(s)$ has a single RHP-zero $z = 4$, hence

$$w(z, \omega) = \frac{8}{16 + \omega^2}.$$

To save us further trouble, we solve the integral numerically with MATLAB:

```
%% Numerical integration of Poisson's sensitivitiy integral
value = integral( @intArg, 0, inf )
```

The argument of the integral is defined as:

```
function val = intArg( w )
num = (1i.*w - 3) .* (1i.*w + 1) .* ( (1i.*w).^2 + 16.*1i.*w + 503.2
  );
den = (1i.*w + 5).^2 .* (1i.*w + 2).^2;
val = log( abs( num ./ den ) ) .* ( 8 ./ ( 16 + w.^2 ) );
end
```

For the right-hand side of the equation we evaluate the RHP-poles of the open-loop transfer function $L(s)$. It has one unstable pole at $p_1 = 3$, therefore $N_p = 1$ and

$$\pi \ln \prod_{i=1}^{N_p} \left| \frac{p_i + z}{\bar{p}_i - z} \right| = \pi \ln \left| \frac{3+4}{3-4} \right| \approx 1.946\pi.$$

The second "waterbed formula" holds.

Problem 7.4: Balancing a rod

Consider the stabilisation problem of balancing a rod, the linearised equation of motion with observation of the rod's tip is

$$G(s) = \frac{-g}{s^2\left(Mls^2 - (M+m)\,g\right)}$$

where $g = 10\,\frac{\mathrm{m}}{\mathrm{s}^2}$ and $M = m = 0.5\,\mathrm{kg}$. The bandwidth of human hand movement is assumed to be $2\,\mathrm{Hz}$.

(a) Determine the minimal length l of a rod that can be stabilised by a human (with $M_T = 2$).

(b) Assume a rod length $l = 30\,\mathrm{cm}$. Is it possible to stabilise the rod with a PI-controller of the form

$$K(s) = K\frac{\tau s + 1}{\tau s}$$

with $K, \tau > 0$ and negative feedback?

Solution:

(a) The system $G(s)$ has the four poles

$$p_i = \left\{0, 0, +\sqrt{\frac{(M+m)g}{Ml}}, -\sqrt{\frac{(M+m)g}{Ml}}\right\}.$$

The RHP pole introduces a limitation for the lower bandwidth that is needed for stabilisation. Hence, assuming complementary sensitivity peak of $M_T = 2$ the bandwidth

$$\omega_{BT}^* > 2\sqrt{\frac{(M+m)g}{Ml}}$$

with assumed bandwidth of human movement $\omega_{BT}^* = 4\pi$. Rearranging yields

$$l > \frac{4(M+m)g}{M\left(\omega_{BT}^*\right)^2} \approx 50\,\mathrm{cm}$$

(b) The transfer function of the closed-loop system or complementary sensitivity is given by

$$\begin{aligned} T(s) &= \frac{G(s)K(s)}{1+G(s)K(s)} \\ &= \frac{\frac{-g}{s^2(Mls^2-(M+m)g)}\frac{K(s+\frac{1}{\tau})}{s}}{1+\frac{-gK(s+\frac{1}{\tau})}{s^3(Mls^2-(M+m)g)}} \\ &= \frac{-gK(s+\frac{1}{\tau})}{Mls^5-(M+m)gs^3-gKs+\frac{gK}{\tau}}. \end{aligned}$$

The denominator of the complementary sensitivity is the characteristic polynomial. Irrespective of the values K and τ, the two coefficients $a_4, a_2 = 0$. The Hurwitz stability criterion implies that a system is stable if and only if the coefficients of the characteristic polynomial satisfy $a_i > 0$ for $i \in \{1, 2, \dots, n\}$. Hence, it is impossible to stabilise the plant using a PI-controller.