Control Theory and Practice
Advanced Course

Computer Exercise:
H_∞ CONTROL DESIGN

Marcus Berner and Peng Zhou, 2008

1   Introduction

In H_∞ control design the sensitivity and complementary sensitivity function are shaped to meet certain desired specifications. The main goal of this lab is to get a feeling for how H_∞ control design can be used to obtain desired specifications on sensitivity and robustness.

Preparations: Recollect the following topics from the basic control course:

- Sensitivity and complementary sensitivity functions.
- Robustness and model errors.
- Poles and zeros effect on the system dynamics and the Bode diagram.

Read this document and answer the preparation tasks, 2.1.1, 2.2.1 and 2.3.1 before the exercise.

Presentation: All problems in this exercise should be solved, but only the tasks on the report form should be handed in. The report form and the date when it should be handed in can be found on the course website. The exercise should be performed in pairs of students.

2   Background

This section will begin with a brief summary of some important basic theory followed by more specific theory for the H_∞ design method.
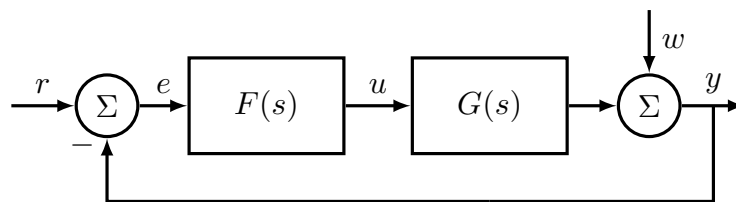

Figure 1: Block diagram of the feedback system used in this lab.

The system used in the entire lab has the structure depicted in fig. 1. The signals indicated in the figure are:

r: Reference value
e: Control error
u: Control signal
w: Disturbance on the output
y: Output

2.1 Sensitivity function and reduction of disturbances

The sensitivity function, denoted S, is the transfer function from the disturbance to the output, see eq. (1). Note that the equation only describes the relation between the disturbance and the output. The reference is therefore assumed to be zero.

$$y = Sw, \quad S = (1 + GF)^{-1} \tag{1}$$

By making the amplification of the sensitivity function small, the effects of disturbances on the output can be reduced. As this lab will show it is not possible to make it arbitrary small for all frequencies. This can easily be realized by looking at the amplification of the controller required to make the sensitivity very small.

2.1.1 Preparation task 1

Use the block diagram in fig. 1 to show that the transfer function from the disturbance (w) to the output (y) satisfies eq. (1).

....

....

....

2.2 Complementary sensitivity and robustness

The complementary sensitivity function T, given by

$$T = (1 + GF)^{-1}GF, \tag{2}$$

can be used to prove robustness to model errors. For that a new system with model uncertainty Δ_G is introduced. The new system is depicted in fig. 2.
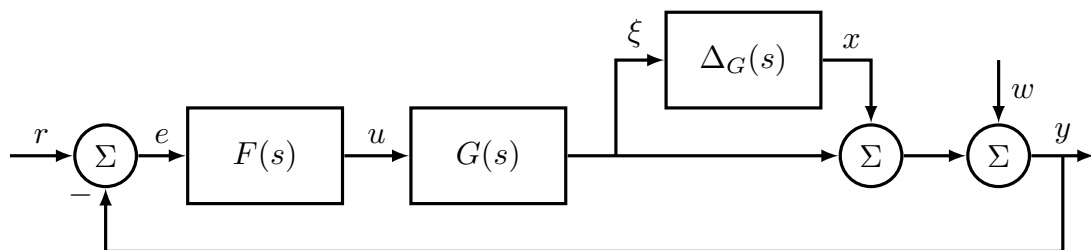

Figure 2: Block diagram of the feedback system with model error

The system in fig. 2 can be rewritten as the system depicted in fig. 3. With the small gain theorem, see the course book [1] or [2], closed loop stability can be guaranteed if Δ_G and T both are stable and condition (3) is satisfied.

$$|T(i\omega)| < \frac{1}{|\Delta_G(i\omega)|}, \quad \forall \omega \tag{3}$$

One important remark about the above result is that the small gain theorem is conservative. The condition on T is therefore sufficient but not necessary for stability.
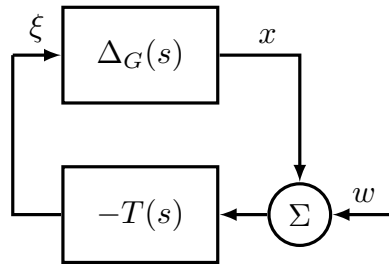

Figure 3: Block diagram of the feedback system with model error on the form used in the small gain theorem

2.2.1 Preparation task 2

Show that the systems in figs. 2 and 3 are equivalent if the reference signal is zero.

....

....

....

....

....

2.3 H_∞ control design

In H_∞ control design the three functions S, T and G_{wu} are shaped to meet desired performance. The first two have already been discussed. Here G_{wu} denotes the transfer function from the disturbance to the control signal, see eq. (4) (where it is assumed $r = 0$).

$$u = G_{wu}w, \quad G_{wu} = -(1 + FG)^{-1}F = -FS \tag{4}$$

It is desirable to make the magnitude of S, T and G_{wu} small. That is unfortunately not possible because they are related to each other. To deal with that, the weights (transfer functions) W_S, W_T and W_U are introduced. They decide how much emphasis to put on minimizing each closed loop transfer function.

After choosing the weights the following problem is solved:

Find F such that

$$\left\| \begin{bmatrix} W_S S \\ W_T T \\ W_U F S \end{bmatrix} \right\|_\infty \leq \gamma$$

with the smallest value of γ possible.

This is an approximate, but computationally convenient, way of expressing the desire that all the individual transfer functions $W_S S$, $W_T T$ and $W_U F S$ should have H_∞-norm less than γ.

The requirement that the individual transfer functions should have H_∞-norm less than γ can be rewritten as

$$\begin{cases} |S(i\omega)| \leq \gamma |W_S^{-1}(i\omega)|, & \forall \omega \\ |T(i\omega)| \leq \gamma |W_T^{-1}(i\omega)|, & \forall \omega \\ |FS(i\omega)| \leq \gamma |W_U^{-1}(i\omega)|, & \forall \omega \end{cases} \tag{5}$$

In this way, we can see the weight functions as a means to define "forbidden regions" for the frequency responses of the individual transfer functions.

To compute the controller that gives the smallest value of γ is far from trivial, especially for higher order systems and weights. It is therefore not done by hand in this lab. The computations are instead done numerically in the design tool.

2.3.1 Preparation task 3

Use the block diagram in fig. 1 to show that the transfer function from the disturbance (w) to the control signal (u) satisfies eq. (4).

....

....

....

3 Software tool

This lab is run in MATLAB. The files needed can be found on the course homepage. A graphical design tool will be used to design the weights and compute the resulting controller. There is also a Simulink model used for simulations. How to use the design tool and do the simulations will be described in this section.

3.1 H_∞ graphical design tool

Before the design tool can be opened a transfer function for the system must be defined. The model has to be strictly stable and proper, which means that is has at least as

many poles as zeros and all poles are in the left half plane. An example sequence of how the tool is started can be seen below.

```
s=tf('s');
G=1e4*(s+2)/(s+3)/(s+100)^2;
Hinf(G);
```

The tool should now open and look like in fig. 4.

There are three Bode plots, (1a), (1b), (1c). They show S, T , G_{wu} and the inverse of their respective weights.

To the right of each Bode there are lists of poles and zeros and some buttons, (4) in fig. 4. Notice that the poles corresponds to zeros in the inverse of the weight that is plotted and the opposite for the zeros.

The weights can be changed in two ways. One is to add and remove poles or zeros with the corresponding buttons. The other is to open a graphical editor for the poles and zero. It is done by clicking the 'Edit Pole-Zero diagram" button. A new window, seen in fig. 5, will open. In the new window different tools can be chosen from the toolbar, (1) in fig. 5.

The tools from left to right are:

Add real pole: Add real pole by clicking in the diagram.
Add complex pole pair: Add complex pole pair by clicking in the diagram.
Add real zero: Add real zero by clicking in the diagram.
Add complex zero pair: Add complex zero pair by clicking in the diagram.
Remove: Removes pole or zero by clicking on it.
Move: Move pole or zero by clicking and dragging.

The weights are on the form (6).

$$W(s) = k\frac{(s - z(1))(s - z(2))\ldots(s - z(m))}{(s - p(1))(s - p(2))\ldots(s - p(n))} \tag{6}$$

To edit the constant k, just type the new value in the gain input field (3) in fig. 4 and press enter.

The weights on T and G_{wu} can be disabled. Just press the disable button (5) in fig. 4. It can be enabled again by pressing the enable button that replaces the disable button.

The magnitude scale is automatically fitted to the weight but the frequency scale has to be set by the user. There are two ways of doing it. One is to enter a maximum and minimum value in the fields (2a) and (2b) in fig. 4 and press enter. The other option is to use the "auto-set frequency" option in the plots menu (6) in fig. 4

To compute the controller from the weights simply use the "compute controller" option in the controller menu, (6) in fig. 4. The controller will then be computed and the plots updated. Some information about the controller can be displayed in MATLAB's command window. The controller can then finally be exported to the workspace with the "export controller" option in the controller menu (6) in fig. 4. It will be saved as F in workspace. If there already exists a variable with that name it will be overwritten.
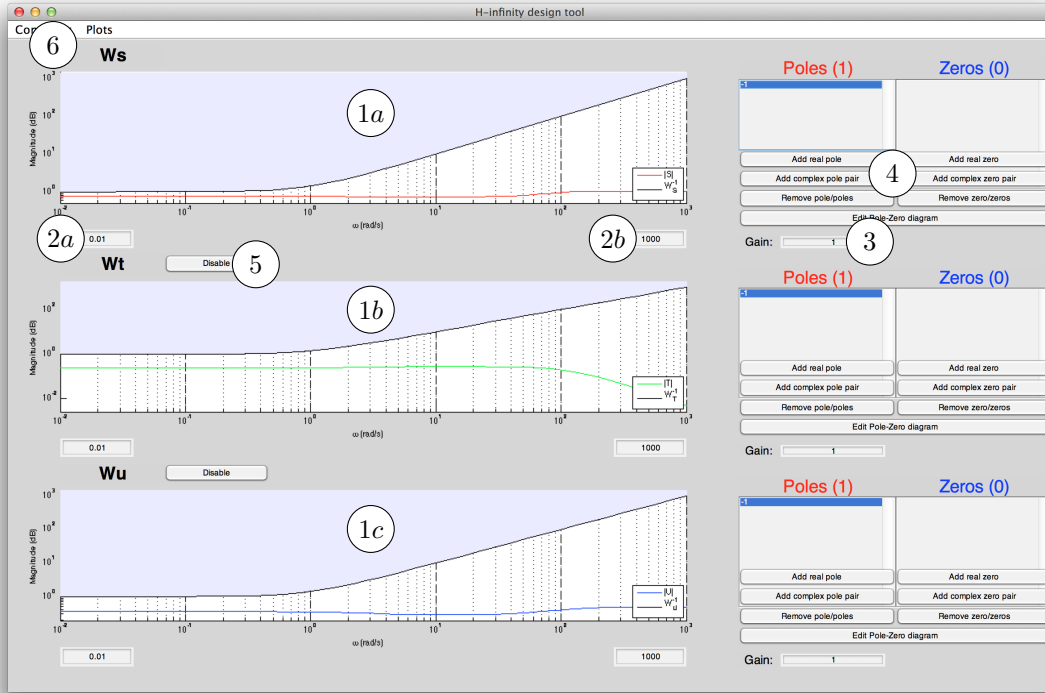
Figure 4: The H_∞ graphical design tool

3.2 Simulations

The Simulink model named `servo1.mdl` is used to simulate the system in fig. 1. A step is used as reference and disturbances can be added as band-limited white noise and a sinusoid. To run the simulation the short `macro` command, found on the homepage, can be used. In the beginning of the file `macro.m` there are some parameters that can be changed to customize the simulation. The parameters are described in the m-file.

To run the macro, the controller and system must first be saved as transfer functions named `Fsim` and `Gsim` in workspace. The macro runs the simulation and plots the results in a new figure.

Below is an example of a sequence that simulates the system. The system and controller are assumed to be defined as transfer functions in workspace with the names `F` and `G`.

```
Fsim=F; Gsim=G ;
% Edit parameters in macro.m
macro
```

Now the simulation should start and plot the results when ready. If the simulation takes very long time it can be stopped by pressing `Ctrl-C` in the MATLAB command window.
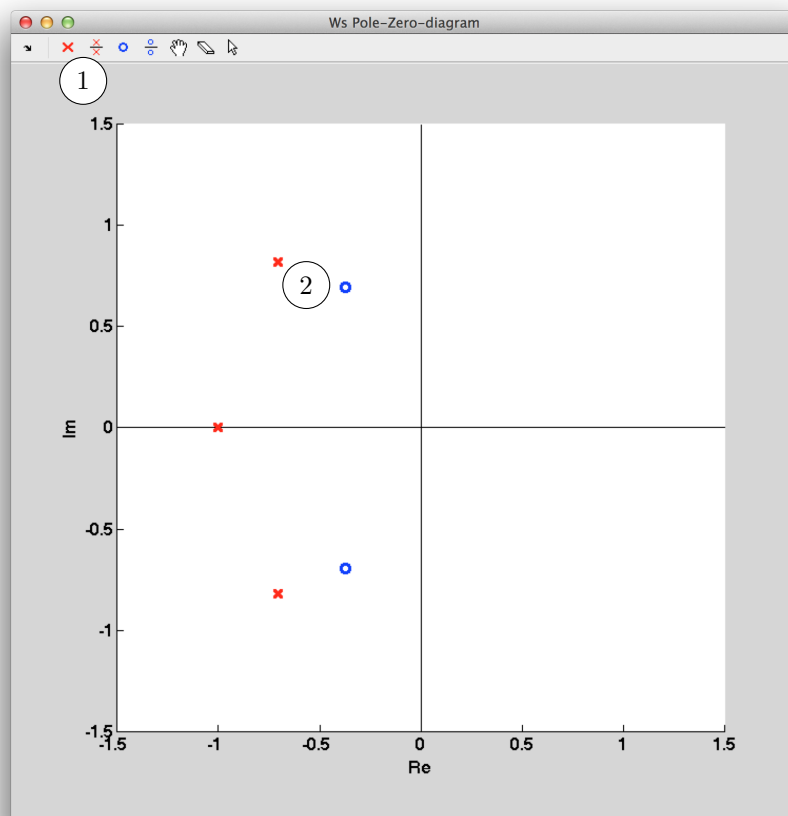
Figure 5: The pole-zero editor
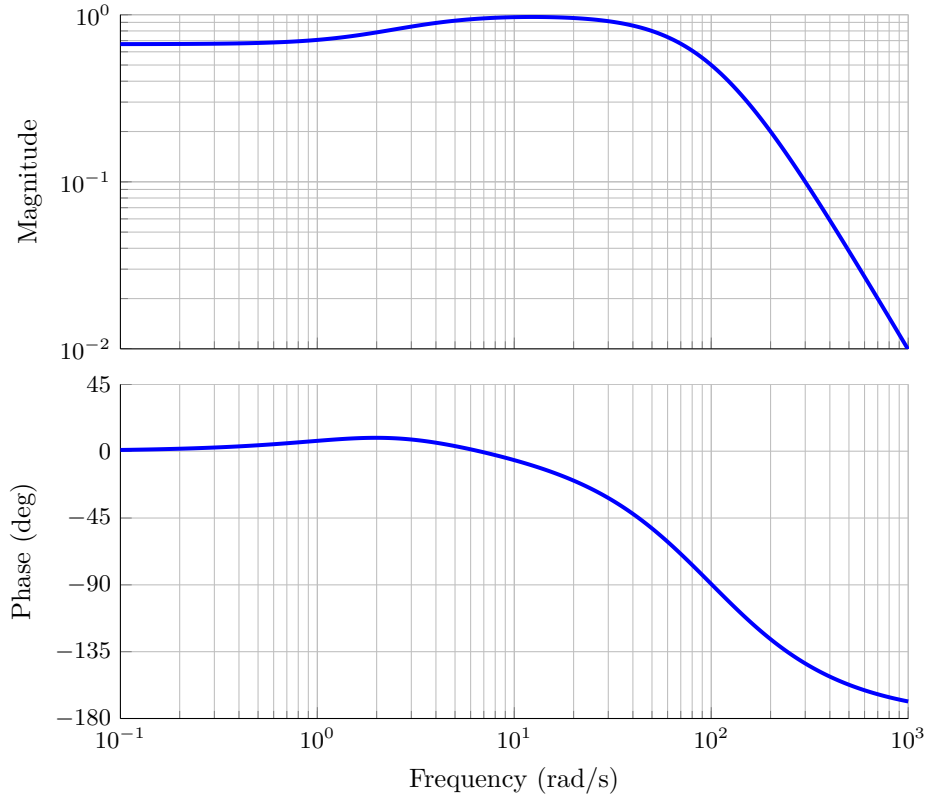
Figure 6: Bode diagram of the system in eq. (7)

4 Exercises

The system to control in this lab is an electrical device powered by the 50 Hz (100π rad/s) power grid. A proposed model of the system is given by the transfer function:

$$G(s) = 10^4 \frac{(s+2)}{(s+3)(s+100)^2} \tag{7}$$

The closed loop system is considered to be fast enough, when it comes to following the reference, without a controller, but not the suppression of disturbances. A controller will therefore be designed focusing on the disturbances.

4.1 Suppression of disturbances

i. The aim is to damp the 50 Hz disturbances. Propose a suitable weight, W_S, by drawing it in the empty Bode diagram in fig. 7. Also draw the expected resulting sensitivity function. Keep in mind that the sensitivity function can not be small for all frequencies.

ii. Now try to design the weight in the software. In this part only the sensitivity is considered. The weights on T and G_{wu} can therefore be disabled. When the
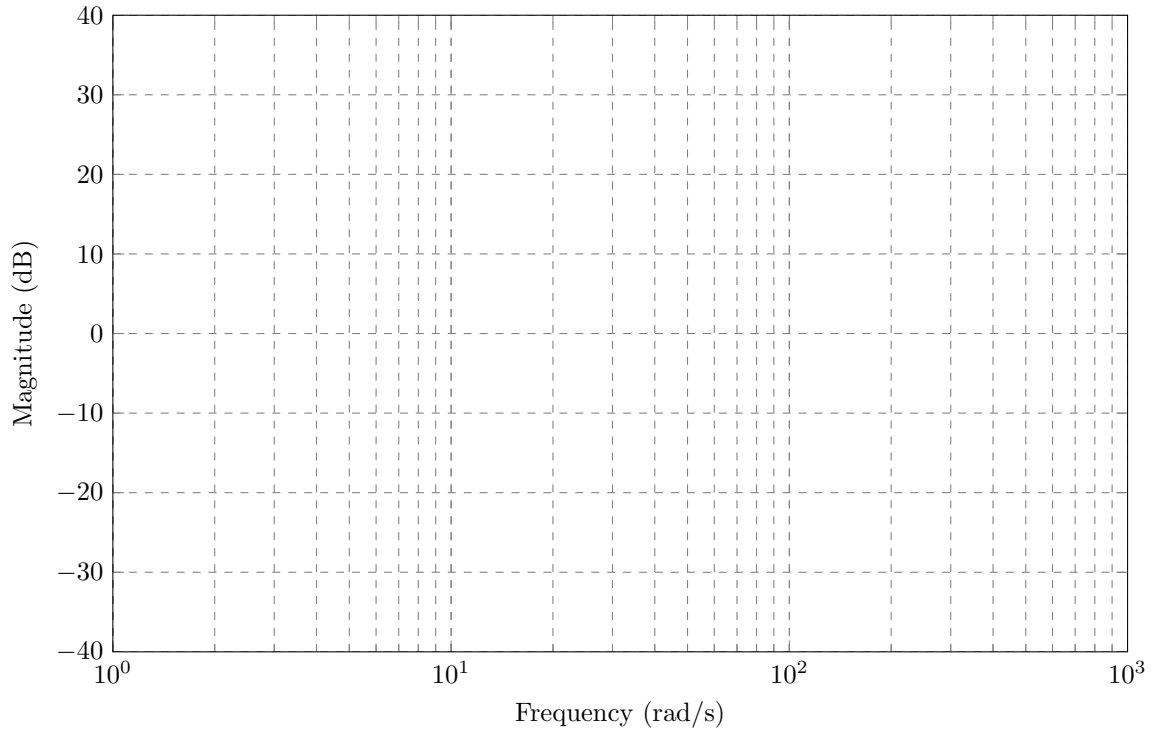
Figure 7: Draw W_S and S here.

sensitivity function is satisfactory, export the controller to workspace and run the simulation with a 50 Hz sinusoidal disturbance as described in section 3.2. Use the parameters in table 1. Fill out table 2 with the results from the simulation.

Hint: Placing poles in $s = -\epsilon \pm i\sqrt{\omega^2 - \epsilon^2}$, where ϵ is small, gives a peak at ω rad/s

Parameter	Value
u_max	∞
sin_dist_freq	100π
sin_dist_amp	1
white_noise_amp	0
step_size	0
sim_time	10

Table 1: Parameters to use in `macro.m` for simulation

Signal	Amplitude
disturbance (w)	
output (y)	
control signal (u)	

Table 2: Results from simulation in exercise 4.1 ii.

How much is the disturbance damped on the output? ....

9

(The rate between the disturbance amplitude and the output oscillations)

Approximately what amplification is required for a P-controller to get the same rate and what are the advantages/disadvantages of such a controller?

....

....

....

Hint: If $|FG| \gg 1$, $|S| \approx |FG|^{-1}$.

4.2 Robustness

Unfortunately the model (7) is not accurate. It was obtained by sending sinusoids, with different frequency, into the system and measure the output amplitude. By ignoring the phase-shift some important dynamics was not detected. The true system is given by eq. (8).

$$G_0(s) = G(s)(1 + \Delta_G(s)) = 10^4 \frac{(s+2)}{(s+3)(s+100)^2} \cdot \frac{(s-1)}{(s+2)} \tag{8}$$

i. What influences will this error have on the system behavior, and will they be a limiting factor on achievable control performance?

....

....

....

ii. Simulate the system with the controller designed in the previous exercise and the system (8). Run the simulation with the same parameters as before.

(The simulation time might need to be increased to see the results)

Comment on the results from the simulations:

....

....

....

....

....

iii. What is the condition on T required to guarantee stability for the new system, using the small gain theorem?

....

....

....

....

....

Is the condition fulfilled? *(Look at the Bode diagram in the graphical design tool.)*

....

iv. Use the software to design a new controller that suppresses 50 Hz disturbances but is stable with G_0. Keep the W_S used earlier but enable the weight on T. Then try to find a weight on T that makes the closed loop system stable and still has good suppression of the disturbance. Don't forget to compute the new controller from the menu. Export the controller to workspace and run the simulation.

Remark: The controller should be designed for G in eq. (7) but simulated with G_0 in eq. (8).

Compare the results to table 2.

....

....

....

....

....

4.3   The use of control signal

Enable the weight on G_{wu} and try to reduce the control signal. Try to reduce the amplitude to half of the one used in exercise 4.2 iv. How is the amplitude of the output affected?

....

....

....

Hint: Remember that the tool is solving the optimization problem (5). What influence does the relative gain have?

5 References

[1] Torkel Glad & Lennart Ljung: *Control theory - Multivariable and Nonlinear Methods*, Taylor & Francis (2000)

[2] Torkel Glad & Lennart Ljung: *Reglerteori - Flervariabla och olinjära metoder*, Studentlitteratur (2007)