EEE 480 LAB EXPERIMENTS

K. Tsakalis

September 5, 2002

1 Introduction

The following set of experiments aims to supplement the EEE 480 classroom instruction by providing a more detailed and hands-on experience on the analysis and design of control systems. All the experiments are simulation-based because of the flexibility, speed, and convenience offered by the modern computer software and hardware tools.

1.1 General Guidelines

- Study the background material before performing each experiment. Do not expect or rely on detailed step-by-step instructions.
- Simulations are very convenient and cheap as instruction/analysis/design tools. However, to extract the maximum benefit you should treat the simulator as an actual system. Be prepared and perform carefully planned experiments. Study the results of every simulation before attempting the next one. Be critical even for the ones that seem to work and pay attention to the details.
- Practical designs of feedback systems often involve a multitude of design parameters so that brute-force trial-and-error is prohibitive, if at all feasible. A quick analysis is often sufficient to reduce the number of experiments to a reasonable level.
- Keep a log of the experiments that you perform with any notes and computations that will allow you to recover the information contained in the results.
- When you need to ask a question, be specific but keep in mind that the person you ask may not be familiar with the details of your problem. Describe briefly what you try to do and the problem you encountered. Then ask the question. "I am working on Lab 3 and it doesn't come out." is not a specific question. "Should a_1 be 1 or 0?" is too specific. "I am working on the pendulum stabilization problem with a lead compensator. I tried several values for the gain but the closed-loop seems to be unstable. Should I change the pole or the zero of the compensator?" may be more suitable.

1.2 Lab report

After completing each lab exercise, you will need to submit a report. A template and an example of a report for Lab 1 is available at the course web page. Be concise but include all pertinent information. The limit is 6 pages for each report with 2 additional pages as an Appendix (optional).

Do not panic! You will be graded on the results and their justification. You will not be graded on presentation style or grammar. The report template should be used so that you present the results in a logical sequence. There is no lower limit on the size of the report; half a page is just as good as long as it contains all the necessary information.

The general structure of reports is outlined below. In general, there is considerable freedom in the section titles and the presentation of the material but the overall picture should be preserved.

Title, name, date Self-explanatory

Abstract A few general sentences to describe the experiment and the main results.

Introduction Introduce the reader to the problem you study. Write for an audience that is educated on the general subject but possibly unfamiliar with the specific details. At the end of this section provide a brief description of your work, results and the structure of the document.

Background or Problem formulation The specific theory you need in your analysis. Be brief and cite references for theory available in the literature. This section may also contain the formulation of a general problem in a specific mathematical framework.

Analysis or Experimental Setup Present the analytical steps that lead to the solution of the problem. Describe the simulation setup or experimental apparatus. You may include some carefully chosen block diagrams, if needed. Include any new algorithms or ideas that are important and cannot be easily deduced from the analysis.[1]

Results Describe the results that support the claims stated in the abstract, introduction and analysis. Include all pertinent details so that the results can be duplicated (simulation parameters, controller gains, initial conditions). Provide a brief discussion and logical arguments showing the agreement of the results with the theory and expectations. Point out any findings that seem important but are not explained directly from the theory. However, if your results contradict the theory and expectations, you will need to revisit the theory and make any adjustments necessary.

Discussion Any additional discussion of the results at a higher level. Issues that may require further attention but extend beyond the scope of this work. Depending on its size and importance, this section may be grouped with the Results or the Conclusions section.

Conclusions A short section to reiterate the main results and the findings.

References Complete citations according to standard formats.

Appendix Any useful additional details.

Other remarks:

- Do not pad the report to increase its volume. This has been made easy -and tempting- by today's technology but it dilutes the information you try to convey.
- Invest some time before writing the final document to decide what are the main points and what is the best way to present them. If you discover that you need an additional plot, get it.
- High-level Simulink blocks may be included in the main body -usually Background or Experimental setup- to aid the presentation, if necessary. The results should contain an adequate discussion of your observations and claims, supported by plots and figures. Do not include all the figures you have generated in the lab. Treat the report as a document that tries to convince the audience of your claims and not as a proof of work.
- Certain script files (programs) may be included in the the appendix if necessary. That is if they contain useful, nontrivial information. Keep in mind that you may always be asked to supply additional information (programs/models) if needed. So, save your work.
- Figures should be sufficiently clear, labeled, and unambiguous. Discuss the results shown in the figures. Statements of the form "The simulated responses are shown in Fig. 1" is usually unacceptable. Do not leave the interpretation up to the reader unless it is clear from the context. Instead you should write something like "The simulated responses of the system with controller gains 5,6,7 are shown in Fig. 1. These results illustrate that higher controller gains improve the performance in terms of disturbance attenuation at steady-state, at the expense of higher overshoots."

[1] This depends on the intended audience. Also, try to avoid footnotes!

- Remember that these are only general guidelines and you should exercise common sense. Your reports may not and need not be perfect immediately. However, by the end of the semester you should be used to writing in a brief, precise, and informative technical style.

2 LAB 1

2.1 Scope

The objective of this Lab is to familiarize the student with the use of MATLAB/SIMULINK as a general and versatile platform for analysis and design of control systems. (See Lab 1 sample report for details.)

2.2 Assignment

1. **Using help:** Certainly the most useful command. `help command_name` displays the syntax, usage, and brief comments about the command `command_name`. Use it for plot, title, ylabel, bode, step, series, feedback, ss, tf, roots, eig, residue, and any other command you encounter.

2. **Vectors, Matrices and Plots:** Plot the function $y(t) = t + 0.01t^3 + \cos t$ in the interval [-10,10].

   Generate a vector of points for the independent variable by `t=[-10:0.1:10]';` and the corresponding values of y by `y=t+0.01*t.^3+ cos(t)`. Then `plot(t,y)`.

   Next, consider the "straight-line approximation" problem $y(t) \simeq at + b$. One possible approximation can be computed by minimizing the sum of the square errors $\sum_i |y(t_i) - at_i - b|^2$ with respect to the parameters a, b. Compute the best least-squares, straight-line fit in the same interval.

   Define the regressor vector W as `W=[t,ones(size(t))];` and compute the least-squares optimal parameters by `ab=W\y;` (ab is vector containing the two parameters a and b.) The least-squares straight-line approximation of $y(t)$, can now be computed as `yh=W*ab;`. Alternatively, `yh=ab(1)*t+ab(2);`

   Compare the original function with its straight-line approximation. E.g., `plot(t,y,t,yh)`

3. **Working with Figures:** Present the results of Experiment 2 in one figure containing two plots with titles and axis labels. The plots should be side-by-side and one should show the function and its approximation and the other should show the difference. Copy and paste the figure in a Word or Powerpoint document.

4. **System Responses:** Use the `bode` and `step` commands to generate the frequency and step responses of the systems with transfer functions

   $$G_1(s) = \frac{1}{5s+1}, \quad G_2(s) = \frac{1}{s^2+0.3s+1}$$

   Note: `bode` generates plots of the magnitude and phase of a transfer function ($|G(jw)|, \arg[G(jw)]$) as functions of the frequency w. Use `g1=tf(1,[5,1])` to define the first transfer function etc.

5. **System responses with Simulink:** Use Simulink to generate the step responses of the systems defined in Experiment 4.

6. **Composite Systems:** Compute the step response of a cascade (series) connection of the systems defined in Experiment 4. Also compute the step response of the feedback system

   $$y(s) = G_2(s)G_1(s)e, \quad e = r - y$$

7. **State-space representations:** Generate state-space representations of the systems defined in Experiments 4 and 6. Use the `ss` command to obtain a state-space representation of a transfer function. E.g., `g1ss = ss(g1)`. Verify that the two systems (g1, g1ss) are the same by comparing their frequency responses.

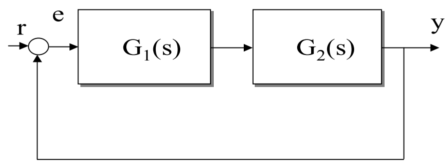

Figure 1: Block diagram of the feedback system in Experiment 6.

8. **Roots of polynomials, matrix eigenvalues, and partial fraction expansion** Find the roots of the denominators of the two systems in Experiment 6. Compute the corresponding eigenvalues of the system matrices (A). And compute the partial fraction expansion for $G_2(s)G_1(s)$. The pertinent commands are `roots, eig, residue`. (Note that the same results can also be obtained by using other properties of the system data structures.) The "residue" function should not be used with transfer functions that have multiple poles.

9. **Nonlinear differential equations with Simulink:** Use Simulink to compute the unit step response of the system described by the nonlinear differential equation:

$$\ddot{y}(t) = (1 - y^2(t))\dot{y}(t) - \mathrm{sat}_{[-1,1]}[y(t)] + 0.6u(t), \quad y(0) = \dot{y}(0) = 0$$

   where $\mathrm{sat}_{[a,b]}[x]$ is the saturation of x in the interval $[a, b]$, i.e., it is a if $x \leq a$, b if $x \geq b$ and x if $a < x < b$.

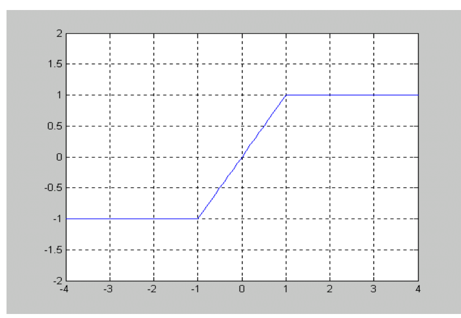

Figure 2: The saturation function in Experiment 9.

10. **Explore the use of a new command and a new Simulink block:** Use the on-line help to find a new command, possibly related to the ones you have already used, and apply it in a suitable way. Do the same with a new block in the Simulink library.

```
% Script for Lab 1, Execute by typing its name in the Matlab command window.
% Written in Matlab V6 (Release 12)
% KST, 8/24/01

disp('*** Experiment 1 ***')
eval('help tf') % for use inside functions
help tfdata     % or simply...

disp('*** Experiment 2 ***')
t=[-10:.1:10]';        % create a time vector (column convention)
y=t+0.01*t.^3+cos(t); % Eval the function
```

```
w=[t 0*t+1];          % form the regressor vector
ab=w\y                % Compute the LS solution; same as inv(w'*w)*w'*y
yh=w*ab;              % Eval the straight-line approximation; same as ab(1)*t+ab(2)
subplot(121)          % Multi-plot figures
plot(t,y,t,yh)        % plot multiple lines
ylabel('y(t),  y_{LS} (t)')    % add axis labels
xlabel('t, in default time units (DTU)')
title('LS straight line approximation')
subplot(122)          % next plot
plot(t,y-yh)
ylabel('LS error')
xlabel('t (DTU)')
title('Fitting error')
disp(' When done hit <space> to continue')
pause                 % pause to examine the figure; resume with <space>

disp('*** Experiment 3 ***')
disp('On the figure window click <Edit/Copy Options> and select Bitmap format...')
disp('  then click <Edit/Copy Figure>, go to your Word document and <paste>...')
disp('  In the document, right click on the figure, select <format.../size>,...')
disp('  and resize the figure as necessary. Keep the <Lock aspect ratio> box checked.')
disp(' ')
disp(' When done hit <space> to continue')
pause

disp('*** Experiment 4 ***')
clf                   % clear figure/reset
G1=tf(1,[5 1]);       % create a transfer function object
G2=tf(1,[1 .3 1]);
disp('Bode plot 1')
bode(G1);pause        % Generate the Bode plot (auto-axis, generic labels)
disp('Bode plot 2')
bode(G2);pause
disp('Step 1')
step(G1);pause        % Generate the unit step response
disp('Step 2')
step(G2);pause

disp('*** Experiment 5 ***')
simulink               % Start Simulink ...
disp('*** Work on the Simulink window... ***')
disp('*** Experiment 6 ***')
disp('G_2(s)G_1(s)')
G21=series(G1,G2)      % Create the cascade connection
disp('Step of G_2(s)G_1(s)')
step(G21);pause
disp('another computation of the same thing')
step(G2*G1,'r'); pause  % Also works like this; *,/,+,- are overlayed operations
                        % with their systems-interpretation when the objects are systems.
                        % The 'r' is to change the display color
disp('The unity-feedback loop of G_2(s)G_1(s)')
Gf=feedback(G21,1)      % Now the feedback
```

```
step(Gf);pause
disp('another computation of the same thing')
step(G2*G1/(1+G2*G1),'m');pause   % just as well...
                                  % but not a good way: it fails for RHP modes
disp('*** Experiment 7 ***')
G21ss=ss(G21);
Gfss=ss(Gf);            % Much easier than the old ss2tf or tf2ss
                        % You could also define a system in state-space by G=ss(A,B,C,D)
disp('A and C -matrices  for the feedback system')
Gfss.a                  % display the A-matrix of the system Gfss
Gfss.c                  % etc
pause

disp('*** Experiment 8 ***')
% Now we need the numerators/denominators and A-matrices of G21 and Gf
[num21,den21] = tfdata(G21,'v');   % 'v' to get the result in vectors
[numf,denf] = tfdata(Gf,'v');
a21 = G21ss.a;
af = Gfss.a;
disp('poles of G21')
roots(den21), pause      % you can use , for displaying results in in-line commands
disp('eigenvalues of A21')
eig(a21), pause
disp('partial fraction expansion of G_2(s)G_1(s)')
[r,p,k]=residue(num21,den21);
disp('residues'),r,pause
disp('poles'),p,pause
disp('feed-through (if any)'),k,pause


disp('*** Experiments 9-10 ***')
disp('... on the Simulink window... ')
disp('END OF DEMO! You should know enough by now...')
```

3 LAB 2

3.1 Scope

The objective of this Lab is to study the time and frequency responses of simple systems and obtain insight on the key response indicators.

3.2 Assignment

1. Normalization: In this type of study, it is convenient to normalize the system transfer functions and eliminate the degrees of freedom that have an easily understood effect on the system response. Such normalizations are output scaling and time scaling that correspond to changes in the units of the output and time.

#1: Show that after output and time scaling, the transfer function of a stable first-order system can be written as:

$$K\frac{s+b}{s+a} \longrightarrow \frac{\tau_z s+1}{s+1},\ \text{or}\ \frac{s+z}{s+1}$$

The first is applicable to systems that have non-zero DC-gain, and the second to systems that have non-zero direct throughput.

Hint: Follow the operations:

$$y = K\frac{s+b}{s+a}u = \left(K\frac{b}{a}\right)\frac{(b^{-1})s+1}{(a^{-1})s+1}u = \left(\frac{Kb}{a}\right)\frac{(b^{-1}a)(a^{-1}s)+1}{(a^{-1}s)+1}u$$

Define $\bar{y} = (a/Kb)y$ and $\bar{t} = at$ so that $\bar{s} = a^{-1}s$, etc.

#2: Similarly, stable second-order systems with no direct throughput can be written as

$$K\frac{s+b}{s^2+a_1s+a_0} \longrightarrow \frac{\tau_z s+1}{s^2+2\zeta s+1}, \text{ or } \frac{s+z}{s^2+2\zeta s+1}$$

2. First-order systems have particularly simple responses. Stable and minimum-phase systems can exhibit lead or lag response depending on the sign of the phase of the frequency response. Non-minimum-phase systems (with RHP zeros) exhibit inverse response. Study the effect of the transfer function zero $(1/\tau_z)$ on the frequency response and step response of the system.

#3: Plot the step repsonses, frequency response magnitudes and phases for three representative cases of τ_z that illustrate the different types of response.

Remarks:

1. Critical points for τ_z are 0 and 1.
2. The step response may be discontinuous at 0.
3. The limit values of the step response at $0^+, \infty$ can be computed using the Laplace limit theorems.

3. Second-order systems may exhibit overdamped or underdamped response depending on the value of the damping ratio ζ. The overshoot of the step response and the peak-magnitude in the frequency response are amplified by zeros near the system bandwidth. Non-minimum-phase zeros result in inverse response. Study the effect of the transfer function zero $(1/\tau_z)$ and the damping ratio ζ on the frequency response and step response of the system.

While there are many useful ways to view the results, here we focus on bandwidth and overshoot:

#4: Let $\zeta = 0.5$ and τ_z range in the interval [-10 , 10] and plot the percent-overshoot and bandwidth as functions of τ_z (use a numerical computation of these quantities).

#5: Repeat for $\zeta = 1$.

Remark: It is convenient to write a script file to perform the computations. This type of plots is useful in controller design where the selection of the dominant pole-pair should be adjusted to reflect the effect of the zero.

4. Third-order systems may exhibit even more complicated behavior. Here, we are interested in systems that can be written as a cascade of a complex conjugate pole-pair and a first order lead or lag transfer function. Such systems arise often in simple compensator design problems where the dominant closed-loop dynamics are second order with the addition of a pole and a zero near each other and within the bandwidth of the dominant pair.

In such cases the system response is primarily dictated by the dominant pole-pair with a relatively small perturbation caused In general, first order lead elements amplify the overshoot of the step response and the peak-magnitude in the frequency response; on the other hand, first order lag elements attenuate both of these measures. The amount of amplification or attenuation depends on the distance between the pole and the zero and their relative location with respect to the dominant pair. Study the effect of the pole and zero of thr first-order element and the dominant pair damping ratio on the frequency response and step response of the system.

Again, there are many useful ways to view the results. One possibility is to fix ζ and τ_z and plot the overshoot (or any other measure of interest) as a function of the "distance" between the pole and the zero.

#6: For $\zeta = 0.5$ generate a plot showing the overshoot as a function of τ_p/τ_z, for a different values of τ_z. Select a few values of τ_z in the interval $[0.2, 10]$ and τ_p/τ_z in the interval $[0.7, 1.4]$.

#7: Based on this study, determine the pole locations p such that the transfer function

$$\frac{s + 0.3}{(s + p)(s^2 + 0.894s + 0.8)}$$

exhibits less than 20% overshoot. Verify with a simulation.

Sample Matlab Script

```
% Script file for the study of overshoot of the special
% 3rd order system as a function of the lead/lag element

zeta=0.5;
ip=[.7:.05:1.4];
iz=[.2  .5  1  1.5 2 3 5 7 10];
po=0*ip'*iz;

for iiz=1:length(iz)
    tauz=iz(iiz)
    for iip=1:length(ip)
        taup=tauz*ip(iip);
        y=step([tauz 1],conv([1 2*zeta 1],[taup 1]));
        poi=max(y)-1;if poi<0;poi=0;end
        po(iip,iiz)=poi;
    end
end
plot(ip,po);title('Overshoot vs. zero/pole ratio for \zeta = 0.5')
hold on;
for i=1:length(iz)
    text(.65,po(1,i),num2str(iz(i)))
end
hold off
```

4 LAB 3

4.1 Scope

The objective of this Lab is to study the concepts of system approximation, model order reduction, and model uncertainty in the context of feedback control.

4.2 Generalities

The basic theoretical framework of system approximation can be found in the class notes. [2] Its main points are summarized below:

System Gain: The gain of a system provides an important measure of its "size," that is, its distance from the zero-system. System gains can be defined in terms of the underlying metric that is used to measure signals.[3] A particularly useful and important gain is defined by the maximum amplification of the square-root of the

[2] 'On the notion of the "size" of a system and its applications.'

[3] These are usually referred to as "induced gains;" more specific terminology is used to distinguish different types of gains, if not clear from the context. This notion of gain (a distance) should not be confused with the "DC-gain" ($G(0)$, a number) or the "loop-gain" ($G(s)H(s)$, a transfer function).

input signal energy (or RMS-value). It turns out that, for a stable system, this maximum amplification is equal to the peak magnitude of the frequency response of the system transfer function: For a stable system with transfer function $G(s)$,

$$\gamma_2[G] = \max_{w} |G(jw)|$$

System Approximation: Using the system gain as a distance, we can quantify the approximation of a system G by another G_0 as the gain of the error system $\gamma_2[G - G_0]$.

Model Order Reduction: This is a type of system approximation where we seek to simplify the system by reducing its order. The order reduction problem can be viewed as the elimination of terms from the partial fraction expansion that have small contributions. A computational reduction procedure (not necessarily efficient) is to rank each PFE term according to their gain and eliminate those below a given threshold.

Uncertainty: In practice, one is often faced with an approximation problem where a single system (usually called "nominal") is used to approximate all the elements of a set of systems. For example, consider the case where the model parameters are known within a tolerance, or the system model changes depending on the operating conditions. In such cases, the term uncertainty is used to signify a modeling error (difference between the actual system and the nominal) that can take different values depending on the system.

The uncertainty is itself a dynamical system but its description (order, parameters) is not known a priori. One reasonable expectation is that an estimate of its maximum gain is available during the controller design. Uncertainty estimates can be produced by computing the gains of all possible error systems and taking the maximum. Experimentally, an estimate can be computed from the spectral properties of the response difference.

In this framework, the statement of the design problem should be adjusted accordingly to reflect the presence of uncertainty. For example,

Robust Stability problem: Achieve the objectives for the nominal system and maintain stability for all possible values of the uncertainty within a prescribed maximum gain.

Robust Performance problem: Achieve the objectives for all possible systems with values of the uncertainty within a prescribed maximum gain.

The classical notions of gain and phase margin are special cases of the robust stability problem.

Weighted Approximations: The characterization of the uncertainty or modeling error by a single number is convenient and consistent with the theoretical framework. However, it may lead to conservative results if the modeling error contains additional structure. An important example of this is a modeling error or uncertainty whose frequency response has a frequency-dependent maximum magnitude. Alternatively, in closed-loop systems, variations in the loop transfer function have a different effect depending on their frequency range. That is, feedback attenuates certain portions of the system uncertainty while it amplifies others.

In an effort to reduce the conservatism in the design of a feedback system, a frequency-dependent weight can be added in the description of the modeling error to reflect the importance of different frequencies. An example of such an application is the weighted model reduction where the reduced system G_0 is chosen to minimize $\gamma_2[W(G-G_0)]$. For a given W, this problem can be solved using the PFE of G and ranking each term according to its weighted gain.

The quantity $\gamma_2[W(G - G_0)]$ preserves its distance interpretation between the two systems G and G_0, as long as W is a stable, minimum-phase, invertible system.[4] The role of the weight is to emphasize the differences between G and G_0 in the frequency range where W has itself a large magnitude and ignore the differences in the frequencies where W is small. (Usually, the choice of the weighting function is dictated by the problem.)

4.3 Assignment

1. Basic Computations: Compute the gains of the systems with transfer functions

$$\frac{1}{s^2+s+1}\,,\quad \frac{2}{s+100}\,,\quad \frac{1}{s^2+0.1s+100}\,,\quad \frac{s+2}{s^2+s+1}\,,\quad \frac{s-2}{s^2+s+1}$$

Hint: Define the transfer functions in the workspace, use the bode command to generate the frequency response and then find the maximum by clicking on the line.

[4] $\gamma_2[W]$ and $\gamma_2[W^{-1}]$ are both finite.

2. **Simple Model Reduction:** Find a first order approximation of the second-order heat transfer model in Section 4 of the class notes *Examples of System Models*.

   Hint: Convert the given state-space description to a transfer function; use the `tfdata` command to obtain the numerator and denominator; use the `residue` command to generate the PFE and then compute the frequency response of each term.

3. **Model Reduction for Feedback Control:** Given a heat transfer process model

$$G(s) = \frac{1000}{(s+0.1)(s+1)(s+10)(s+50)}$$

   we seek a reduced order model that is suitable for the design of a controller such that:

   - The closed-loop bandwidth is 2.
   - The closed-loop bandwidth is 15.

   Hint: Roughly, the robust stability constraint is that the closed-loop bandwidth should be less than gain crossover frequency of the multiplicative uncertainty/modeling error. A additional safety margin can/should be included as a quick approach to obtain a form of robust performance. For example, instead of the unity gain crossover, the bandwidth constraint may be selected as the m-gain crossover frequency of the uncertainty, where $m < 1$ is a safety parameter (say, $m \sim 0.3$).

   To implement this idea:

   1. Generate the PFE and rank the terms according to their gain.
   2. Group the terms into a nominal transfer function, say G_0, and an uncertainty/modeling error, say Δ.
   3. Compute the frequency response of the multiplicative uncertainty $\Delta_m = \Delta/G_0$ and find the crossover frequency (note:$0.3 \simeq -10dB$).
   4. Repeat for different orders of G_0 (1,2,3).

4. **Justification of Model Reduction for Controler Design:** For the process model of # 3, a PID controller has been designed to achieve approximately a bandwidth of 2 for a second-order nominal system and have reasonable sensitivity properties. The controller transfer function is

$$C(s) = 3.27 + \frac{2.67}{s} + \frac{1.53s}{0.01s+1}$$

   Investigate the validity of the assumption that the controller can be designed for the nominal (reduced-order) system and then applied to the full-order system.

   - Find the closed-loop transfer functions from the reference to the output for:
     1. The second-order nominal plant with the PID controller.
     2. The full-order plant with the PID controller.
   - Compare the two transfer functions by computing the gain of their difference.
   - Visualize the difference by comparing the step and frequency responses of the two closed-loop transfer functions (use either Matlab or Simulink for this).
   - Verify that their difference is small for low frequency reference inputs by computing the gain of their difference weighted by a low-pass weight, e.g., $W(s) = (0.1s+1)/(2s+1)$.

5. **A Different Example:** In the design of feedback controllers for mechanical systems, a rigid body assumption is often employed in the modeling of the system. This assumption essentially amounts to the reduction of the full-order system by eliminating the flexible modes. For such cases, it is well known that the controller design will be successful provided that the closed-loop bandwidth is well-below the bandwidth of the flexible modes.

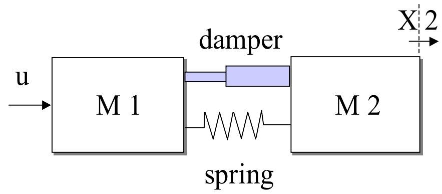

Figure 3: The two-mass system in Experiment 5.

An example to demonstrate the main issues of this class of problems is the position control of a two mass system. The two masses are linked with a spring and a damper; a force (control input) is applied on the first one. We would like to control the position of either one of the two masses, assuming that a measurement of that position is available. The first-principles model of the two mass system is

$$
\begin{aligned}
m_1\ddot{x}_1 &= u - b_1\dot{x}_1 + k(x_2 - x_1) + b(\dot{x}_2 - \dot{x}_1) \\
m_2\ddot{x}_2 &= -b_2\dot{x}_2 - k(x_2 - x_1) - b(\dot{x}_2 - \dot{x}_1)
\end{aligned}
$$

where x_i are the mass positions, m_i are the masses, b_i are the friction coefficients, k is the spring constant, and b is the damper constant. On the other hand, the rigid body approximation is

$$(m_1 + m_2)\ddot{x} = u - (b_1 + b_2)\dot{x}$$

The rigid body equation can be used to design a controller for either one of the two positions, as long as the flexible modes are not excited (sufficiently low bandwidth). If the controller bandwidth is too high then the closed-loop response can deviate considerably from the nominal one and the loop may even become unstable. The multiplicative modeling error bounds provide design constraints[5] to guarantee that:

1. the nominal design does not destabilize the full order system;
2. the full order closed-loop response is close to the nominal one.

Notice that it is not necessary that the violation of the constraints will result in an unstable closed-loop system. However, its behavior is unpredictable from the reduced-order data alone.

Use the following script file to investigate and comment on these statements.

```
% Script file for the 2-mass system problem
%
% Define the problem parameters
m1=1;b1=.1;
k=10;b=.1;
m2=.5;b2=.1;
w=logspace(-3,2,300);
%
% Define the controllers (tuned for the rigid approx.)
PID1 =[  6.1954e-001  1.2065e-001  1.4504e-002
  1.0000e+000  1.0000e+000            0
  1.0615e-001  1.4504e-002  5.1340e-001];
PID2 =[  1.6442e+000  6.3475e-001  1.9407e-001
  4.0000e-001  1.0000e+000            0
```

[5] Here, the resonance creates sharp variations in the frequency response and the simplified version in terms of the crossover frequency alone may fail; instead the full robust stability criterion should be used.

```
  5.5712e-001  1.9407e-001  1.4213e+000];
PID3 =[  3.4173e+000  2.4201e+000  1.4943e+000
  2.0000e-001  1.0000e+000           0
  2.1212e+000  1.4943e+000  2.9931e+000];
%
% Set up the system in state space and compute transfer functions to positions
A=[0 1 0 0;[-k -b-b1 k b]/m1;0 0 0 1;[k b -k -b-b2]/m2];
B=[0; 1/m1; 0; 0]; C1=[1 0 0 0]; C2=[0 0 1 0]; D=0;
gf1ss=ss(A,B,C1,D); gf2ss=ss(A,B,C2,D); gf1=tf(gf1ss); gf2=tf(gf2ss);
% The rigid body approximation is the same for both
gr=tf(1,[m1+m2 b1+b2 0]);
% Compare the frequency and impulse responses of the two models and
%   the rigid body approximation
bode(gr,gf1,gf2)
disp('Freq. response of the two models and the rigid body approximation');pause
impulse(gr,gf1,gf2)
disp('Impulse response of the two models and the rigid body approximation');pause
%
% Generate plots of the inverse multiplicative for each case
% The magnitude is the upper bound for the complementary sensitivity (T)
bode(gr/(gf1-gr),gr/(gf2-gr),w);
disp('Inverse Multiplicative Uncertainty (T constraints)');pause
%
% Evaluate the designs for the control of the first or the second mass displacement
% 1st controller, tuned with the rigid model for a BW ~0.6
gpid=tf(PID1(1,:),PID1(2,:));
bode(gr/(gf1-gr),gr/(gf2-gr),feedback(gpid*gr,1),w);
disp('Low BW controller, constraints and nominal T (plenty of margin)');pause
step(feedback(gpid*gr,1),feedback(gpid*gf1,1),feedback(gpid*gf2,1));
disp('Low BW controller, step responses (very predictable)');pause
%
% 2nd controller, tuned with the rigid model for a BW ~1.7
gpid=tf(PID2(1,:),PID2(2,:));
bode(gr/(gf1-gr),gr/(gf2-gr),feedback(gpid*gr,1),w);
disp('Medium BW controller, constraints and nominal T (virtually no margin)');pause
step(feedback(gpid*gr,1),feedback(gpid*gf1,1),feedback(gpid*gf2,1));
disp('Medium BW controller, step responses (excitation of the flexible modes)');pause
%
% 3rd controller, tuned with the rigid model for a BW ~3.5
gpid=tf(PID3(1,:),PID3(2,:));
bode(gr/(gf1-gr),gr/(gf2-gr),feedback(gpid*gr,1),w);
disp('Hi BW controller, constraints and nominal T (constraints are violated)');pause
t=[0:.01:5]';
step(feedback(gpid*gr,1),feedback(gpid*gf1,1),feedback(gpid*gf2,1),t);
disp('Hi BW controller, step responses (instability of the 2nd mass loop)');
```

5 LAB 4

5.1 Scope

The objective of this Lab is to study a simple compensator design problem using Root-Locus techniques.

5.2 Assignment

Consider the car cruise control problem discussed in the class notes *Examples of System Models*, linearized around the 55 mph steady-state. Use root-locus techniques to design a compensator for this system that meets the following specifications:

- Zero steady-state error to constant disturbances.
- Overshoot less than 25%.
- 2%-Settling time less than 15 s.
- Closed-loop bandwidth $BW \leq 1.5$ rad/s (constraint due to neglected lags, such as engine dynamics).
- "Small" steady-state error to ramp reference inputs.

Hint: First, convert (or at least interpret) the specs in terms of Root-Locus relevant quantities. The zero steady-state error to steps implies a compensator pole at the origin. Keeping things simple, consider a PI-type compensator ($k(s+a)/s$). Avoid using a double pole at the origin to meet the ramp objective. (Why?) Instead translate the objective in terms of k, a. Next, decide on the pole locations for the dominant pole pair to satisfy the specs. Here, the entire closed-loop is only second-order but keep in mind that the compensator will contribute a zero to the closed-loop transfer function. Once you have a compensator that solves the problem (at least approximately), you may improve the design by a search over a reasonably small interval of the compensator parameters.

6 LAB 5

6.1 Scope

The objective of this Lab is to study a simple compensator design problem using frequency-domain (Nyquist/Bode) techniques.

6.2 Assignment

Consider the car cruise control problem discussed in the class notes *Examples of System Models*, linearized around the 55 mph steady-state. Use Nyquist/Bode techniques to design a compensator for this system that meets the following specifications:

- Zero steady-state error to constant disturbances.
- Phase margin greater than 65 degrees
- Closed-loop bandwidth $BW \leq 1.5$ rad/s (constraint due to neglected lags, such as engine dynamics).
- Reasonable settling time and overshoot.
- "Small" speed fluctuations for low frequency disturbances (rolling hills). Provide an estimate of the speed fluctuations for various reasonable cases.

Hint: First, convert (or at least interpret) the specs in terms of frequency response relevant quantities. The zero steady-state error to steps implies a compensator pole at the origin (PI compensator ($k(s+a)/s$). This is not the usual lag compensator and the standard design procedure needs to be adjusted. Once you have a compensator that solves the problem, you may improve the design by a search over a reasonably small interval of the compensator parameters.

7 LAB 6

7.1 Scope

The objective of this Lab is to perform a more demanding controller design that combines lead and lag compensation.

7.2 Assignment

Consider the torque-pendulum control problem discussed in Section 5 of the class notes *Examples of System Models*. We would like to design a compensator for the linearized approximation of this system that meets the following specifications:

- Steady-state attenuation of constant disturbances $\leq -40dB$. "Good" attenuation of other low frequency disturbances.
- Closed-loop bandwidth $BW \leq 20$ rad/s.
- Reasonable Sensitivity and Complementary Sensitivity peaks.
- Reasonable settling time and damping ratio.
- Small step-response overshoot (5-10%).

Remarks: You may use frequency domain or root-locus techniques or a combination of the two. The compensator may contain an integrator to meet the first specification and it must contain a significant lead element to stabilize the system. In this problem, the lag element causes a necessary peaking of the complementary sensitivity. A reasonable choice for crossover is around 5 rad/s, while the S and T peaks should be less than 1.2 and 2, respectively. Also, the dominant poles should have a fairly high damping ratio (0.7 or more) to account for the lag zero and a reasonable settling time is around 2 s. Finally, the step response overshoot can be adjusted by a prefilter or a suitable cascade-feedback (2-DOF) decomposition of the basic lead-lag compensator.

8 LAB 7

8.1 Scope

The objective of this Lab is to address more complicated and realistic problems that arise in control systems design, including the effect of control input saturation and the design of cascade control structures.

8.2 Introduction

The first problem (input saturation) is present in almost every control system and its severity depends on the type of the plant. When operating beyond the saturation limit, special modifications are required to avoid the so-called integrator wind-up. This phenomenon occurs when the controller requests a large input signal to reduce the error but, due to saturation, a much smaller input signal enters the plant. As a result, the controller sees a smaller-than-expected decrease in the error and increases the requested control input even more. This "wind-up" of the controller states can lead to undesirable behavior or even instability. Controllers with integral action or very slow poles are susceptible to wind-up problems. A simple (though not always successful) remedy for that is to use limited integrators that retard the error integration when the control input saturates.

The second problem (cascade control systems) appears frequently in industrial and large-scale control systems where nested loops are often used to maintain system integrity in the case of failures. They also enable partially manual control of certain variables while others are controlled by local feedback loops. In general, nested (cascade) loops cause a deterioration in the achievable performance but they allow for simpler controller tuning strategies and easier maintenance of the control loops.

8.3 Assignment 1.

Design a compensator for the reduced-order model of the tube temperature system (Section 4 of the class notes *Examples of System Models*) that meets the following specifications:

- Good attenuation of low frequency disturbances (below 0.5 rad/min).

- Closed-loop bandwidth around 6 rad/min.
- Reasonable Sensitivity and Complementary Sensitivity peaks (around 1.5).
- Fast settling (in the order of a few minutes).
- Negligible overshoot to ramps (10 deg/min, possibly with a prefilter/2-DOF implementation).

The effectiveness of the control should be demonstrated on a Simulink model that includes control input saturation.

Remarks: In contrast to an actual system, many controller tunings will work for the simplified model. You should design a controller that, above all else, should obey the bandwidth limitation. You should also simulate such a controller without any anti-windup modifications to gain an appreciation for the wind-up problem. In practice, to avoid input saturation and maintain a controlled operation at all times, temperature commands are issued as ramps. Still, saturation occurs during transients, start-up operations, and with large disturbances. For this reason, the controller should be able to tolerate control input saturation, possibly for extended time periods.

8.4 Assignment 2.

Design a controller for the cart-pendulum system presented in Section 6 of the class notes *Examples of System Models*, to meet the following specifications:

- Asymptotic tracking of step commands at the cart-position.
- Asymptotic convergence of the pendulum angle to the inverted position when the reference input is constant.
- Reasonably fast settling.

Remarks: This is a fairly hard problem, hence the lack of detailed specs. That should not be interpreted as great flexibility in the controller design. The controller should have enough bandwidth to stabilize the pendulum while cart movements should respect the corresponding RHP zero. Control input saturation does not allow the use of very fast controllers and keeping the pendulum inverted requires low sensitivity peaks.

The first issue in the controller design for such a system is the selection of the control structure. Since the system has two outputs and one input, a cascade controller is a reasonable choice. The inner loop should be controlling the pendulum angle, stabilizing it in the inverted position. It should achieve that fairly fast (otherwise the cart runs out of the track) and without excessive overshoot. In addition to the pendulum instability, the plant (cart force to pendulum angle) has a zero at the origin that makes the controller design more difficult. After closing the inner loop, the outer loop should control the cart position by issuing commands to the inner loop. The corresponding model (angle reference input to cart position) has a RHP zero and a double pole at the origin. It can be stabilized with even small amounts of lead compensation but achieving good sensitivity properties requires a more careful design. [6]

Finally, it goes without saying that the range of operation of the cart-and-pendulum is fairly narrow. The controller should not be expected to handle large deviations in the pendulum angle, and all test inputs should reflect that.

[6] For more details on appropriate specs for each loop you can look at the response of the controller included with the Simulink model of the cart-and-pendulum, `http://www.eas.asu.edu/ tsakalis/coursea/sblocks.zip`.