S2

Phase Lead Compensation of an Inverted Pendulum

This contributes 5% of the marks for ELEC2222.

You will investigate analogue control of an inverted pendulum. You will use the root-locus tools in MATLAB to complete initial designs on a simplified model, and then investigate the effect of the dynamics that were neglected.

Schedule		
Preparation time	:	3 hours
Lab time	:	3 hours
Items provided		
Tools	:	None
Components	:	None
Equipment	:	Bytronic inverted pendulum rig, 2x100 Ohms resistors op-amp circuit.
Software	:	Matlab
Items to bring		
Essentials	:	A full list is available on the laboratory website at: https://secure.ecs.soton.ac.uk/notes/ellabs/databook/essentials/

Before you come to the lab, it is essential that you read through this document and complete **all** of the preparation work in Section 2. If possible, prepare for the lab with your usual lab partner. Only preparation which is recorded in your laboratory logbook will contribute towards your mark for this exercise. There is no objection to several students working together on preparation, as long as all understand the results of that work. Before starting your preparation' read through all sections of these notes so that you are fully aware of what you will have to do in the lab.

Academic Integrity – If you undertake the preparation jointly with other students, it is important that you acknowledge this fact in your logbook. Similarly, you may want to use sources from the internet or books to help answer some of the questions. Again, record any sources in your logbook.

This exercise uses the standard **mark scheme** available on the Laboratory website at http://secure.ecs.soton.ac.uk/notes/ellabs/markscheme/

Mark Scheme

Preparation	
0	No preparation completed before the lab began *or* 'Essentials' not brought to lab
1	Partially completed, no understanding
2	Partially completed, minimal understanding
3	Partially completed, good understanding *or* All completed, minimal understanding
4	All completed, good understanding
5	All completed, excellent understanding
Progress & Quality	
0	No progress made/lab not attended
1	Minimal progress (<50% of section 3)
2	Completion of <u>most</u> of section 3
3	Completion of <u>all</u> of section 3
4	Excellent completion of <u>all</u> of section 3 *and* attempted section 4
5	Excellent completion of <u>all</u> of section 3 *and* <u>all</u> of section 4
Understanding	
0	No learning or understanding – just sitting there?
1	Very little understanding of section 3
2	Some understanding of section 3
3	Good understanding of section 3
4	Excellent understanding of <u>all</u> of section 3
5	Excellent understanding of <u>all</u> of section 3 *and* section 4
Logbook	
0	No work recorded in the logbook (or proper logbook not used)
1	Minimal record of the lab in the logbook, of very little benefit
2	Minimal record of the lab in the logbook, of some benefit
3	Good record of some activities/problems in the logbook, but sporadic coverage
4	Complete record of activities/problems in the logbook, some improvement possible
5	Excellent and complete record in logbook

Section 1

Aims, Learning Outcomes and Outline

This laboratory exercise aims to:

- Make you use PID and lead compensation control techniques on a physical system.
- Make you apply the root locus plot in designing controllers.
- Give you experience in the use of Matlab's Control Toolbox.

Having successfully completed the lab, you will be able to:

- Understand how the root locus is used in the design of controllers.
- Understand PID and lead compensation.

Section 2

Preparation

Read through the course handbook statement on safety and safe working practices, and your copy of the standard operating procedure. Make sure that you understand how to work safely. Read through this document so you are aware of what you will be expected to do in the lab.

2.1 General preparation

Before the lab you should make yourself acquainted with:

- the use of the Control Systems Toolbox in Matlab. In particular, you should check (via the help <command> facility available in Matlab), the following commands:

  tf, step, feedback, rlocus, series, bode, rlocfind

- the root-locus plot (see ELEC2222 notes site.

2.2 Modelling the system

The plant consists of an inverted pendulum (see Fig. 1).

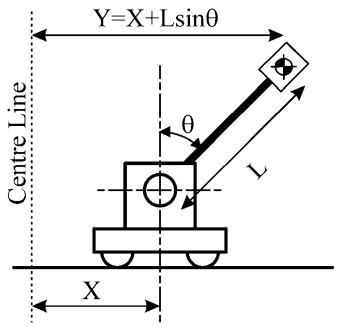


Figure 1: the inverted pendulum

Here X is the position of the cart, controlled via a servo; and Y is the position of the mass on top of the rod. Linearizing the system equation, we obtain the transfer function $P(s) = \frac{Y(s)}{X(s)} = \frac{-\omega^2}{s^2 - \omega^2}$, where $\omega = \sqrt{\frac{g}{L}}$, with L the length of the pendulum, and g=9.81 $\frac{m}{s^2}$ is the gravity constant. With the mass at the top of the pendulum rod, the effective pendulum length is L=0.20m.

Questions

Please answer the following questions in your **log book,** incorporating the **step** response plot.

1. Compute the numerical values for the transfer function P(s).
2. Write down the Matlab command to set up a transfer function pendulum corresponding to P(s).
3. In Matlab, use the step command to plot the step response of the system.
4. Is the plant stable or unstable?

In the rest of this experiment, we will design a compensator that will guarantee that Y moves accordingly with X. Later we will incorporate in our model the servo in our initial model, as in Figure 2.

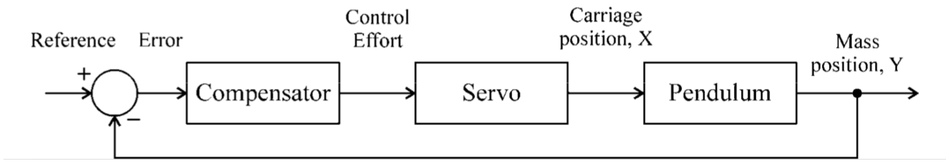


Figure 2: the control system

SECTION 3

Laboratory Work

3.1 Position control

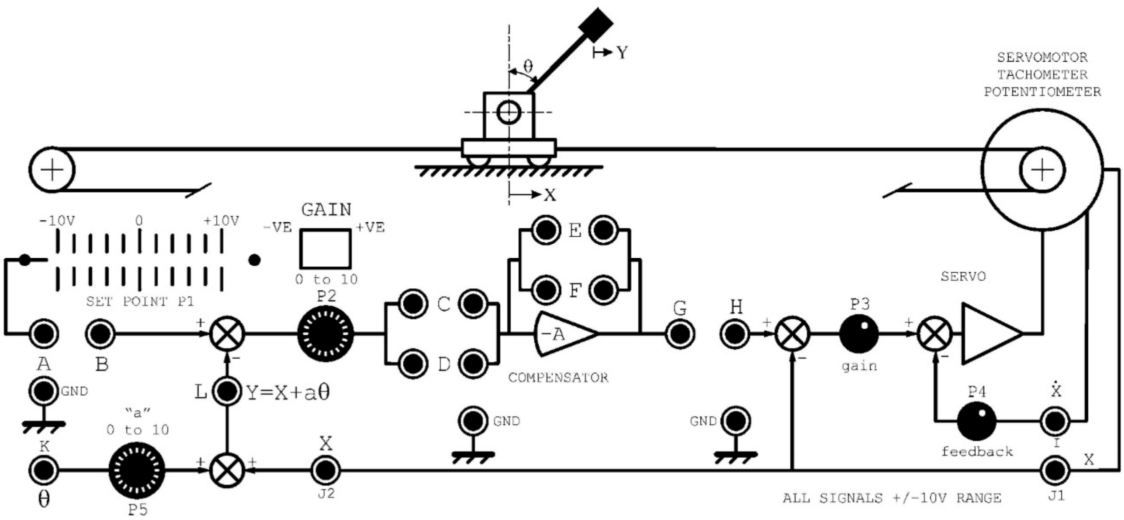


Figure 3: the Bytronix rig

To get acquainted with the control console, we first consider the position control of the pendulum carriage. Refer to Fig. 3 for the following instructions.

- Ensure the power to the pendulum unit is off.
- Unscrew the pendulum from the mount and place to one side.
- Position the set-point potentiometer (P1) to 0V.
- Connect the set point (A) to the servo amplifier input terminal (H).
- Position the servo gain (P3) at maximum, and the velocity gain (P4) at approximately the half way point. The P3 and P4 settings should remain fixed for the remainder of the lab.

Turn on the power, and verify that the position of the set-point potentiometer P1 determines the position of the carriage (X).

We need to control the position of the pendulum mass Y, which however, is not directly measured, but can only be inferred from measurements of X and the angle θ , see figure 1. For small angles θ is approximately equal to sin(θ), and consequently $Y = X + L\sin(\theta)$ is approximately equal to $Y = X + L\theta$. The X position is represented by a voltage V_X from the carriage potentiometer, which is added via an operational amplifier circuit to the voltage from the carriage potentiometer, which can be scaled by a factor a, controlled by potentiometer

P5. Thus we have: $V_Y = V_X + aV\theta$.

A detailed calibration of P5 is not needed in this lab. ***Set P5 to 2.7*** and ensure the pendulum weight is at the end of the rod.

3.2 Proportional control

Proportional control is the “P” in “PID”. Firstly we investigate it simulating the controller in

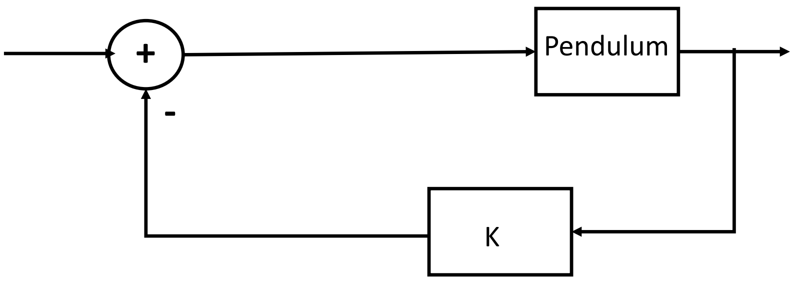
MATLAB.

Questions

5. With the variable pendulum defined to be the transfer function (see section 2.2), use the commands

   ```
   k = 1
   sysclp1 = feedback(pendulum,k)
   step(sysclp1)
   ```

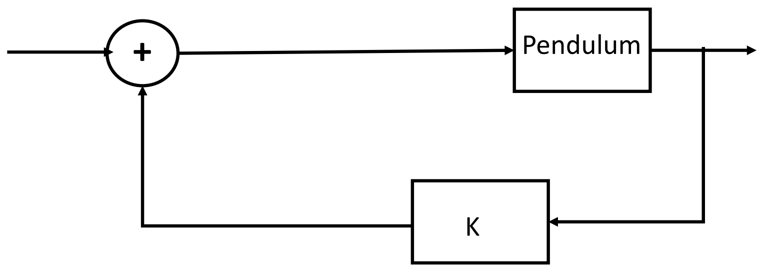
   to investigate the response of the system under negative feedback (the default option for the feedback command) for k=0,1,2,10. Comment on the stability of the system.
   Notice that you are working on the following configuration:



6. Investigate positive feedback i.e. k=0,1,2,10:

   ```
   k = 1
   sysclp12 = feedback(pendulum,k,+1)
   step(sysclp2)
   ```

Notice that you are working on the following configuration:



7. Plot the root locus using MATLAB:

   rlocus(pendulum) %Negative feedback
   rlocus(-pendulum) %Positive feedback

8. Comment on the stability in both cases. Can proportional control stabilize the system?

Bearing in mind your conclusions from above, we will now implement feedback proportional control on the physical pendulum.

Switch off the power. Attach the pendulum rod into the mounting in the carriage. Ensure the potentiometer P5 is set to 2.7, and the pendulum weight is securely fixed at the end of the rod. Connect the set point potentiometer to act as a reference input to the control system by linking A and B.

Set up the compensator to act as a unity gain inverting amplifier by placing a 100kOhm resistor across terminals D and another 100KOhm resistor across terminals F. Connect the output of the operational amplifier directly to the servo reference input (i.e. link G and H). Ensure that the gain switch is set to negative.

Set the gain potentiometer (P2) to zero. Support the pendulum with your hand and switch on the power. Adjust the set-point potentiometer to bring the carriage into the centre of the rig. Slowly increase the gain P2. Observe the carriage action as you move the pendulum to one side.

Questions

9. Compare the behaviour qualitatively for different values of the gain. Is it consistent with the root-locus predictions?

3.3 Lead compensation

In this section we design and implement a stabilising controller based on lead compensation. This is a controller described by a transfer function

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

Based on root-locus considerations, we need to place the zero of the compensator in the left-half plane (to hopefully attract the right-half pole of the plant). The simplest choice is to place the zero of the compensator to directly cancel the *stable* pole of the plant, i.e. we choose

$$\frac{1}{c\tau} = \omega$$

and then to use a `high' value of c, say c=10. In reality, we cannot exactly choose $\frac{1}{c\tau} = \omega$, and consequently we will use a compensator of the form:

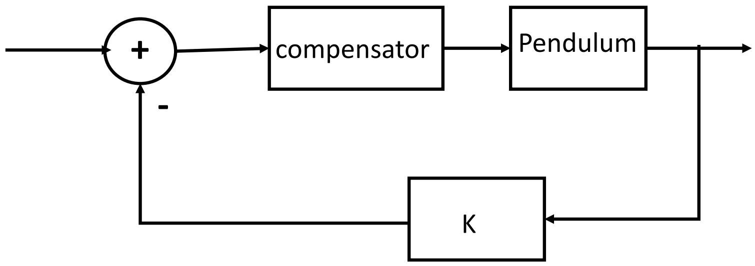
$$H_1(s) = \frac{1 + 0.1s}{1 + 0.01s}$$

Questions

10. Build a Matlab transfer function compensator for $H(s)$, and connect it in series with the plant and a negative gain:

    sys2=series(-compensator,pendulum)

    Notice that you are now working on the following configuration:



    Plot the root-locus with rlocus(sys2). Comment on the plot.

11. Build a transfer function compensator2 for $H_1(s)$, and plot the root-locus under positive feedback with

    sys3=series(-compensator2,pendulum)
    rlocus(sys3)

    Comment on the plot.

Note that this `approximate' pole cancellation can only be effective around a stable pole: trying to cancel unstable poles has a very different effect. In fact, it does not provide stability at all.

Physically, the transfer function $H_1(s)$ is implemented by means of an op-amp compensator circuit: see figure 4.

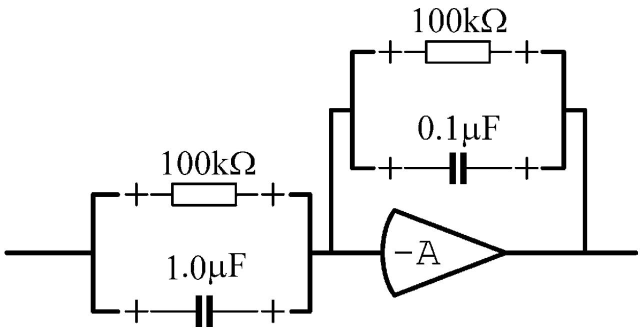


Figure 4

To verify that the op-amp circuit corresponds to the desired compensator, denoting the output voltage by V_0, the input voltage by V_1, the feedback impedance around the op-amp by Z_f, and the impedance between the input voltage and the summing junction by Z_1, we note that $\frac{V_0}{V_1} = -\frac{Z_f}{Z_1}$. For the parallel circuit consisting of a resistor R and capacitance C_1 we have $\frac{1}{Z_1} = \frac{1}{R} + C_1 s$; similarly, $\frac{1}{Z_f} = \frac{1}{R} + C_f s$. Thus

$$\frac{V_0}{V_1} = -\frac{1 + RC_1 s}{1 + RC_f s}.$$

We are now ready to physically implement the control scheme:

- Ensure that the pendulum mass is fixed in the maximum position on the rod.
- Set a to 2.7.
- Ensure that the servo gain (P3) is at maximum, and the velocity gain (P4) at approximately the half-way point.
- Set the gain (P2) to 0.
- Ensure that the gain switch is set to negative.
- Place the compensator components across terminals E and C as shown in figure 4.
- Hold the pendulum upright in the centre of the track.

- Connect the controller output to the servo input (ie. connect G and H).
- Loosely supporting the pendulum, gradually increase the gain (P2) until stability is achieved.
- You can now fully release the pendulum.

If you have difficulty in balancing the pendulum, try adjusting the potentiometer (a) slightly.

Questions

12. Explain using the root-locus plot why small values of the gain P2 do not lead to a stable closed loop.

3.4 The neglected servo dynamics

The servo has significant, if fast dynamics, which were neglected in our design procedure. We now investigate the effect of these dynamics, and show that they can indeed be neglected in design (up to a point...).

With the servo gain (P3) and the servo velocity feedback (P4) fixed in their original positions, the servo dynamics are approximately given by the second order model

$G(s) = \dfrac{1}{0.00025s^2 + 0.02s + 1}$, i.e. a second order system with natural frequency

$\omega = 10Hz$ and

damping ratio $\zeta = 0.7$. In our design we neglected these servo dynamics, assuming they were fast enough that they could be ignored.

To incorporate their effect in a MATLAB model and referring to Fig. 2, type

```
servo=tf([1],[0.00025 0.02  1])
```

actual=series(servo,pendulum)

sys4 = series(-compensator2,actual)

Now plot the root locus with rlocus(sys4).

Questions

13. Using the command rlocfind determine the upper and lower values for the range of gains for which the system remains stable. Validate the upper value qualitatively on the hardware by increasing the gain P2 until the onset of instability.

Note that the root-locus for the system without the servo dynamics (rlocus(sys3)) did not predict this onset of instability.

3.5 Robustness

When you have made good choices for the gain P2, say setting it to around 1.2, you will notice that the closed-loop system is quite robust. Tapping the pendulum does not cause it to loose stability, nor do changes in the pendulum itself (e.g. positioning the mass not completely on top) affect significantly the performance of the control system. The relative insensitivity of a control system to parameter changes in the plant is related to its *robustness*: the controller stabilises not only the plant it has been designed to control, but also other plants obtained from the original one changing its parameters. The purpose of this section is to have you investigate measures of robustness and verify experimentally the robustness of your controller.

Questions

14. Use k= rlocfind(sys4) to determine a stable value of the gain k, and define sysclp3=feedback(sys4,k). Draw the Bode plot:

    bode(sysclp3)

    and estimate the gain and the phase margin.

The gain and the phase margins are rough estimates of how robust a controller is. Denote the plant transfer function by *P(s)*; then the gain margin is the largest *a* such that the (fixed) controller *K(s)* stabilises the plant *aP(s)*. The phase margin is the largest phase in *P(s)* so that the controller *K(s)* stabilises the shifted-phase plant. The geometrical meaning of the phase- and the gain margin is easy to understand using a *Nyquist plot* of the closed-loop transfer function, and the *Nyquist stability criterion*. They can also be measured using Bode plots.

Questions

15. Move the pendulum mass down half-way the rod. Comment qualitatively on the relative stability by again tapping the weight.
16. Move the weight down to the bottom of the rod, and repeat the test. Comment on the response.
17. With the pendulum weight at the base of the rod, retune potentiometer P5 to get a good response.
18. Now move the pendulum weight to the top of the rod, and assess the relative stability.

Section 4

Optional Additional Work

Marks will only be awarded for this section if you have already completed all of Section 3 to an **excellent** standard and with **excellent** understanding.

Questions

19. Define new transfer functions pendulum2 and pendulum3 corresponding to the situations of Questions 15 and 16, respectively.
20. Redraw the relevant root-locuses.
21. Using rlocfind, investigate the range of stable gains, and compare this to the hardware response.