**ENEL441**

**Lab 3 Design of Compensators with Root Locus**

**Winter 2020**

The objective of this lab assignment is to design a compensator for a pendulum control and inverted pendulum using root locus analysis tools. Matlab tools such as SISOtool and Simulink are used to develop solutions and to assess performance.

Start with a pendulum on a fixed anchor as shown in Figure 1. The objective is to design a control loop that will move the pendulum to a specific angle with minimal overshoot and ringing assuming that we can apply a torque to the pivot point. The pendulum is shown in Figure 1.

L



u

M

Mg

torque is applied here based on an electric motor actuator, Tm

***Figure 1*** *Pendulum model*

The differential equation for the pendulum is determined based on evaluating the torque around the pivot point. The rotational inertia is  and the torque applied by gravity is given as . Also assume that the torque applied by the motor is



where  is the current through the motor and is the usual torque constant. Adding these components we get the DEQ related to the torque as



First assume that the motor current is zero such that no torque is generated such that



Assuming that  such that then we can linearize the DEQ as



This gives a marginally stable plant model with closed loop poles at . Hence the impulse response would oscillate forever as there is no loss in the system which is of course an unrealistic idealization.

For simplification in this present problem let and such that



The transfer function of the pendulum plant is given as

 with , and.

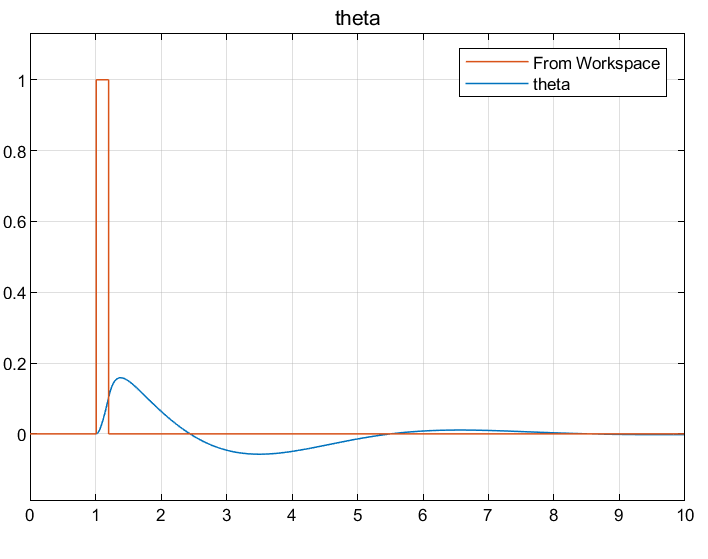
The initial objective is to design a compensator that sets the angle of the pendulum , to a desired reference angle denoted by r(t). That is control the motor current i(t) such that the error of is driven to zero. The following example shows a compensator of



Here the zero at s= -0.2 pulls the closed loop poles away from the jw axis and into the LHP. As a pole must always be associated with a zero, a pole far into the LHP at s=-10 is added. A Simulink model is shown in Figure 2 along with simulation results in Figure 3.



***Figure 2*** *Pendulum control simulation*



***Figure 3*** *Simulation of disturbance response with r(t) as a short pulse (orange) and pendulum response of theta*

For the following questions, use Matlab tools and routines such as SISOtool(), rlocus(), bode(), Simulink etc. to determine the answers. Show your work with supporting Matlab output as necessary. Include your M files and simulink models as well.

**QA)** First determine if a compensator with **proportional control** can result in a satisfactory control loop. Satisfactory in this present context implies that the loop has to be stable and also has to drive e(t) to zero. Support your answers with simulation output or other analysis.

**QB)** In **QA** you should have concluded that proportional feedback is not satisfactory. Then we try for a PD controller such that the compensator is given as C(s)=a+bs where a and b are constants. Show that a possible PD implementation can be realized with a settling time of several seconds and an overshoot of about 20 percent but that the DC gain of the closed loop response is not unity. List the final PD compensator you came up with.

**QC)** The PD compensator is unsatisfactory as it is not a type-I loop such that a gain factor or scaling factor must be applied to the input r(t) such that (t) tracks r(t). To fix this, consider a PI (proportional and integrator) compensator instead that has the form of



Is it possible to achieve a suitable control loop with the compensator configured in this way?

**QD)** As the PI is not satisfactory try to use a lead or lag circuit in addition to the PI controller. Show the final value of the compensator with the step response, closed loop bode and the root locus. Discuss any remaining deficiencies with the design.

**QE)** Model the motorized pendulum in simulink and verify that the same step response is achieved as in part **QD**. Use the linearized model in this case based on the plant transfer function of . Show the simulink model as well as the step response output. Verify that this is the same step response as calculated based on SISOtool.

**QF)** The advantage of Simulink is that we can model the nonlinearities of the control loop components. Add a nonlinear pendulum simulation to your simulink model of QF and run both outputs into a scope for comparison. That is represent the pendulum with the nonlinear DEQ of



Use the same compensator C(s) with both the linear and the nonlinear loops. Apply a step function input and couple the outputs to the scope for display and comparison. Initially use a small step of r(t)=u(t). Then use r(t)=4u(t) and you will start to see some differences in the nonlinear and linear loop step response outputs. Hint – to build up the nonlinear DEQ one possibility is to use two integrator blocks and a sine function block.

**QG)** Next use the model of **QF** to look at the motor drive currents for the linear and the nonlinear models of the pendulum positioner. Start with a small step size input of r(t)=0.1u(t) and note that the motor drive currents are very similar for the linear and the nonlinear. Then increase the input to r(t)=u(t) and then r(t)=3.14u(t). Explain the results. Why are the results different for larger step amplitudes of r(t)?

**Inverted pendulum**

Balancing an inverted pendulum has many applications especially in robotics. The Segway personal transport is an example of a practical inverted pendulum.



The inverted pendulum is the same as the non-inverted pendulum considered before only now we have instead of assuming . To determine an approximate linearized model of the inverted pendulum, take the general nonlinear DEQ of the pendulum and linearize it for the angle around . The following steps go through this linearization and determination of a compensator for a control loop that will maintain the inverted pendulum in a balanced position.

**QH)** Take the nonlinear DEQ of the pendulum as discussed in the initial part of this lab assignment and linearize it for the angle around  so that you can determine a linearized model of the inverted pendulum. Define a new angle variable of  and express the linearized DEQ in terms of instead of . How have the plant pole locations changed?

**QI)** Design a compensator for the inverted pendulum that is based on the linearized model of  with the criteria that the closed loop must be stable and must be a type one loop such that a step input of r(t)=u(t) will result in . Aim for a settling time of less than 4 seconds and an overshoot of less than 20 percent. Go through the same steps as before using SISOtool to develop the appropriate compensator. Your compensator should have the lowest complexity possible while meeting the criteria.