ELE459 Lab #1: Modeling, Design, and Implementation of an Analog Position Control System

Introduction

This is a two-week lab. In Week 1 you will record experimental motor velocity step responses to obtain a transfer-function model for the motor system. Before coming to lab for Week 2, design a control system (phase-lead compensator with prefilter), similar to the third control system of Homework 1. In Week 2, the position control system is implemented using Simulink to control the hardware system.

1. Modeling the Hardware (watch the on-line lab lecture first!)

1. On the actual hardware system, put in step inputs of various amplitudes (1-volt, 2-volts, 3-volts). Note that these constant-voltage signals are applied as inputs to the motor power amplifier. Send the data from the motor velocity scope to the workspace so it can be plotted in Matlab.

2. You may also want to save the data. The way to do this for the 1-volt response is as follows: at the Matlab prompt type `save 1_volt`. This will create a file called `1_volt.mat` containing all variables in the Matlab workspace. Whenever you want to make a plot do the following: at the Matlab prompt type `load 1_volt` and all variables (including the scope data) will be imported to Matlab.

3. For each of the step responses obtained, find the corresponding mathematical model. These models should be approximately the same (but perhaps not identical).

4. Once all three velocity signals have been obtained and individually modeled, use the procedure described in the lab lecture to find a single model for the hardware system.

Here is the code to design a discrete-time system (it could also be called a filter) that digitally differentiates and low pass filters an incoming signal. The resulting filter is in the form of a discrete-time state-space model `(Af,Bf,Cf,Df)`.

```
Ts=0.15; % approximate settling time of plant
Tsf=Ts/20;  % desired  settling time of analog and digital filters
bwfp=(-4.62+j*4.62/)/Tsf; % create poles for 2nd-order Butterworth filter
den=poly([bwfp conj(bwfp)]);
num=[den(end) 0];
analog_filter=tf(num,den); % create analog filter
digital_filter=c2d(analog_filter,T,'tustin'); % convert analog filter to discrete time
[Af,Bf,Cf,Df]=ssdata(digital_filter); %state-space model of digital filter
```

2. Design and Simulation (do this before coming to Week 2)

1. Using the numerical values of α and β found for the hardware system, repeat the third control system design from homework 1, which consists of a $C(s)$ phase-lead compensator plus a prefilter. Recall that the design in homework 1 had two constraints for a reference input of 60 radians: (1), the plant output could not exceed 60 radians (no overshoot) and (2), the plant input could not exceed five volts.

3. Controlling the Hardware (Week 2)

1. Implement the control system in Simulink and connect it to the motor/cart system using the hardware blocks found in Simulink Desktop Real-Time.

2. Obtain the experimental step response to a 60-radian reference command and record the input and output of the plant.