ELE 459 : Digital Control Systems Lab #3

Digital State Feedback Regulation of a Cart-Pendulum System

Noah Johnson

Week 1 - *Simulation of Digital State Feedback Regulator System*
Week 2 - *Implementation and Analysis of Digital State Feedback Regulator System*

I. Summary

During the first week of this lab, the goal was to simulate a Digital State Feedback Regulator and derive feedback pole locations such that the resultant system would have acceptable stability bounds. During the second week, the previously derived values were utilized to implement a physical Digital State Feedback Regulator system that balanced a rod upright. The simulation was done using Matlab and Simulink, and the hardware implementation was accomplished using a cart-pendulum positioning system.

II. Results

During the first week, two sets of results were derived, corresponding to the Scaled Bessel and Added Damping pole methods.[1] The continuous time pole locations (or "spoles") were as follows:

$$spoles_{Bessel} = \frac{s_4}{T_{s\,(bessel)}} \tag{2.1a}$$

$$spoles_{Added\ Damping} = \left[-26.66 \ \ -6.442 \ \ \frac{s_2}{T_{s\,(Added\ Damping)}}\right] \tag{2.1b}$$

Where "s_n" is the set of n^{th} order bessel poles, and "T_s" is the settling time of the overall system which was user optimized. This value was set to 635 ms for the scaled bessel design, and 835 ms in the Added Damping design. The following Gain vectors were found:

$$K_{Bessel} = [29.055,\ 4.3918,\ -0.0759,\ -0.0356] \tag{2.2a}$$

$$K_{Added\ Damping} = [29.1129,\ 4.5192,\ -0.0450,\ -0.0328] \tag{2.2b}$$

The simulated stability margins for the Bessel design were 30.1 dB, -6.26 dB, and 30 degrees (for upper, lower and phase margins, respectively.) The simulated stability margins for the Added Damping design were 22.19 dB, -7.57 dB, and 55 degrees (for upper, lower and phase margins, respectively.)

In the second week these derived values were used to design a Full state feedback regulator for balancing an inverted pendulum.

The hardware tests were successful.

---

[1] Note that the plant model is unchanged, $\alpha = 26.6667, \beta = 133.33$.

III. Equipment List

A. Hardware Components
   1. PC Tower running Windows Vista
      a) Software includes: MATLAB 2017a, Simulink 2017a
   2. Power Amplifier
      a) Aerotech 4020 Linear Servo Amplifier
      b) Serial Number: EFA401
   3. Motor Driven Cart System
      a) Aerotech 1000DC Permanent Magnet Servo Motor
      b) Part Number : 1050-01-1000
      c) Serial Number : 53012
   4. Digital to Analog Converter
      a) Part Number : PCI-DAS1002
   5. Angular Position Sensor
      a) Part Number: HEDS-6500 B05

IV. Procedure (Week One)

The first week of the lab was dedicated to deriving a model of the DSFR system and implementing it using Simulink. This was done for both the Bessel and Added Damping cases. The overall simulation model was unchanged between these two cases; as such the model will only be discussed once. The high level plant schematic can be found in figure 5.1.

A linearized model of the pendulum system was provided by Professor Vaccaro, such that the state space model was as follows:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \pm A & 0 & 0 & \mp\frac{AC}{ng} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -C \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ \pm\frac{AD}{ng} \\ 0 \\ D \end{bmatrix} u(t)$$

Figure 4.1 - *Linearized State Space Model of the inverted pendulum system.*

It is given that in the above model[2]:

$$A = 41.5 \frac{rad}{sec^2}\,,\ C = 25.0\, sec^{-1}\,,\ D = 2,633 \frac{rad}{(volt{-}sec^2)}\,,\ n = 495\,,\ g = 9.81 \frac{m}{sec^2} \qquad (4.1)$$

And that the state space variables are:

$$\mathbf{x} = \begin{bmatrix} x_1(\text{pendulum position}) \\ x_2(\text{pendulum velocity}) \\ x_3(\text{motor position}) \\ x_4(\text{motor velocity}) \end{bmatrix} \qquad (4.2)$$

Note that “A” here is denoted “Amat” in all Matlab files to avoid collision with plant state space variable A.

[2] *Control System Design, A State Space Approach*, Richard J. Vaccaro. (Table 3.9)

From this model, a Simulink diagram was made and connected with a simulated plant and the feedback gain vector **K** to be derived. The following filter initial conditions where defined as well.

$$X_1[0] = \begin{bmatrix} 0.17 \\ 0 \\ 0 \\ 0 \end{bmatrix}, X_2[0] = \begin{bmatrix} 0 \\ 0.17 \\ 0 \\ 0 \end{bmatrix} \tag{4.3}$$

These were used to test that the balancing capabilities of the system functioned as expected. Note that if $x_1 = 0.17$ rad, as in $X_1[0]$, this corresponds to an approximately 9.7 degree variation from vertical, for which the technical term is a "tap."

```
A = 41.5;
load sroots;
C = 26.6667;
D = 20* 133.33;
n = 495;
g = 9.81;
T = 0.01;

At = [ 0 1 0 0; A 0 0 -(A*C/(n*g)); 0 0 0 1; 0 0 0 -C];
Bt = [0; ((A*D)/(n*g)); 0; D];
Ct = eye(4);
Dt = zeros(4,1);

Ts = .635;%.85156;
Den = poly(s2/(Ts/20));
Num = [Den(end) 0];

vfilter_a = tf(Num,Den);
vfilter_d = c2d(vfilter_a,T,'t_vistin');

[Af,Bf,Cf,Df] = ssdata(vfilter_d);
spoles_nb = [-26.6667 s3/Ts];
spoles_b = s4/Ts;
zpoles= exp(T*spoles_b);
[phi, gamma] = c2d(At,Bt,T);
K = place(phi, gamma, zpoles);
```

Figure 4.2 *-MatLab Script written to generate the feedback gain vector* **K**. *The accompanying Simulink Diagram can be found in Figure 5.1*

Then the Matlab script seen above was written that took a given set of user defined s-plane poles, either Bessel or Added Damping in this case, and mapped them to the Z-plane. It then used

```
> [phi, gamma] = c2d(At, Bt, T)
```
(4.4a)

to convert the continuous time plant model to discrete time, and finally used the command

```
> K = place(phi, gamma, zpoles)
```
(4.4b)

to solve for the matrix value **K**. Stability margins for this model were calculated using the "dsm()"[3] command and recorded. These bounds were tested using small perturbations described by initial conditions that corresponded to either an initial pendulum velocity or angle and in this way the simulation was verified.

This concluded week one of this lab.

[3] Not pictured in Figure 4.2. : `> dsm(phi, gamma, K)`

V. Procedure (Week Two)

During the second week of this experiment, the previously constructed Simulink model was modified to integrate the physical system, and several tests were run.

The first of several major considerations was the fact that the system when initialized would not be in its regulated state. Rather, it would be in the exact *opposite* state than that which was desired. Furthermore, the "zero point" of the rotary encoder for the pendulum was 180 degrees opposite the desired position. Therefore, a subtraction block[4] was first added to the pendulum position to correct for the inverted position reading. Then, to prevent the system from immediately going unstable as the pendulum was righted, a switch and constant zero block was added. This allowed for safe adjustment of the system.

From there, the real time test was started. The system was observed first from a safe distance, then it was tapped to gain experimental data. This portion of the lab procedure was repeated several times to ensure data accuracy. One small issue was experienced with the experimental setup, that being the pendulum rotary encoder spontaneously reset to 0 degrees several times, causing the system to immediately go unstable and drift off to one end of the track. This was fixed by using a different cable to connect the ADC to the rotary encoder.

This process was completed for both Bessel and Added Damping feedback pole locations.

[4] Pictured in Figure 6.2 on the left side of the diagram.

VI. Data and Calculations

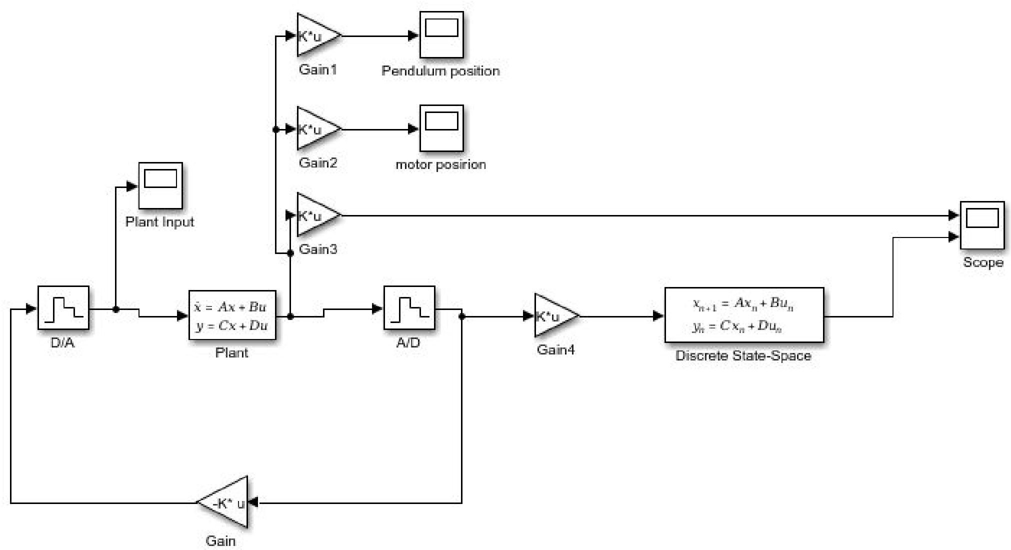

Figure 6.1 - *Simulink diagram for simulation of Inverted Pendulum System. Used during week 1 of Lab to derive Feedback Gain vector* **K** *and compare methods of derivation.*

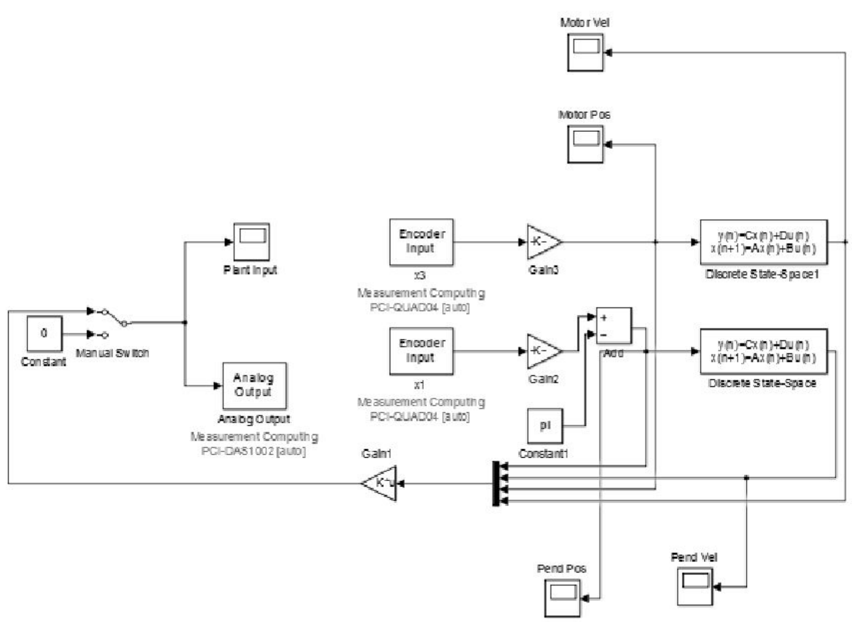

Figure 6.2 *-Simulink Diagram of Hardware implementation used in Week Two. Note that the gain blocks on the encode inputs serve as state variable isolation.*

As most, if not all of the calculations in this lab where done using MatLab, the derived values will be discussed.

First recall that the absolute bounds on classical gain and phase margins are as follows:

$$UGM = 20log_{10}(q_{max}) \,,\ UGM > 3\ dB \,, \tag{6.1a}$$
$$LGM = 20log_{10}(q_{min}) \,,\ LGM < -3\ dB \tag{6.1b}$$
$$PM \geq 30° \tag{6.1c}$$

And consider the table below.

Margin Type	Scaled Bessel	Added Damping	Acceptable Bounds
Lower Gain Margin (dB)	30.1	22.19	3
Upper Gain Margin (dB)	-6.26	-7.57	-3
Phase Margin (°)	30	55	30

Figure 6.3 *- Derived inverted pendulum system stability margins, Classical.*

Considering that the derived values are not infinitely precise, nor has a thorough analysis of error been conducted, a system model with a phase margin of *exactly* 30 degrees seems inadvisable. The phase margin of the Added Damping system is much better as it is above the strict 30 degree minimum, however all gain margin values for both systems are well within acceptable bounds.

The more relevant result in this simulation is the length of the settling time. Given that the circuit is noisy, the shortest possible settling time is not desirable, as it will introduce “jitter”[5] into the implemented system. This is not visible in the simulation and must be considered. In hardware, this was solved by including a high pass filter in the integrating block of the feedback loop. This drastically reduced noise and made the system usable. This block is visible in all the simulink diagrams as the discrete time state space model on the right side of the screen.

[5] Visible in all Hardware pictures

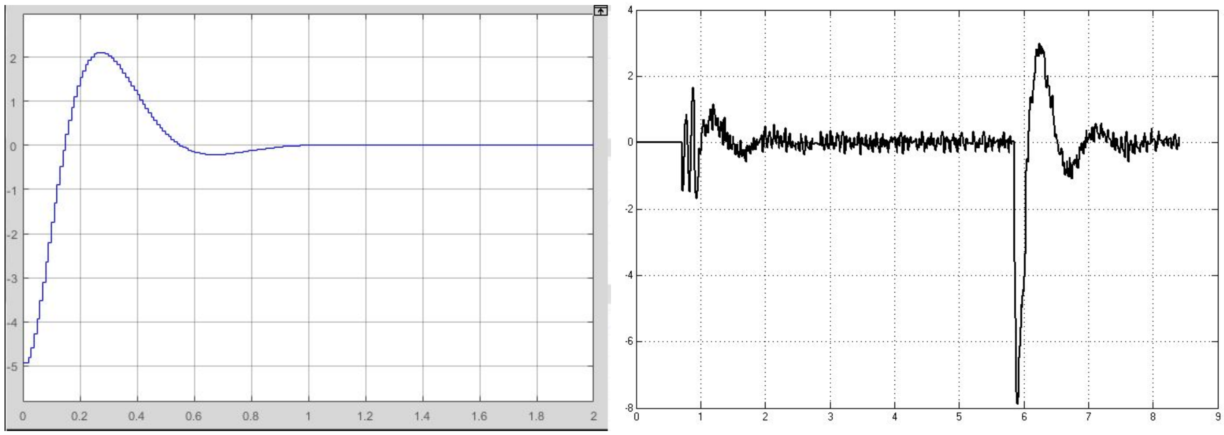

Figure 6.4 *- Plant Input as simulated (left) with initial condition X_2 and implemented in hardware (right). Both use Bessel poles. Note the similarity in shape.*

Considering Figure 6.4, the similarity in wave shape and response shows that the simulation was fairly accurate and a valid model. The only major difference is that graph in hardware shows a far sharper negative peak, corresponding to a external stimulus that was much stronger than the one simulated using initial conditions.

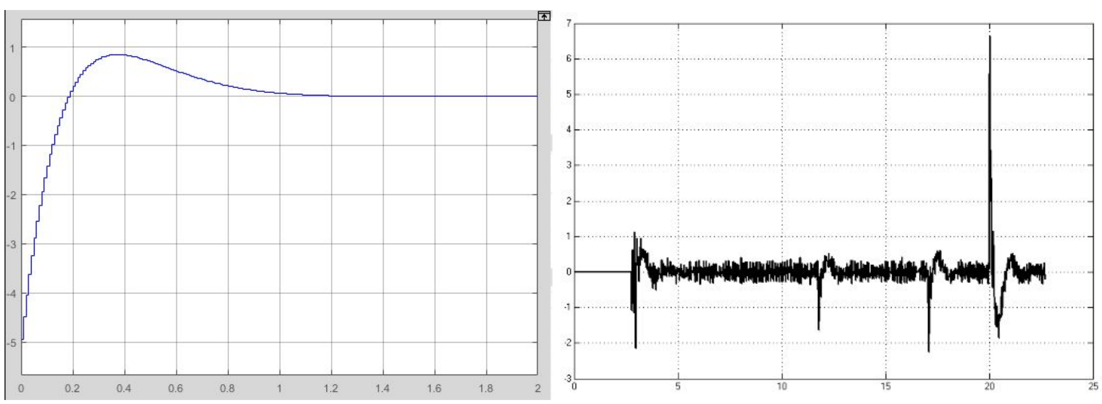

Figure 6.5 *- Plant Input as simulated (left) with initial condition X_2 and implemented in hardware (right). Both use Added Damping poles. The lack of similarity is attributed to an error in Simulink simulation, and when unnoticed until time of writing.*

Many of the other discrepancies in the charts hint at an error in either experimental design or in code debugging. However, the difference in noise between the Bessel and Added Damping models is clearly visible by a reduced noise signal in the Added Damping graph.

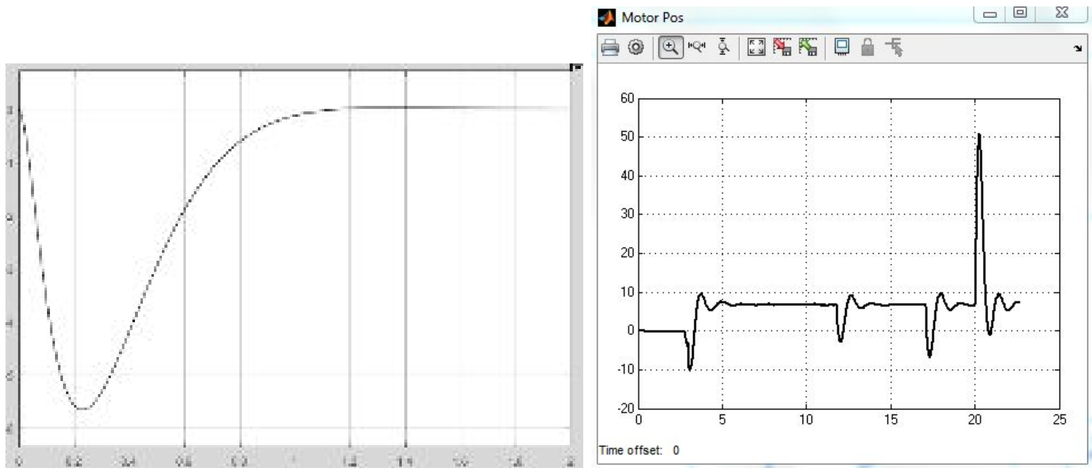

Figure 6.6 *- Motor Position as simulated (left) with initial condition X_2 and implemented in hardware (right). Both use Added Damping poles. The difference of almost a factor of 2 is attributed to especially hard taps in hardware testing/*

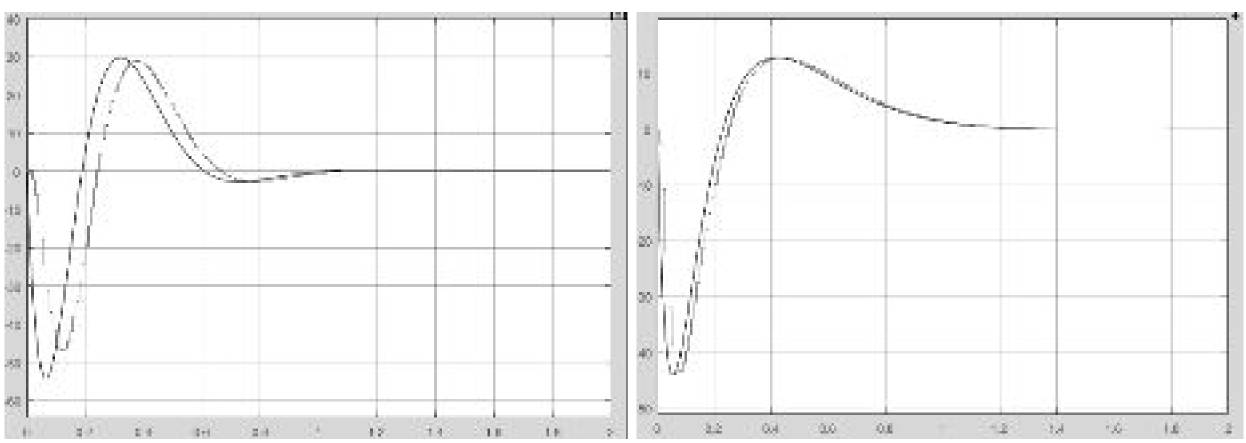

Figure 6.7 *- Motor Velocity with a 5 (left) and 20 (right) times faster filter implemented in simulation. The difference in output tracking speed is very noticable. This change in filter response time allows the system on the right to balance the pendulum with minimal jitter.*

While implementing a filter did reduce the noise in the feedback loop by nearly 60%, it was still very visible in the final optimized system. It remained present to nearly the same degree when the filter settling time was increased slightly to “smooth” the plant input.

Overall it can be deduced that the full state feedback method does not provide the optimal results for this type of control system, but the achieved results could be deemed acceptable.

This concludes Lab 2.