Root Locus Design
of a Linear-Flex System

Student : Corey Gravelle
Perm Number: 8094211
E-mail: cgravelle@umail.ucsb.edu

11/26/2009

Department of Electrical and Computer Engineering, UCSB. ECE147A

1. Abstract

The purpose of this lab is to control the response of a cart even while it is connected to another one via a spring. One should compare the differences, if any, between theoretical simulations and actuality and draw conclusions on why these differences are made. By using two different transfer functions, one taking the second mass into account and one not, one may find the robustness of a response is not entirely dependent on the given plant.

2. Introduction

One may assume the following approximations in modeling the system shown in Figure 1... F is the force supplied based on an input voltage V:

$$F = \frac{K_m K_g}{R_m r} V - \frac{K_m^2 K_g^2}{R_m r^2} \dot{x}_1$$

Given this and the acceleration state equations as follows:

$$\begin{aligned} F - k(x_1 - x_2) &= m_1 \ddot{x}_1 \\ k(x_1 - x_2) &= m_2 \ddot{x}_2. \end{aligned}$$

One may find the Fourier transform and combine the equations to arrive at:

$$\begin{aligned} X_2(s)(m_2 s^2 + k) &= k X_1(s) \\ X_1(s)\left(m_1 s^2 + \frac{K_m^2 K_g^2}{R_m r^2} s + k\right) &= \frac{K_m K_g}{R_m r} V(s) + k X_2(s) \end{aligned}$$

Plugging in actual values for Km, Rm, Kg, r, m1 and m2, one will arrive at the final transfer functions:

$$\begin{aligned} \frac{X_1(s)}{V(s)} &= 2.97 \frac{s^2 + 61.2}{s^4 + 13.24 s^3 + 127.15 s^2 + 810.37 s} \\ \frac{X_2(s)}{X_1(s)} &= \frac{61.2}{s^2 + 61.2}, \end{aligned}$$

Using these transfer functions as a model, one should be able to find desirable controllers to manage both positions with relative ease.

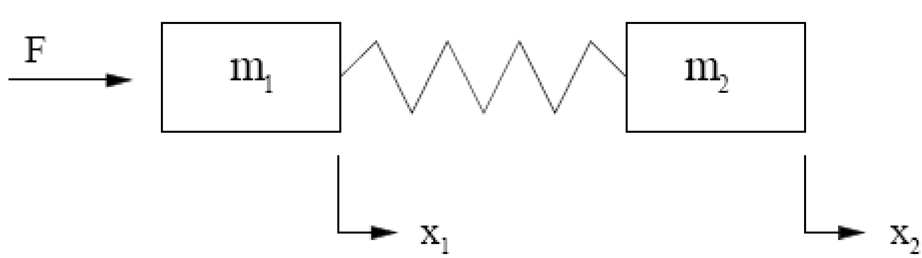


Figure 1: Free body diagram of two carts m1 and m2 connected by spring

3. Procedure

1. Use root locus design techniques to design controllers and improve the response of the system using the transfer function X1(s) / V(s). Compare the actual step responses with the theoretical, and note any differences you may find. Explain why these differences arose and attempt to tune your controller for the following:

a) Design a first or second order controller using root locus design techniques to pull the locus as far to the left half-plane as possible.

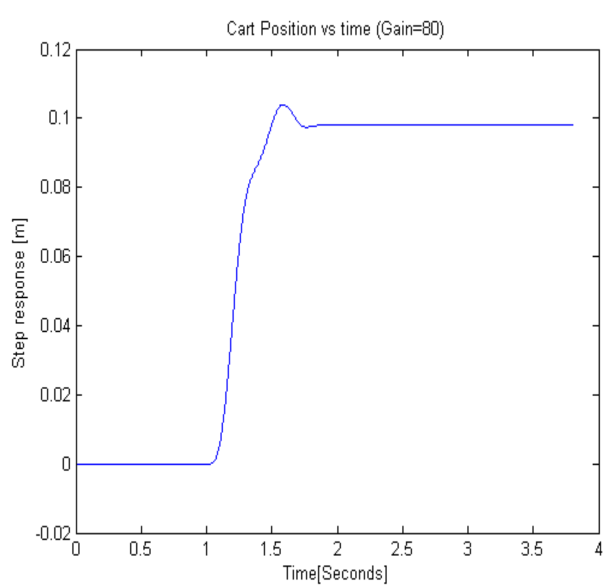
b) Choose a second order controller which cancels the left-plane oscillatory poles and has a desirable response.

c) Design a third order controller that results in a better step response with little time spent in saturation.

2. Repeat the above experiment, but for the plant X2(s) / V(s). Note the differences and similarities between the controllers designed for each cart.

3. Results and Discussion

1a. Using MATLAB's `sisotool()` function, we designed a filter with a zero at -13.4 and a pole at -31.7. These were found by adjusting the closed-loop gain until the imaginary poles had the most negative real part as possible, to provide the most stable response.



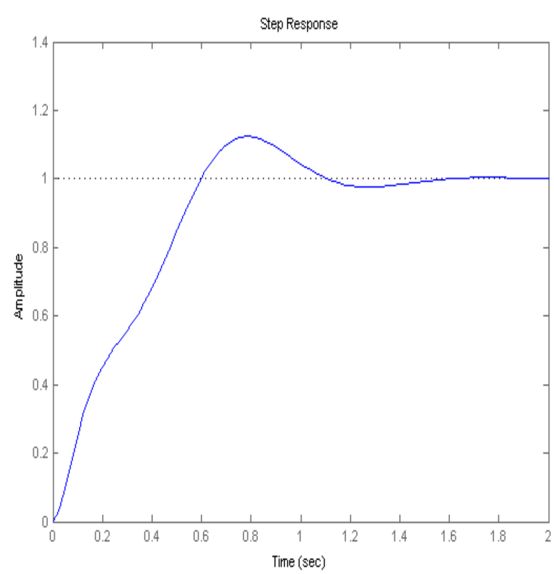


Figure 1: Actual (size 0.1) and theoretical (size 1) step response with a first-order controller. Because the system is linear, we may assume scaling the step would correspond to a scaled step response.

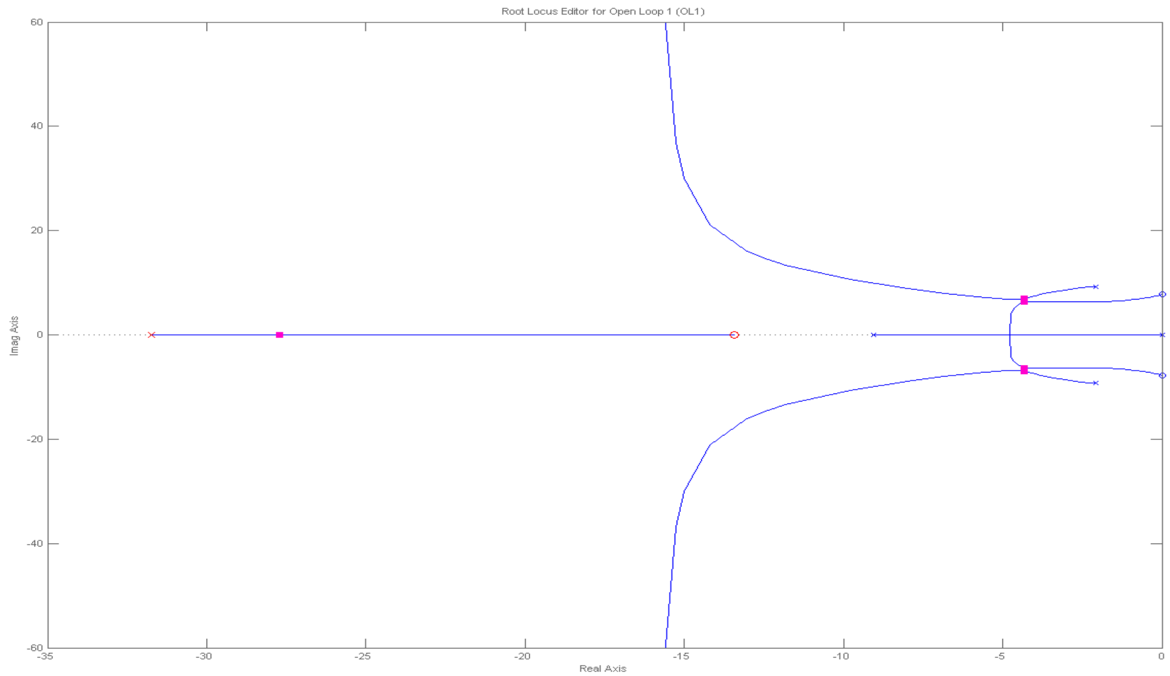


Figure 2: Root locus plot to find ideal pole-zero location and corresponding gain.

1b. As in part 1a, we used the root locus plot tools with Matlab to adjust poles and zeros to find controller zeros at -2.1 ± 5.22j and poles at -1.1 ± 11j. Figure 3 shows the improved rise-time and Figure 5 shows that, ideally, the settling time is improved as well.

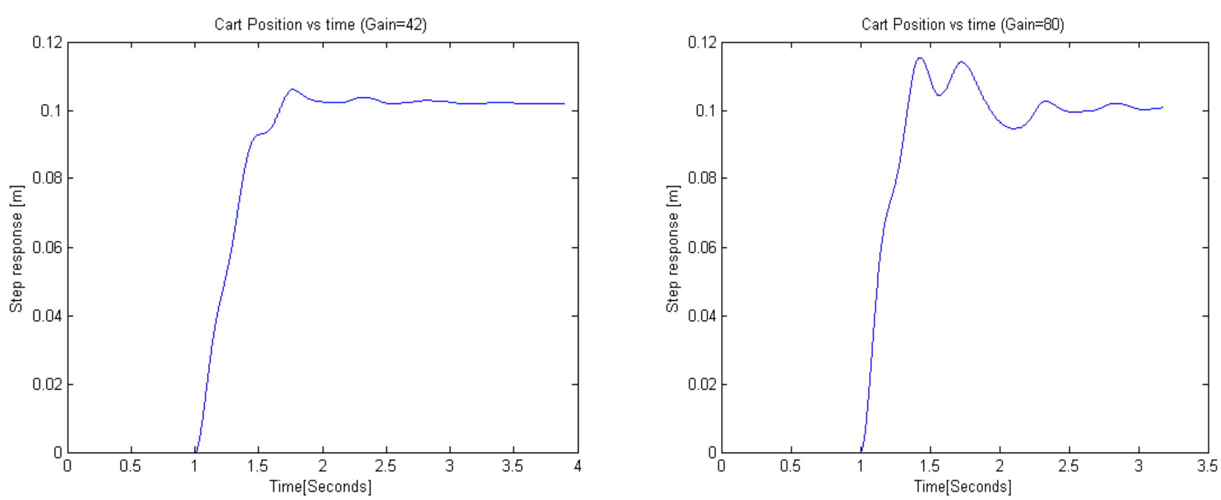


Figure 3: Actual step response for second-order controller with K= 42 and K = 80, respectively. This controller requires less gain for minimum settling time based on a faster rise-time from the added poles' trait of adding to the magnitude of the negative real part (increasing overall ωn).

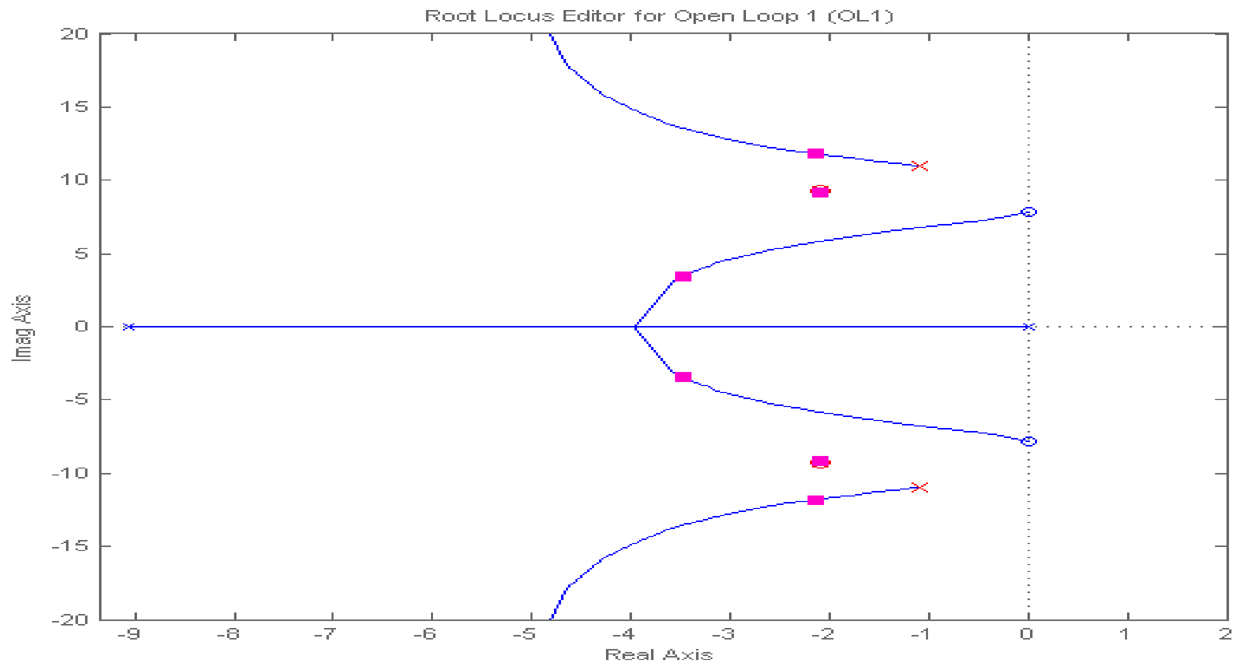


Figure 4: Root locus plot used to find ideal second-order poles, zeros, and gain of X1(s)/V(s)

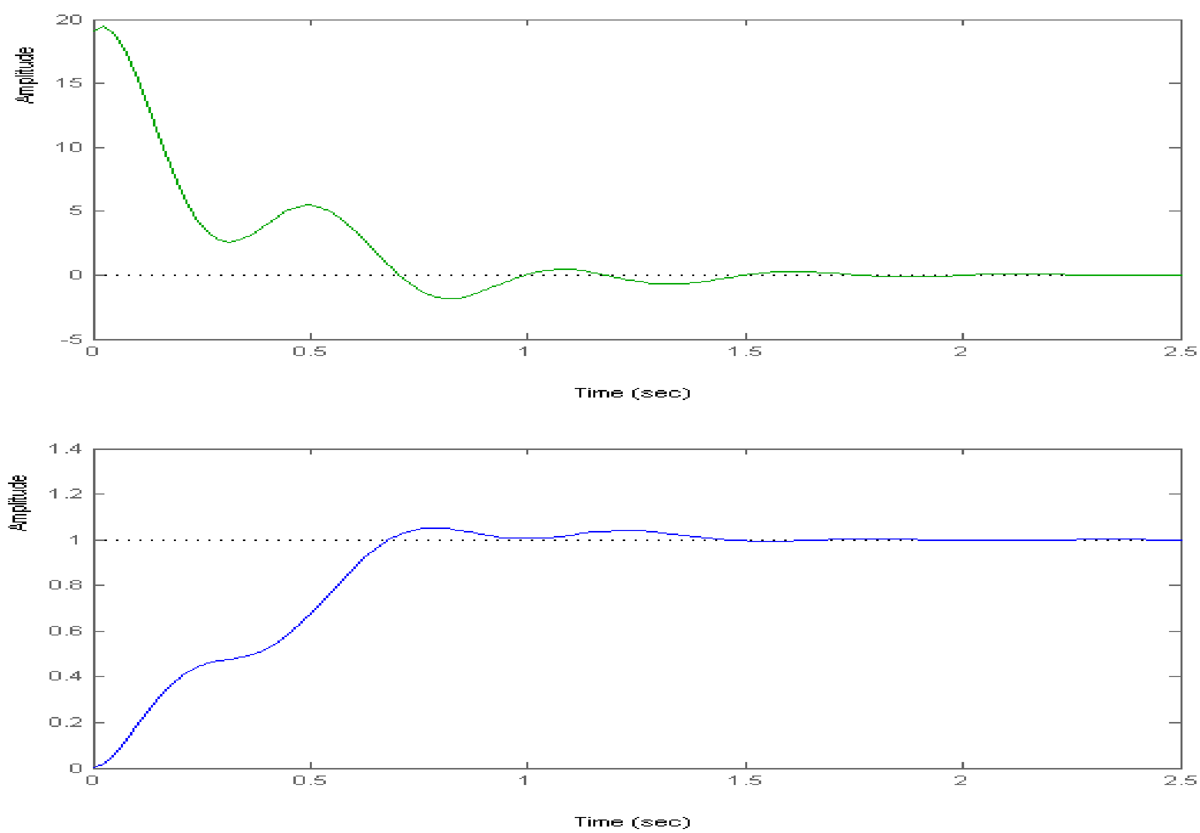


Figure 5: Step response of voltage input and output position for a step of 1. We are only interested in a step response of 0.1, so the amplitude of the input voltage peaks at about 2V, rather than 20.

1c. Using similar techniques, we found a third-order controller with poles at -13.7 and -13.8 ± 59.3j and zeros at -6.96 and -14.5 ± 35.9j. This results in a robust transfer function which allows practically any closed-loop gain to be implemented and results in a short rise and settling time. Though simulated with no steady-state error, in practice there is a lower threshold which must be passed for the motor to move the cart and adjust its position. Practically, a pole added at zero may help neutralize the real-life steady-state error.

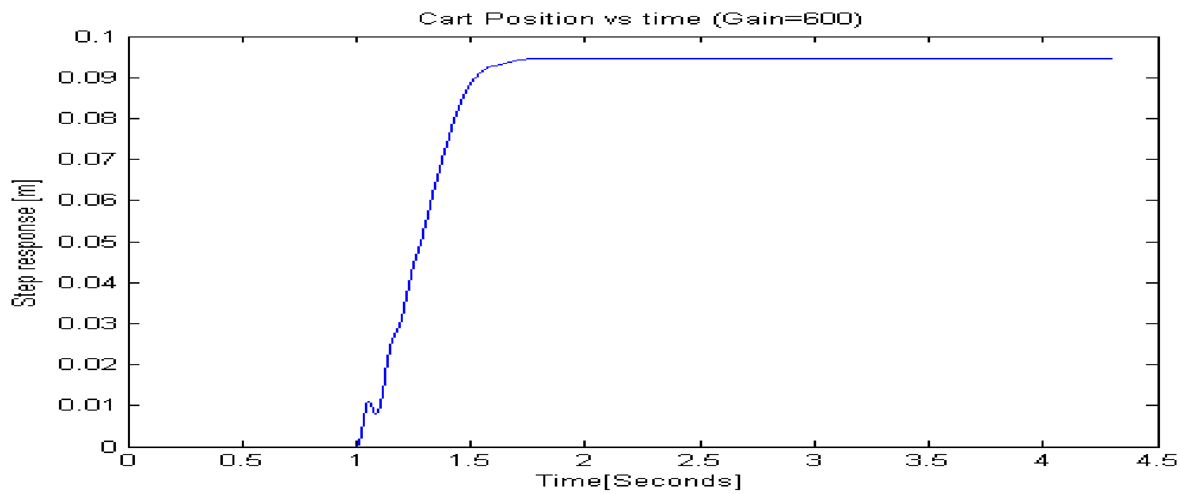


Figure 6: Actual step response of X1(s)/V(s) for a step of 0.1.

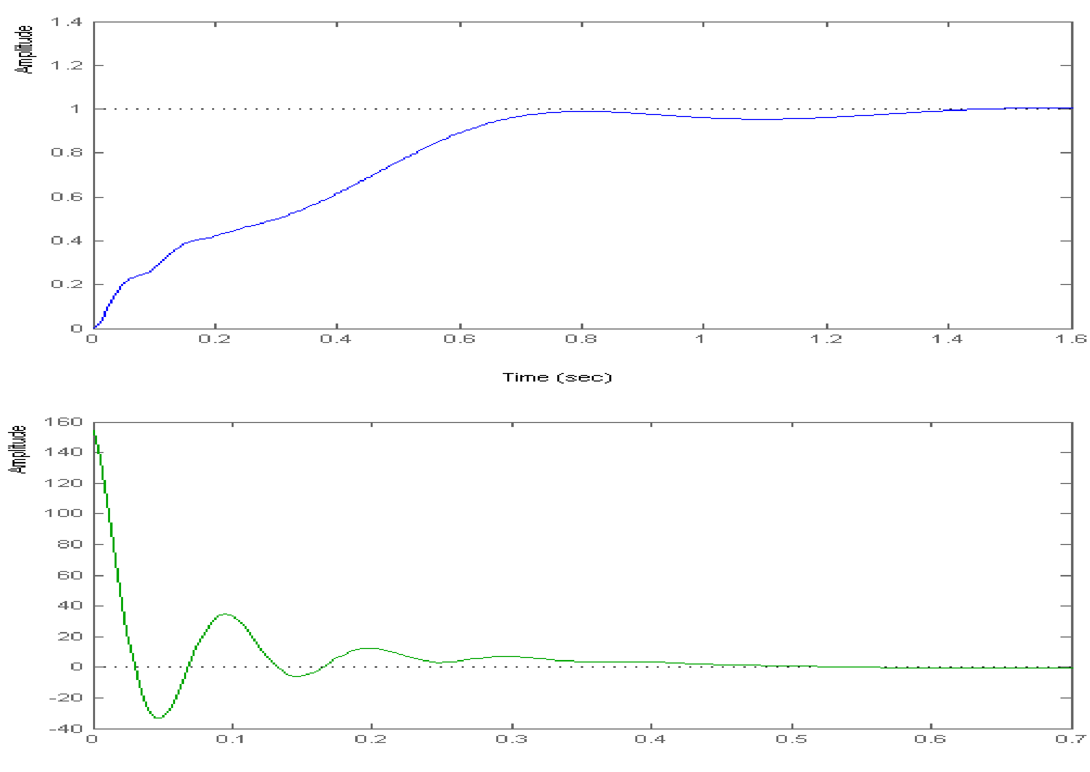


Figure 7: Simulated step response and voltage input of X1(s)/V(s) for a step of 1. From our desired step of amplitude 0.1 and saturation of 5V, one may find the voltage input shall not exceed 50 in this graph for any extended period of time; in this case, it only does so for about 20ms.

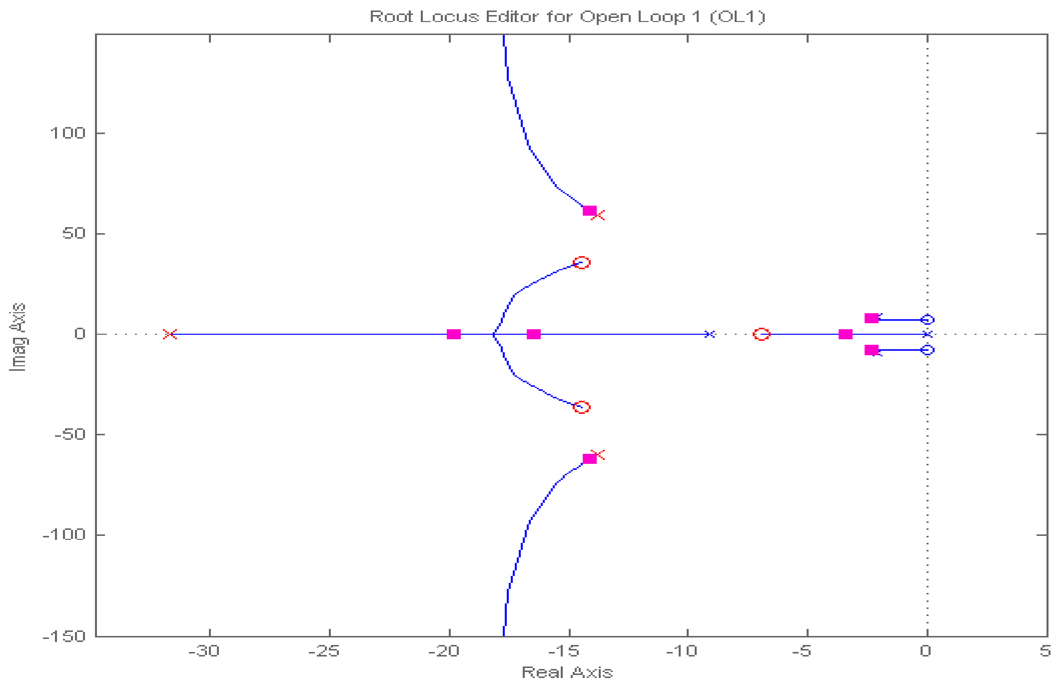


Figure 8: Root locus plot for third-order controller of X1(s)/V(s)

2a. Utilizing the same root locus techniques as in part 1, we found the ideal first-order zero to be -20.1 and pole to be at -7.6. Calculating the ideal gain to be 6.12, we compensated for resistance and lower motor threshold by testing with a gain of 10.

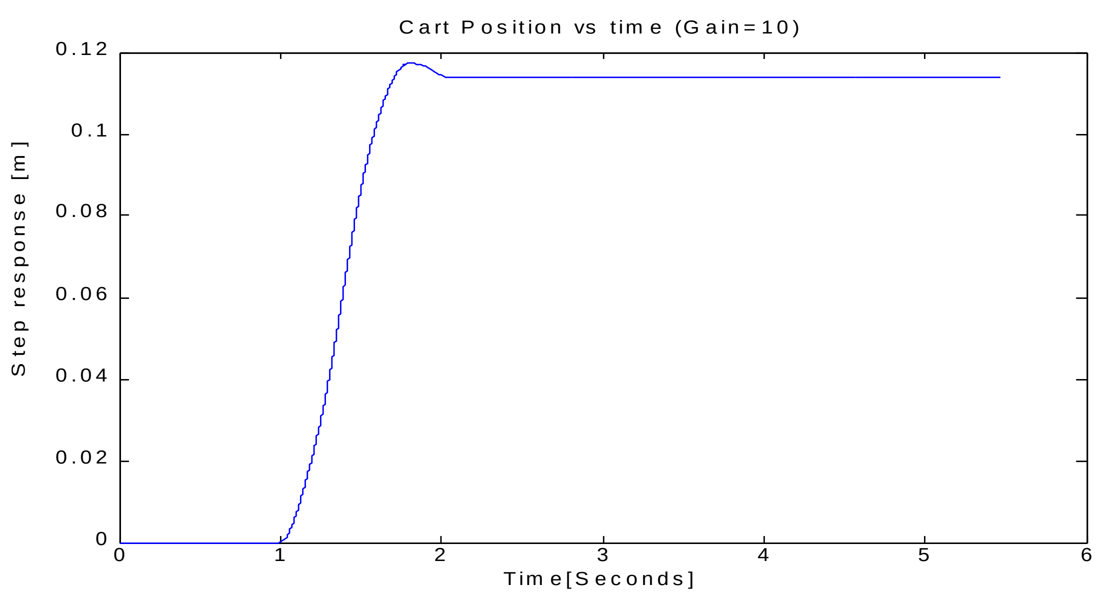


Figure 9: Experimental step response for first-order controller of X2(s)/V(s)

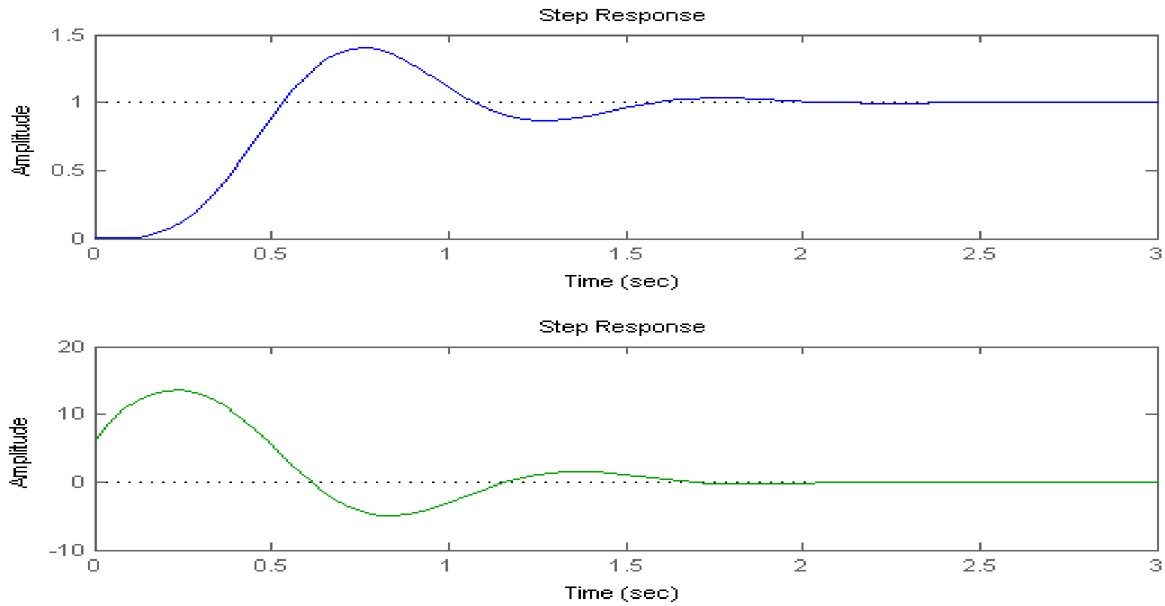


Figure 10: Simulated step response of magnitude 1 for X2(s)/V(s)

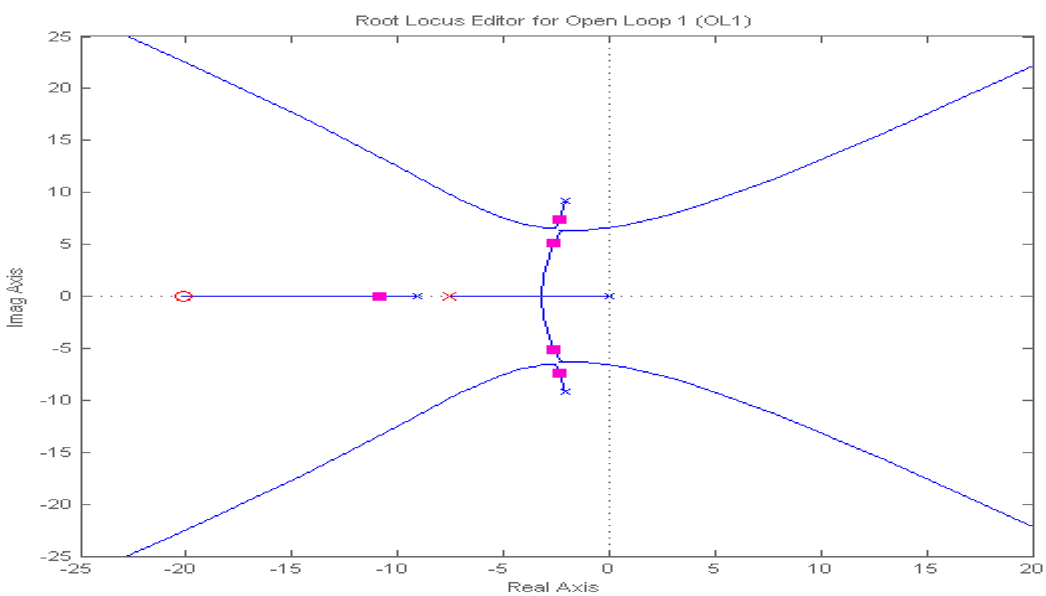


Figure 11: Root locus plot for first-order controller for X2(s)/V(s)

2b. For the second-order controller, we attempted to cancel the instability of the first-order system with a zero near the poles to replace the pole with a more negative real-part pole and allow for an improved rise-time characteristic and less steady-state error as well. We used root locus techniques in calculating the ideal zeros at -2 ± 9.2j, poles at 4.66 ± 15.44, and gain at K = 60. This however proved to cause a ringing effect as it possibly brings the poles too near the imaginary axis, corresponding to a small dampening factor (for the imaginary poles with a denominator of {s^2 + 11.25s + 434}, this is 11.25/(2*sqrt(434)) calculating the dampening factor to be a mere 0.27.

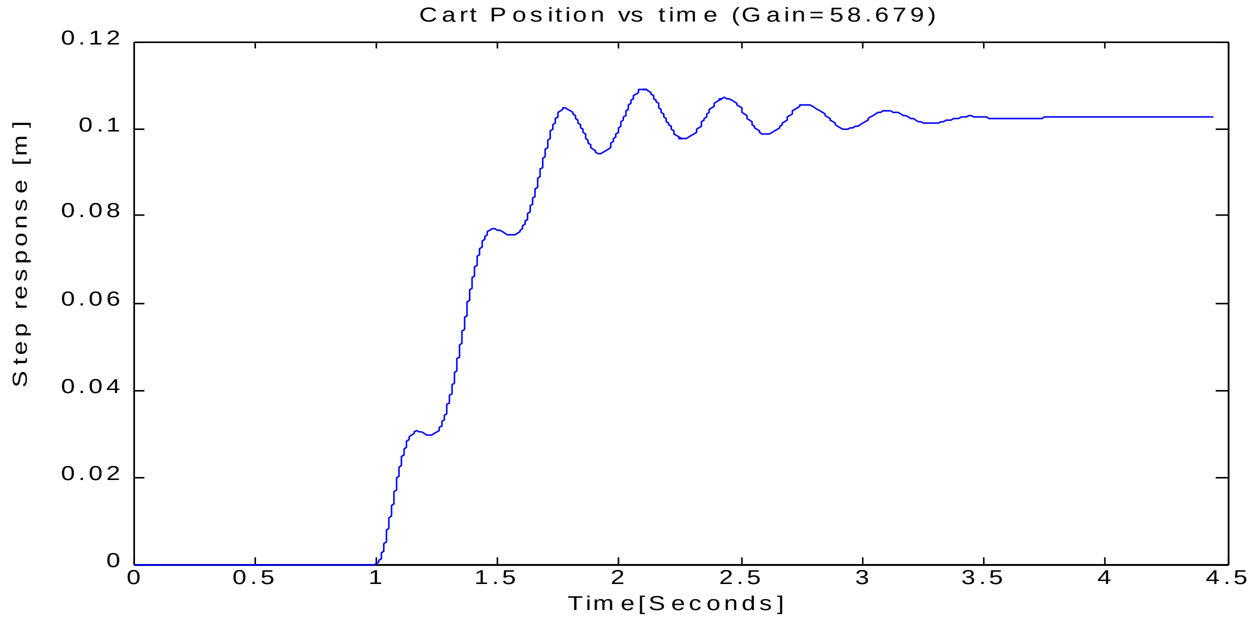


Figure 12: Step response for the second-order controller of X2(s)/V(s)

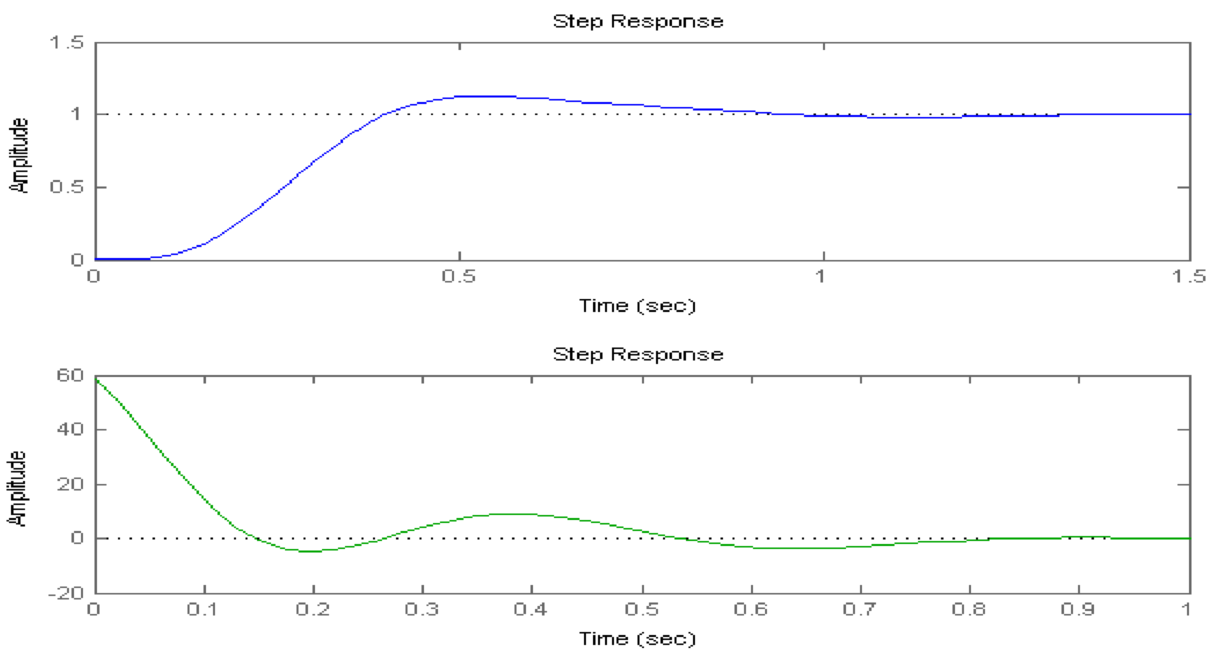


Figure 13: Simulated response for the second-order controller of X2(s)/V(s) and the voltage input corresponding to a step size of 1.

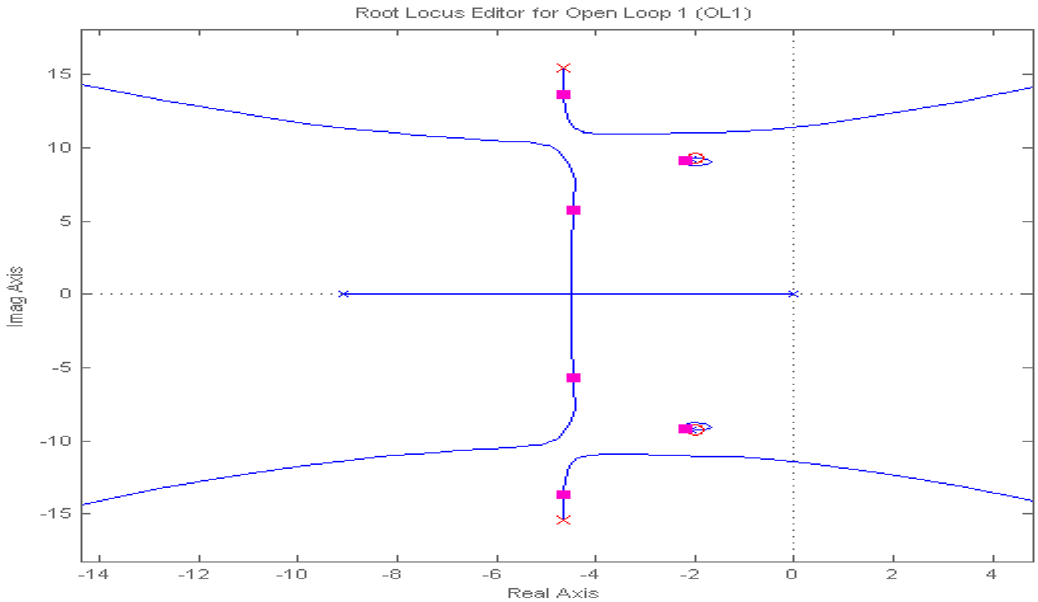


Figure 14: Root locus plot to find controller traits of second-order controller for X2(s)/V(s).

2.3 The poles for the third order controller, found using root locus techniques, are -7.18 and -2.2 ± 9.2j. The zeros were found at 9.37 and -5.6 ± 20.05j, and the gain was calculated at about 140.6. These added zeros fixed the overshoot, but did little to quicken the rise time or settling time of the step response.

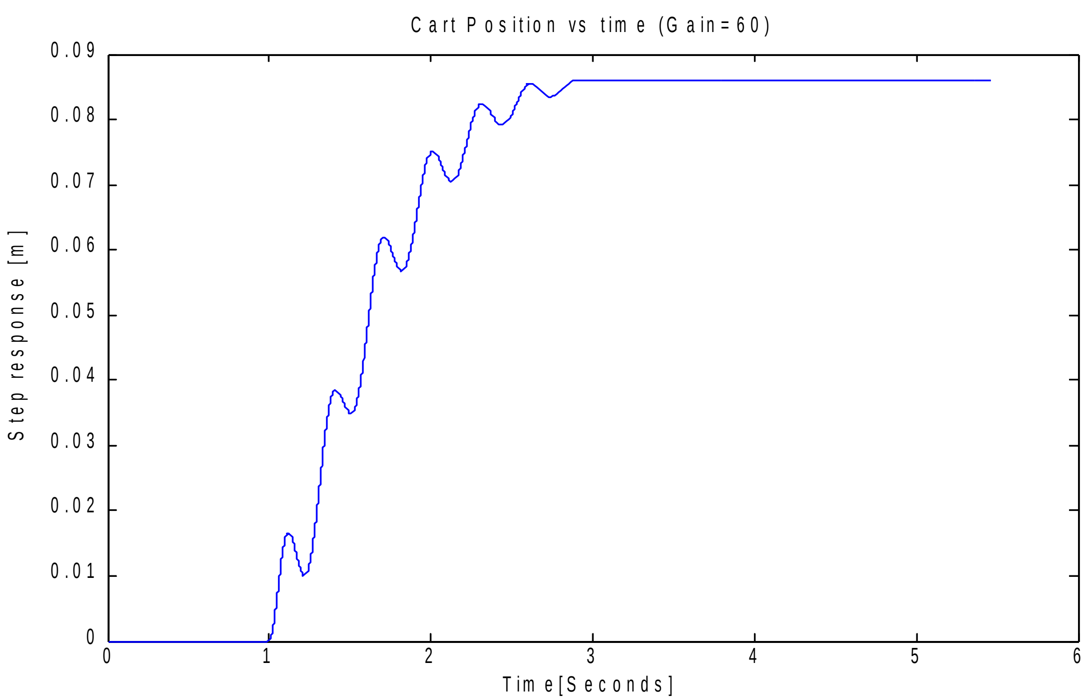


Figure 15: Experimental step response of third-order controller for X2(s)/V(s).

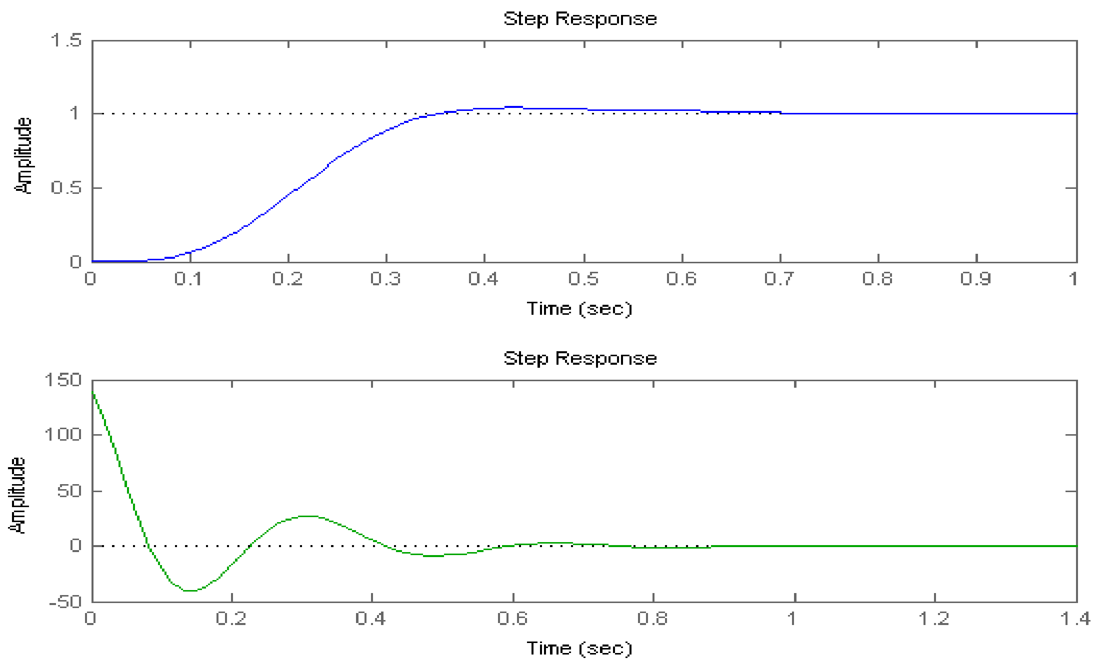


Figure 16: Simulated step response of third-order controller of X2(s)/V(s).

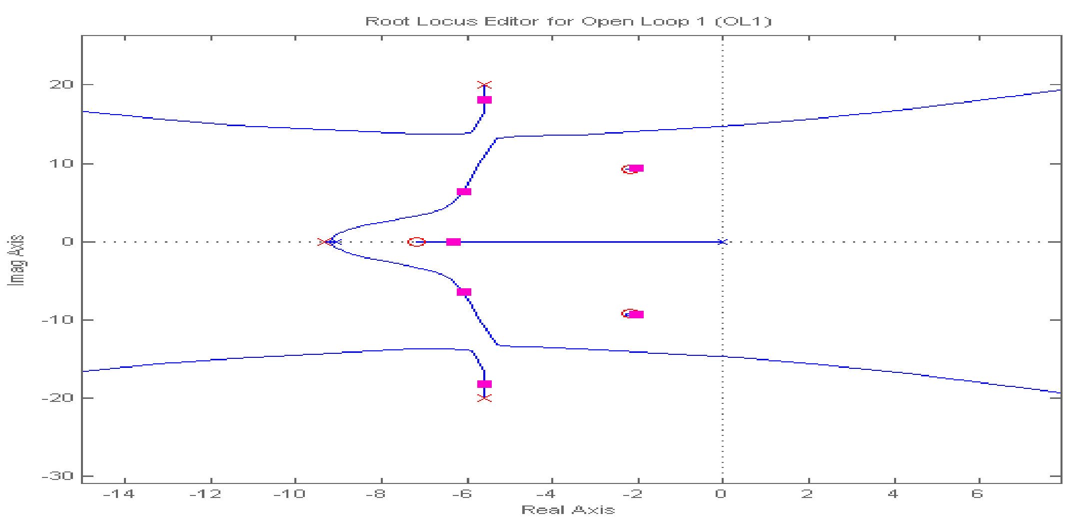


Figure 17: Root locus plot used to find possible third-order poles, zeros, and gain.

Conclusion

Based on findings from the lab, I conclude that increasing the order of controllers does not necessarily improve the response. Even if the simulated response has better characteristics, often there are factors in real implementation such as phase delay and minimum voltage requirements to overcome friction. It is desirable to solve a problem with lesser degrees of uncertainty and therefore lower-order controllers; one must weigh the gains of adding pole-zero orders to the losses in robustness.