ECE 311 - Lab 3

Shadman Kaif

09/12/2020

1 - INTRODUCTION

In this section, you will summarize the main purpose of this lab and briefly describe the experiments you performed.

The purpose of this lab is to design a cruise control system for a car using satisfying certain technical specifications. In particular, using the model estimation you completed in the previous lab, I designed a PI controller using the root locus method. I found the gain and T_I value associated with poles being situated near s = -20 and s = -30. This led me to design the normal and aggressive controller, in which I was able to find settling time, and effects with and without disturbance.

2 - EXPERIMENTS

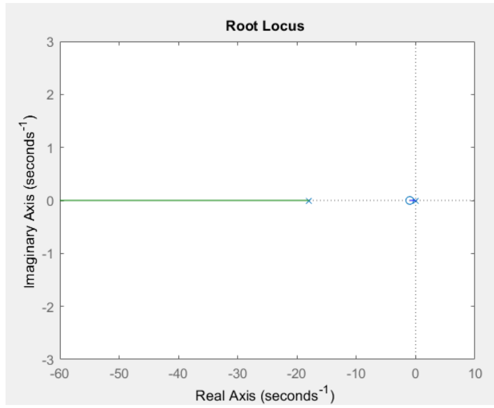
In this section you will report all your experimental simulation: discuss your hypothesis and results.

2.1 CONTROLLER DESIGN USING MATLAB

In this section you will describe all the steps performed to design your controller using the root locus method with MATLAB.

Be sure to include the following material and discussions:

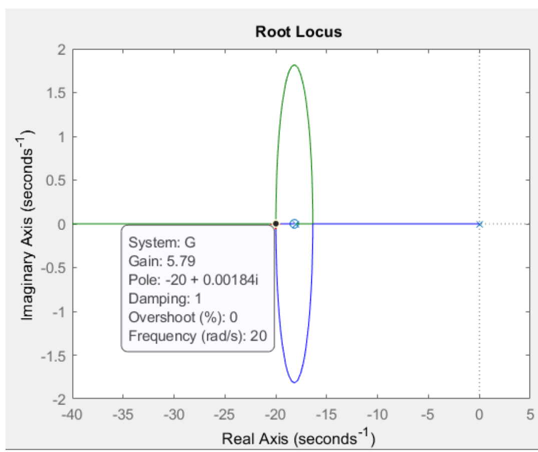
1. Your first root locus plot with the initial choice of T_I=1. Why there doesn't exist K > 0 such that the closed-loop has two poles on the real axis with real part ≤ −20? Justify your answer using the root locus plot.



The green and blue lines represent the values of the poles. The values for the green line pole encompass around -18 to negative infinity, while the values for the blue line pole encompass 0 to -1. Thus, while the green line pole can take a value lower than -20, the blue line pole can never take on that value regardless of the varied gain.

2. Your final choice of T_I and the corresponding root locus plot.

Final T_I = 0.055

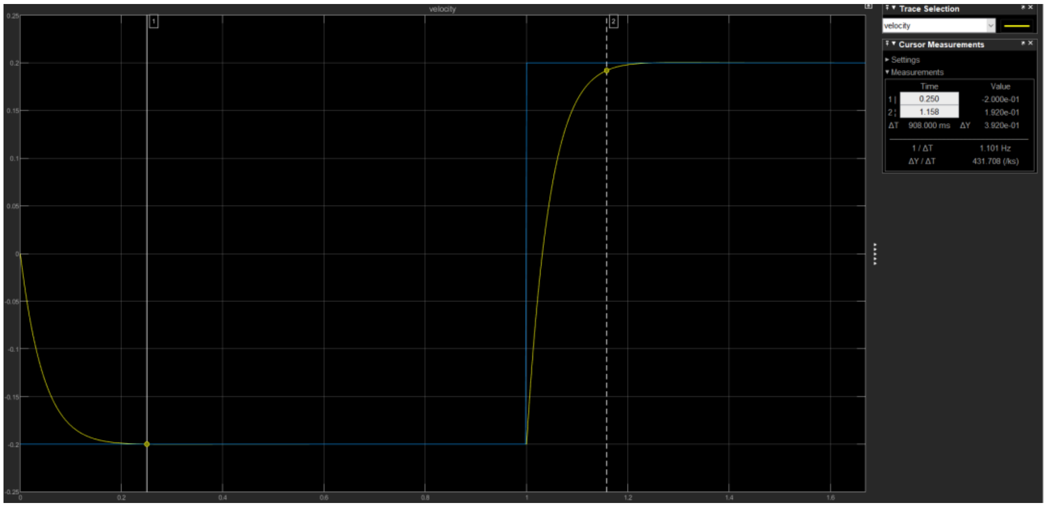


3. The values of *K* you obtained using the function **rlocfind.**

Using rlocfind, as seen from the plot above, the gain value, K, attained was 5.79.

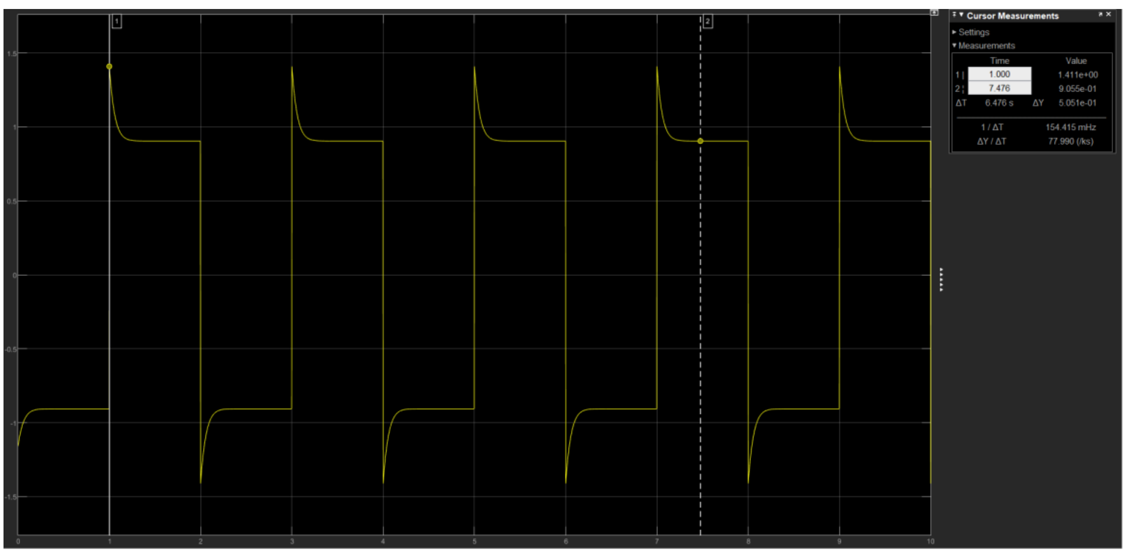
4. Report the response of the closed loop system with the PI controller. What is the settling time T_s you estimated from your plot? Is the control specification **approximately** met?

T_s = 0.158 seconds based on the plot and after 0.158 seconds, the plot falls under the range [0.192, 0.208] which was previously calculated by using v(inf) = 0.2 and v(0) = -0.2 for the range given. Controller specification 4 is approximately met as 0.16 seconds is less than 0.2 seconds.



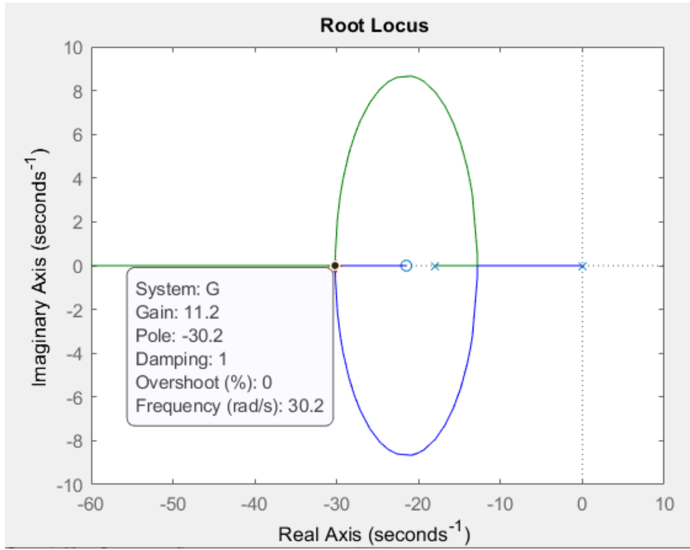
5. Is SPEC5 met? Include the plot of your voltage input to support your claim

   As we see, the magnitude voltage is much lower than the SPEC5 maximum of 11.75V; therefore SPEC5 is met.



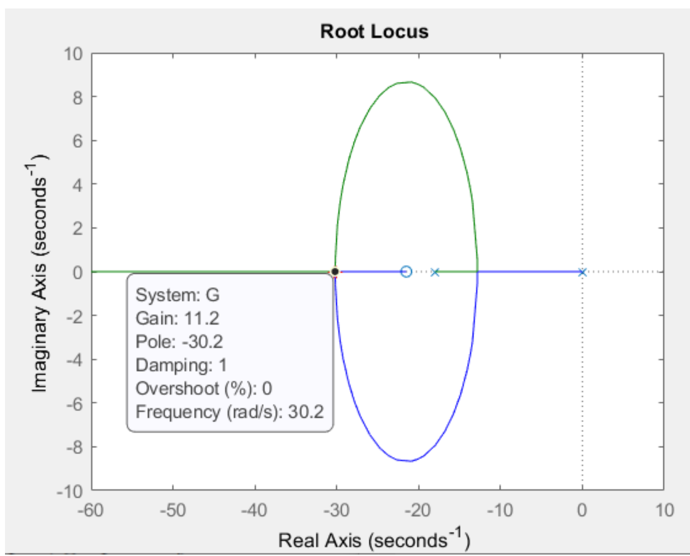
6. Report here the value of T_I such that there exists $K > 0$ such that the closed-loop system has two poles close to $s = -30$ and the corresponding root locus plot.

   $T_I = 0.465$



7. Your final choice of T_I and the corresponding root locus plot.

T_I = 0.465



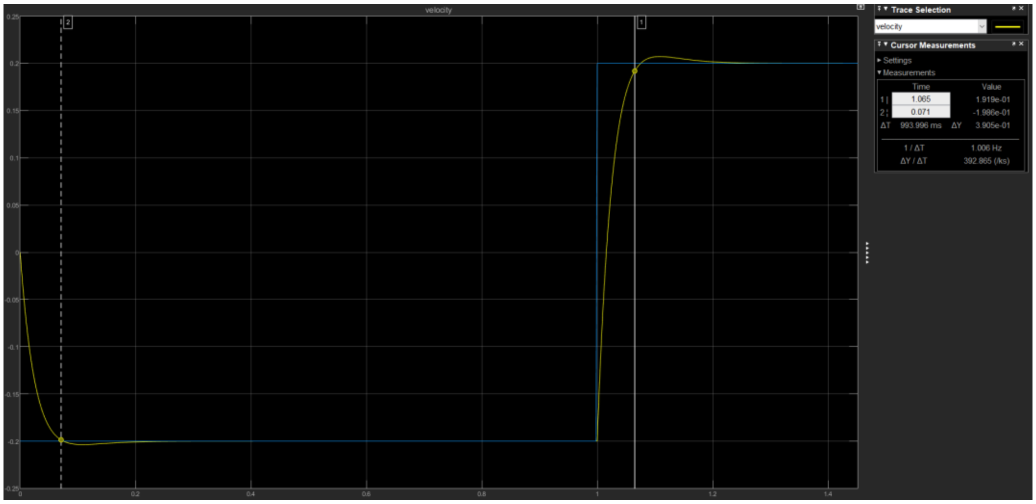
8.

9. The values of *K* you obtained using the function **rlocfind**

K = 11.2

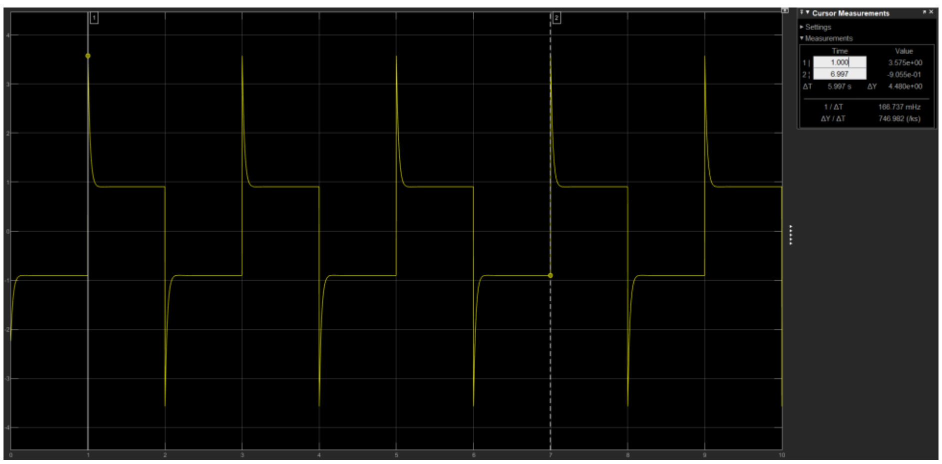
10. Report the response of the closed loop system with the new PI controller. What is the settling time T_s you estimated from your plot? Is the control specification **approximately** met?

T_s = 0.065 seconds based on the plot and after 0.065 seconds, the plot falls under the range [0.192, 0.208] which was previously calculated by using v(inf) = 0.2 and v(0) = -0.2 for the range given. Controller specification 4 is approximately met as 0.065 seconds is less than 0.2 seconds.



11. Is SPEC5 met? Include the plot of your voltage input to support your claim

    As we see, the magnitude voltage is much lower than the SPEC5 maximum of 11.75V; therefore SPEC5 is met.

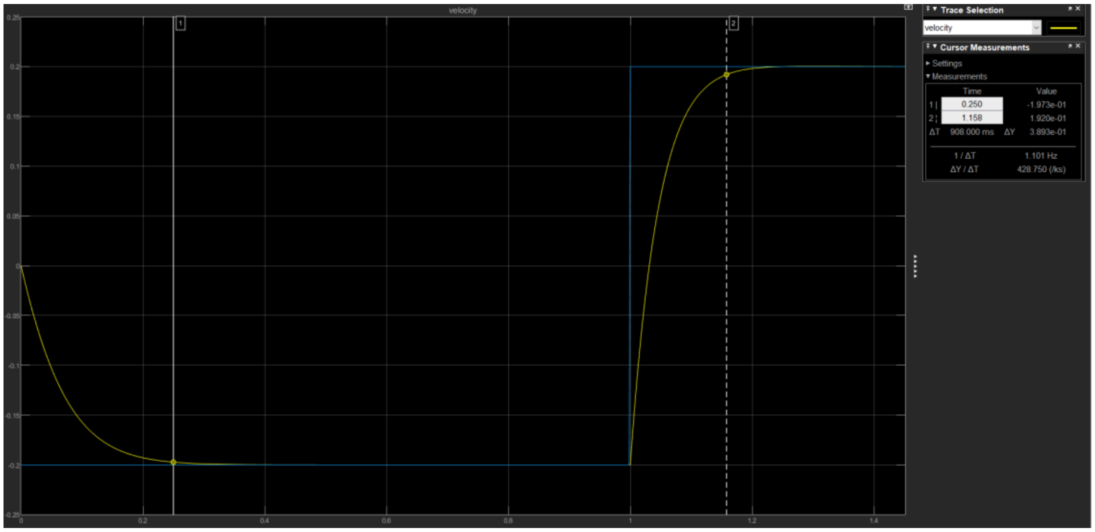


2.2 REJECTION OF DISTURBANCES

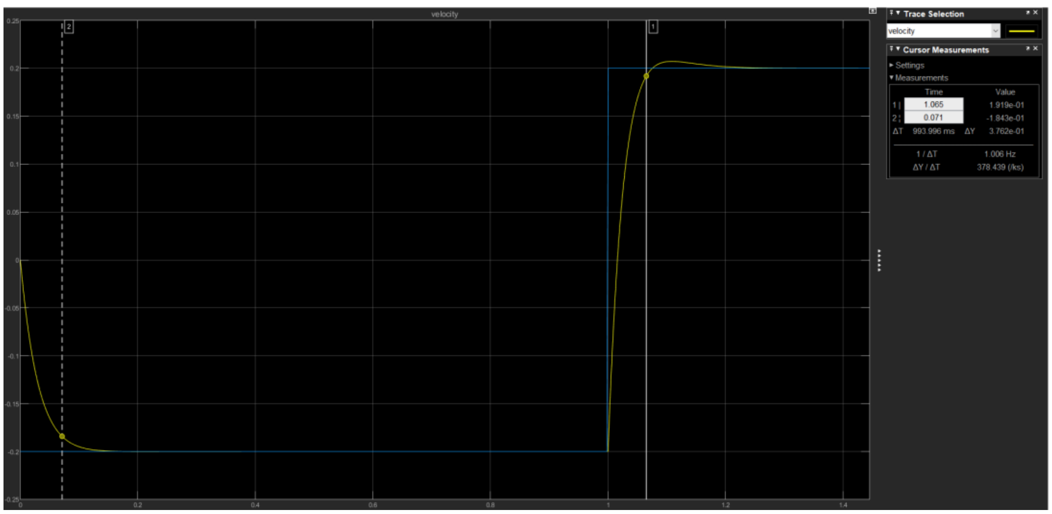
Be sure to include the following material and discussions:

1. Include the closed loop responses for both controllers

   Controller with T_I = 0.055 and K = 5.79:

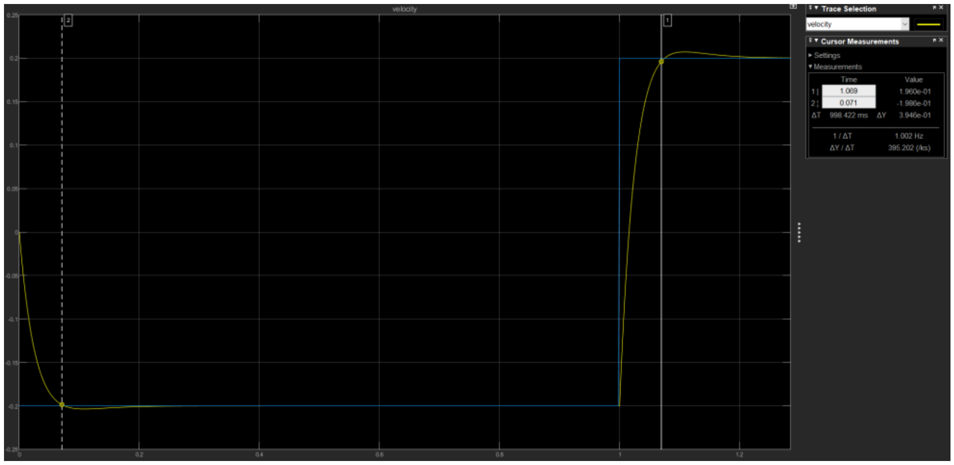


Controller with $T_I = 0.0465$ and $K = 11.2$:

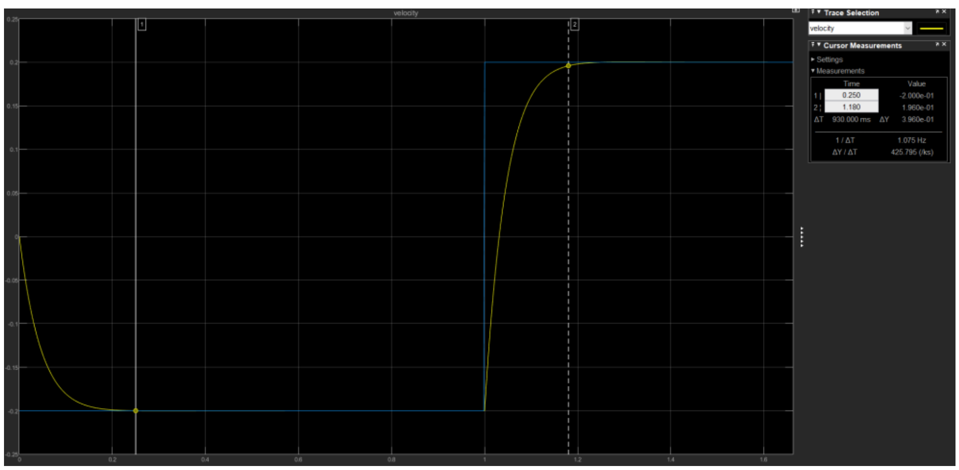


2. Are the two controllers able to reject the disturbance?

Yes the controllers are able to reject the disturbance for the most part. There is one slight difference between the velocity responses with and without disturbances for both controlllers. The beginning of the velocity response is different. Without the disturbance, this is the plot of the aggressive controller:



As we see, the velocity response shoots below the signal generator response. This is not noticed for the same configuration with a slight disturbance. Zero disturbance ensures that the velocity response goes to the -0.2 mark quicker, but causes it to shoot slightly below the -0.2 mark. For the normal controller, the plot without disturbance is displayed below:



Feel free to juxtapose with the plots shown in Question 1 of 2.2 to confirm as I have done. As you can see, it takes longer for it to stabilize at the -0.2 mark with the disturbace. This is consistent with the aggressive controller as well.

3. How the two controllers compare, with respect to disturbance rejection? Support your arguments referring to the velocity responses you obtained.

   In terms of disturbance rejection, both controllers performed equally well. They provided us with similar velocity responses with the disturbance as without the disturbance. This can be observed as the period remains the same and furthermore the settling times for both controllers before and after the disturbance stayed the same. However, as pointed out before, the initial dip to -0.2 to emulate the signal generator slightly varied with and without the disturbance. We see that it stabilized quicker without any disturbances than with the disturbance.

3 - CONCLUSIONS

In this section you need to answer the following questions:

1. How do the settling times and overshoots compare?

   The settling time for the controller with $T_I = 0.055$ and $K = 5.79$ had a settling time of 0.158 seconds while the controller with $T_I = 0.0465$ and $K = 11.2$ had a settling time of 0.065 seconds. The aggressive controller thus had a lower settling time. However, the aggressive controller had a higher overshoot, peaking over the signal generator higher than the normal controller.
2. Which controller is best suited to meet the specs?

   Both controllers are well-suited to meet the specs. They both meet SPEC4 and SPEC5 as previously discussed with their $V_M(t)$ peak being lower than 11.75V and settling times being lower than 0.2 seconds respectively. I believe that the normal controller is slightly better suited to meet the specs. Although the aggressive controller has a lower settling time, which is desired, it also has a higher overshoot and peak $V_M(t)$. Thus, it is more beneficial to go with the normal controller which is well below the 0.2 second settling time maximum. Drawing less voltage is more ideal as it mitigates chances of being bottlenecked.
3. What is the cause of the differences you observe?

   The aggressive controller has poles close to $s = -30$ while the normal controller has poles around $s = -20$ which means that the aggressive controller will have a faster decaying exponential compared to the normal controller, resulting in a lower settling time as seen in the experiment. However, in order to get poles at $s = -30$, the aggressive controller required a greater gain, K, compared to the normal controller (aggressive needed $K = 11.2$, whereas normal needed $K = 5.79$), resulting in a greater overshoot value and greater peak value of $V_M(t)$.
4. Are both controllers able to reject the disturbances?

   In terms of disturbance rejection, both controllers performed equally well. They provided us with similar velocity responses with the disturbance as without the disturbance. This can be observed as the period remains the same and furthermore the settling times for both controllers before and after the disturbance stayed the same. However, as pointed out before, the initial dip to -0.2 to emulate the signal generator slightly varied with and without the disturbance. We see that it stabilized quicker without any disturbances than with the disturbance.