EEN G307: Intro to Feedback Control

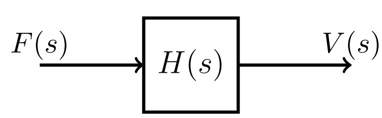
Fall 2020

Homework Assignment #12

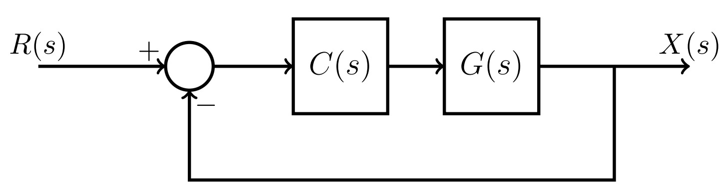
Due: Wednesday, Dec 9th, 11:59pm

1. Determine the transfer function of a Honda Element using force of the engine as input and the positon of the car as output.
   (a) Draw the model as a translational mecahnical system with a mass and an air resistance modeled as a dampener.
   (b) The mass is 4,400lbs, convert this to kg to determine m. To determine the drag coefficient b, you will first need to find the maximum force generated by the engine using $F = ma$. To determine the acceleration, use the stated 0 to 60mph time of 9.7 seconds. Make sure to use SI units. Once you have the force determine b by examining the sum of the forces on the Honda Element at its terminal velocity of 100mph. Make sure to use SI units, round your values to 2 significant figures, and represent your constants in scientific notation.
   (c) Use the model to determine the transfer function for $\frac{X_{out}(s)}{F_{in}(s)}$. Substitute in your values for m and b.
   (d) Using the previous transfer function, write out the transfer function for the velocity of the Honda Element; $\frac{V_{out}(s)}{F_{in}(s)}$.
2. Use your model to produce the step response (in Matlab) of the Honda Element's velocity vs time under full acceleration. This means that you should multiply the step input by the maximum force generated by the engine. From this plot, determine the 0 to 60mph time for the model Honda Element as well as the top speed (convert to mph). Expect some discrepencies from the actual values given in a previous step.

   To be clear, you are looking at the open-loop step response of your model as illustrated below. In this figure $F(s)$ is the step input representing a maximum force being applied to the Honda Element. $H(s)$ is the transfer function for the velocity and $V(s)$ is the velocity of the Honda Element.



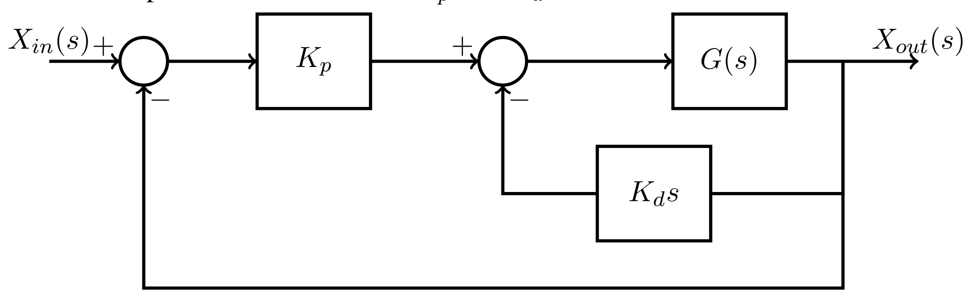
3. Build a controller, $C(s)$ in the figure below, to automously park the Honda Element, given by $G(s)$ in the figure below. The control input will be a command to move forward 10 meters (into the parking space). The design requirements of this controller are a 10% overshoot and 10 second settling time.



   The input to the control system is $r(t) = 10u(t)$. The Honda should have zero steady state error and very little overshoot because we do not want it running into objects beyond its parking space.

(a) Use sisotool to build the root locus for a proportional controller. Overlay the given design requirements to show that a proportional controller is not feasiable. Screen shot the Root Locus Editor for LoopTransfer_C as the answer for this question.

(b) Use the sisotools to determine the gains for a PD controller using the techniques discussed in class. $G(s)$ is the position transfer function derrived earlier. Aim for 0% overshoot by just getting the poles on the real axis. Choose a zero location that produces an integer term in the $C(s)$ equation. Likewise choose a integer value for the gain.
You answer for this question has two parts, a screen shot of the Root Locus Editor for Loop-Transfer_C plot and the values for K_p and K_d.



(c) Derrive the transfer function for the PD controlled system, with target position $X_{in}(s)$ as the input and the actual car position $X_{out}(s)$ as the output. Use the values for K_p and K_d.

(d) Use this transfer function to predict the rise time, settling time and percentage overshoot.

(e) Show the step response for the parking problem your PD controller.

(f) List the rise time, settling time and percentage overshoot of the step response of the PD controlled system using the stepinfo command in Matlab.

Solutions:

1. (a) Model not shown.
   (b) $m = 2.0 * 10^3$kg and $b = 1.2 * 10^2$
   (c) Answer not provided.
   (d) $\frac{V_{out}(s)}{F_{in}(s)}$ is a first order system.
2. Step response not shown.
   The 0 to 60mph time is estimated to be 10 seconds.
   The top speed is 45m/s = 100mph.
3. (a) Root loci with design requirements plot from sisotool not provided.
   (b) Root loci with zero added with design requirements plot from sisotool not provided. Controller values will vary slightly due to choices in zero location and gain.
   (c) The transfer function will be a second order system, $\frac{X_{out}(s)}{X_{in}(s)} = \frac{\omega_n^2}{\alpha s^2+2\zeta\omega_n s+\omega_n^2}$ You will need to make sure to normalize this transfer function by dividing top and bottom by α because the coeficient in front of the s^2 term is not 1.
   (d) Even with a different controller, the performance values that I got should be close to yours. $t_r = 4.7\text{seconds}$
   Since my $\zeta \geq 1$ then my controller is over-damped. This means that the poles are real and that the overshoot = 0%. $t_s = 9.8\text{seconds}$
   (e) PD controlled step response not shown.