ESA Rocky Individual Reflection

Nabih Estefan

March 20, 2021

At the start of the Rocky project, my confidence was low. It was a mix of me having worked little with moving robots in the past (I've worked with Arduino, but it was mostly static robot with preset moves, not a moving one that reacted to its surroundings) and little feedback from our previous assignments. I was pretty sure that the work I had done for Homework and the Quiz but we hadn't received most of the feedback by the time the project had started, so my confidence that this work was actually good wasn't as high as I would've liked. In comparison, at the end of the project, not only had I gotten feedback back on my previous work, I had also gotten the Rocky to stand up by itself and was able to understand no only the Arduino code, but also the logic and code needed to calculate the Controller Parameters.

Over the project, I grew a lot as a practicing engineer in applying analytical tools. One of the things I learned the most was PI control, and how to use different PI controls to stabilize our Rocky. Before the project we had only done system with single PI controls, but the project introduced me to not only multiple PI controls in a single system. Not only did I learn how to use multiple PI systems, I also learned how to combine these into single equations, instead of having two separate PI controls, I learned how to combine them to have one single control that has 2 degrees of change (s and s^2).

At the start of the project, i thought my ability to use MATLAB was pretty advanced, but somehow throughout the project, I realized that it was not as good as I thought, and learned a lot on new things to do in MATLAB. Not only did I learn how to use functions like `fplot()` and `fit()` better, I also learned how to use symbolic equations better and how to use polynomial (and also other types of equations) to find the variable values of the previously mentioned symbolic equations. Besides that, I also learned a lot of Arduino code. My partner and I tried to get Rocky to turn and go straight on certain time-steps, and all our tweaks and testing led to both of us learning a lot about the code and how it works, which developed my knowledge of Arduino considerably.

I also learned a lot of connecting theory and experiment. Besides the logical one, like calculating natural frequency by measuring as we had done in homework before with data, or combining our symbolic MATLAB calculations with our actual Rocky programming, one of the biggest moments where I felt I connected theory and experiment was when calculating the poles of the main Rocky transfer function. We were in a hassle of how to get these values, but we had decided that we wanted a slightly under dampened system. After a lot of trial and error, my team remembered a diagram we had been showed in class that related angle, damping, and natural frequency to the real and imaginary values of poles. At this moment, we realized that if we decided what angles we wanted our Rocky to normally oscillate at when pushed, we could calculate values for our poles, and then use these to calculate our controller parameters.

Finally, I really enjoyed the project overall. Being able to work with actual, physical items was a very refreshing change of pace in Zoom-land, and besides that the project was really fun. Even though we didn't do much actual code to the Arduino, having Rocky finally balance on his own was a very rewarding experience (besides the fact that it looks great on portfolios) and I had a lot of fun coming up with ideas of what to do in our video. Overall, I really enjoyed the project.