Alex Nguyen
Suhail Chakravarty
ECE/ME 179D

Lab 1B: Trajectory Planning with Two-Link Arm

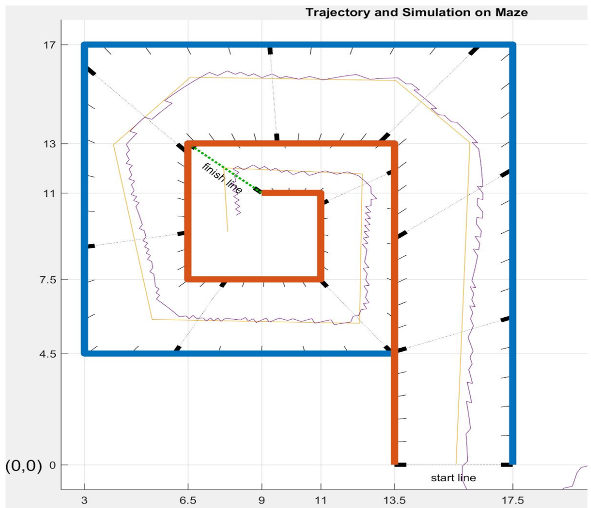

Figure 1: Planned Trajectory Plot (Prelab 1b)

Figure 1, above, shows the intended path the laser should track during the lab session. The plot was generated using ginput() to specify points in the maze, then the points were interpolated, between each other, to obtain the intended path to follow. Initially, the spline method was implemented to output a spiral-like path, but this did not yield appropriate results, so we used lines to connect the points in the maze instead. The K_p and K_d values were tuned for the "Lego Mindstorm Motor" simulink block, which emulates the output of the actual Lego Mindstorm motors. These proportional constants were changed for our actual lab where we used two motors from the Lego Mindstorm Kits. The values for the prelab were $K_p = 170$ and $K_d = 10$.

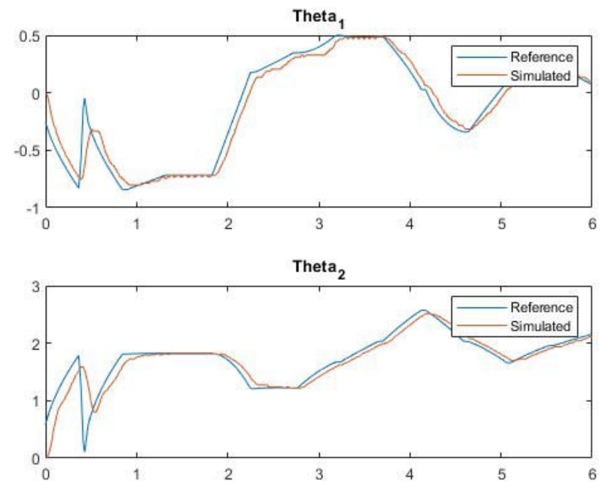

Figure 2: Angle Tracking for Joints 1 & 2

Figure 2, above, shows the tracking of the motor joints to the reference signal provided in the simulink model. The Lego Mindstorm Motors had some resistance when they were rotated by hand. This led us to creating two different proportional and derivative constants for the motors. Joint 1 needed higher proportional constant values than motor joint 2 because it had more friction from the external weight when turning the motor. The constant values we chose ended up being: $K_{p1} = 170$, $K_{d1} = 10$, $K_{p2} = 265$, and $K_{d2} = 8$. Although the data output from the simulink model showed very good tracking, the actual tracking of our laser pointer was erroneous. Also, our group had trouble figuring out the best way to position the two motors since the links were heavily affected by gravity due to the weight applied at the end effector by the laser pointer. Ultimately, the motor was clamped to the table to point the laser at the ground rather than on the wall. The constants could be tweaked a small amount, but the tracking output is sufficient (in theory) for the maze tracking.

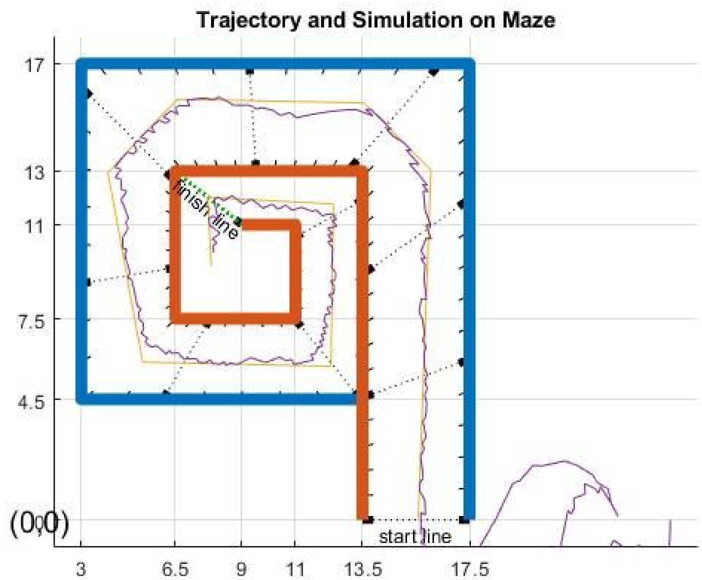

Figure 3: Simulated Trajectory Output Through Maze

Figure 3, above, shows the intended path of the laser (yellow) with the actual path the laser tracked during the lab (purple). The data output from the simulink model showed strong tracing of the intended path. However, as can be seen in the video, the reality was that the laser did not track the reference nearly as well shown. In the beginning, there is always some small shaking of the two links attached to the motors in order for the end effector to get to the start of the maze. Initially we were having issues with runtime since the trajectory calculation in real time had too much overhead and caused severe timing issues. This problem was fixed using precalculated trajectories. Eventually, our best run of the actual laser tracking, ended up yielding ~ 60% completion of the maze within a time of 2 secs. The laser would cut a corner once it got to the leftmost pathway in the maze even when attempting to adjust the points such that there is more buffer. This could be due to the noise/disturbance of the Lego Mindstorm motor, which yields poor tracking in application (since the theoretical and actual plot vary vastly) and additionally issues from the additional torque applied by the environment.