ECE 5463 Final Project Report

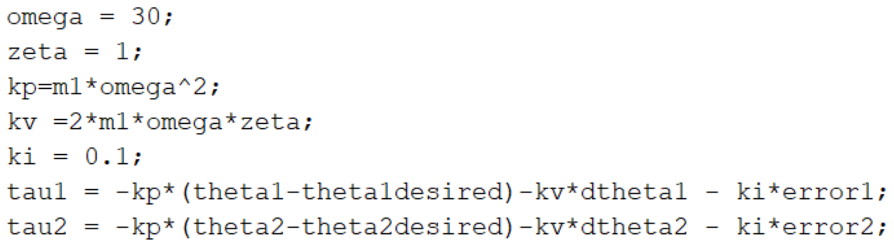
Date: 12/09/2022
Wafeeq Jaleel.5

PD vs PID

I chose the PID controller for this simulation because it has a lower steady-state error and more flexibility compared to the PD controller. Since there are two links to control, PID was more reliable in providing the expected output.

Parameter Tuning

I used zeta and omega to determine the values of kV and kp. One of the requirements was to keep zeta 1. Initially, I had omega as 10, where I observed the arm did not reach the endpoint. After increasing the omega values by a factor of 5, I settled with 30. The arm required less time and was accurate when reaching the end effector. Tested ki for higher values as 50 and 100. It led to an overshoot of the desired theta values. So kept decreasing until the response plot had no overshoot and ended up with 0.1 as the final value.



```
omega = 30;
zeta = 1;
kp=m1*omega^2;
kv =2*m1*omega*zeta;
ki = 0.1;
tau1 = -kp*(theta1-theta1desired)-kv*dtheta1 - ki*error1;
tau2 = -kp*(theta2-theta2desired)-kv*dtheta2 - ki*error2;
```

Figure 1: Parameter values

Response plots

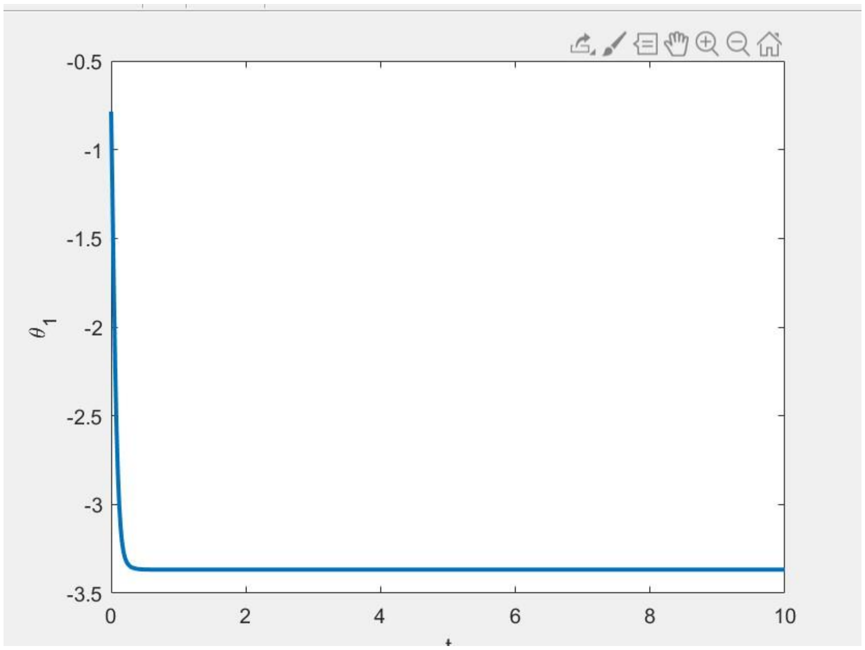


Figure 2: Theta1 vs t plot

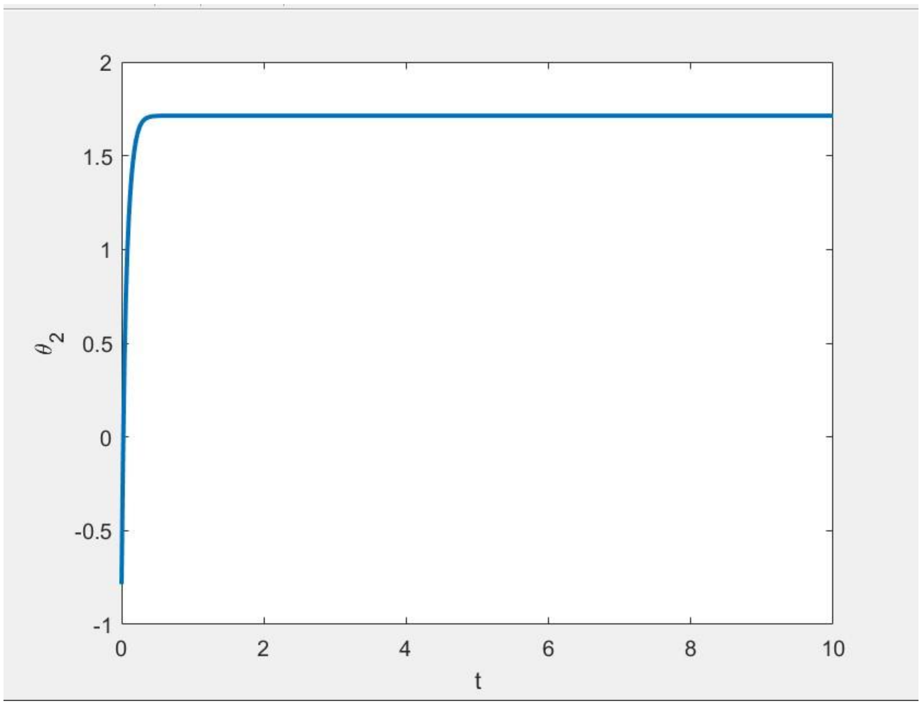


Figure 3: theta2 vs t plot

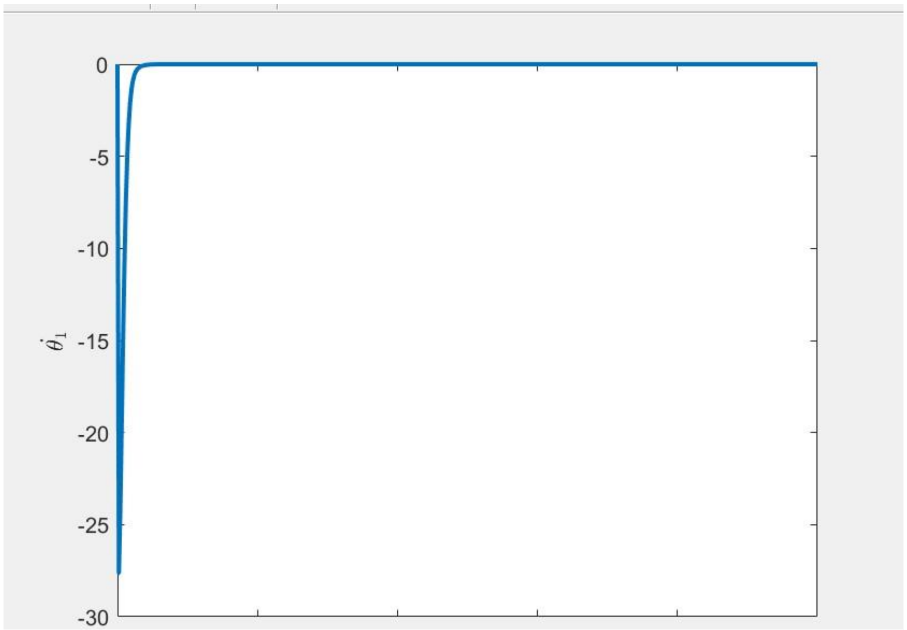


Figure 4: First Arm velocity

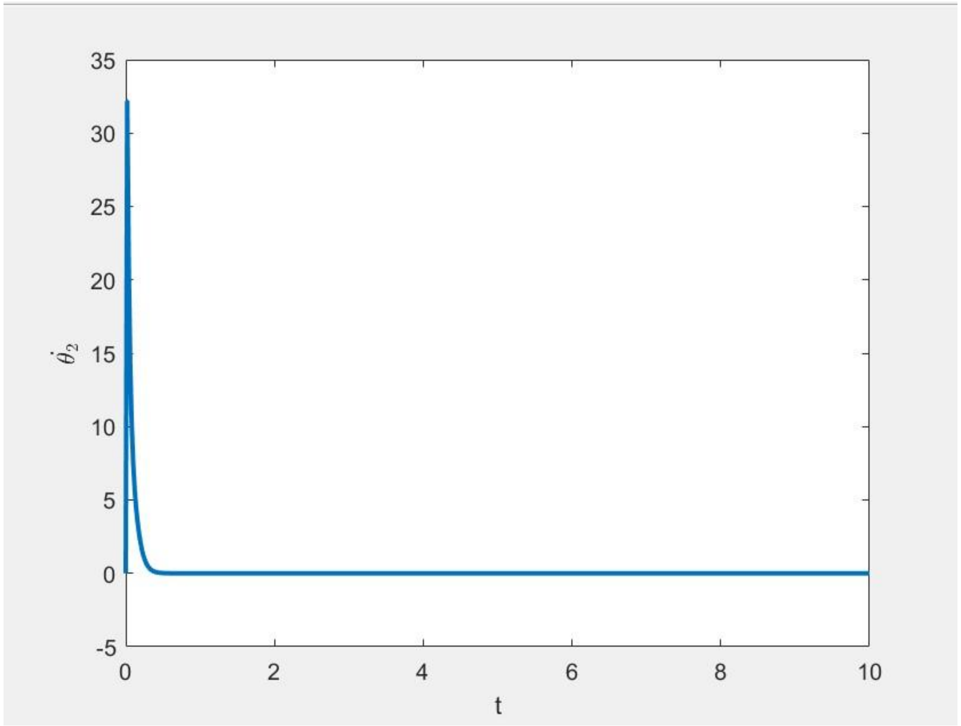


Figure 5: Second Arm velocity

Theta1 and Theta2 reached the desired value as shown in Figure 2 and Figure 3. Confirmed it by looking at the values calculated from the inverse kinematics formula. Theta 1 value is negative because the first arm rotated clockwise. There was no steady-state error in both theta 1 and theta 2. Velocities of both arms shot to around 30 rad/sec before settling back to 0.

References

[1] HW 4 Solution from Dr. Ziaeefard

[2] CP17 code from Dr. Ziaeefard