Alex Nguyen
Suhail Chakravarty

Lab 1A Two Link Arm: Draw a Triangle with a Laser Pointer

The lab had our group use actual motors from the lego mindstorm kits, while the prelab utilized a block diagram created to simulate the lego mindstorm motors. The prelab simulink model was almost identical to the lab simulink model, with the difference being changing the proportional constants to produce a smooth tracking of the reference signal and having to adjust the lengths for our arms. For our casual velocity filter, the value f was kept the same but the velocity signal output was noisy when plotted. Our group needed to configure the setup of the lego motor, as done in lab 0, before starting the lab. The prelab needed no setup, just to tune parameters for the constructed simulated lego motors.

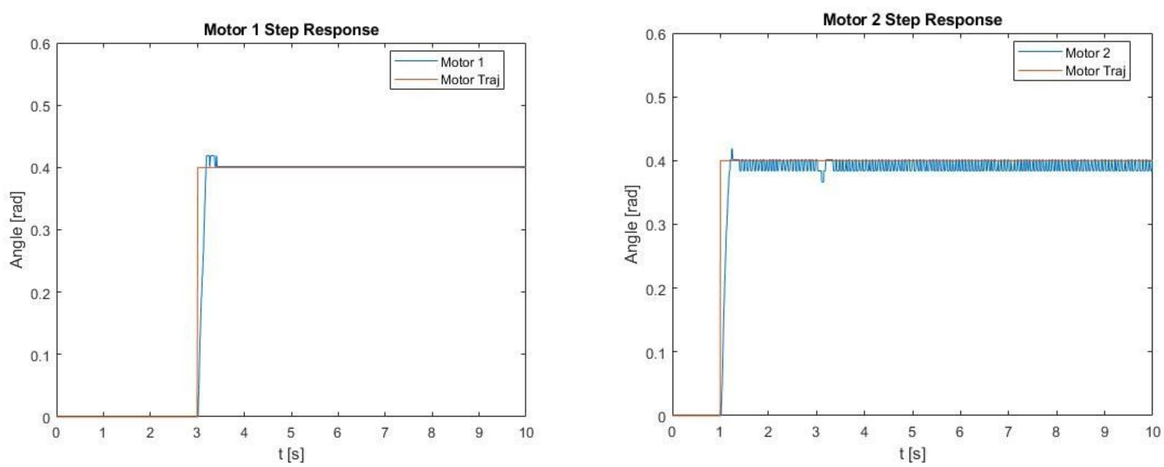

Figure 1: Step Reference with Magnitude of 0.4 Radians

Figure 1, above, shows the results of testing of our group's two motors to ensure an accurate tracking of a reference signal. As seen above, both motor joints track the reference an ample amount. The tracking signal produces a small amount of overshoot for both motors, but the steady-state value reaches the intended 0.4 radians (about 23°) in a small amount of time. There are oscillations for motor 2, but this could be due to a noisy motor since the motors were originally from kid's LEGO Mindstorm Sets and the noise is motor 2 can also be amplified in the noise from motor 1. For the tuning parameters, the derivative constant was kept the same for both motors but we used different proportional constants for both motors. This was because motor 1 was very damped and resisted motion when the proportional constants were kept the same. The final tracking values for the reference signal were $K_d = 0.7$, $K_{p1} = 330$, and $K_{p2} = 100$.

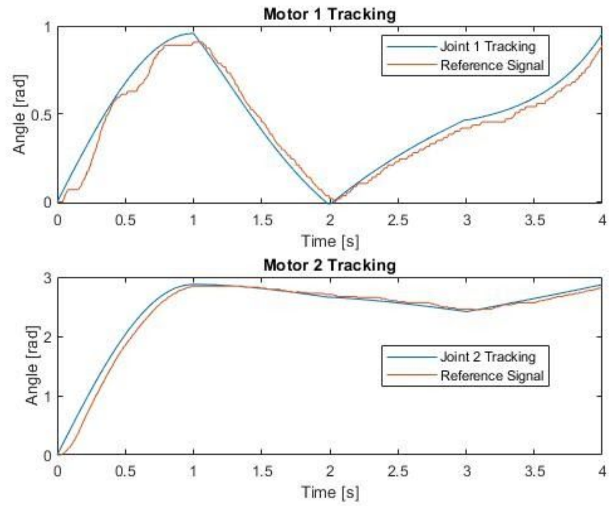

Figure 2: Reference and Tracking Signal for Joints 1 & 2

Figure 2, above, shows how well our motor joints tracked the intended (theoretical) path of the laser written using a MATLAB function in a simulink model. The simulink model was constructed to move a two-link arm to draw an equilateral triangle using a laser pointer. The tracking shown above, figure 2, of the intended versus actual angles by our joints are close. There are some rocky points, but this could be due to the noise or disturbance of the motor. The noise could have a fixed value of error after the motor was constructed, while the disturbance could arise from the environment (shaky table, motor not clamped enough, sticky/hard to move joints, ect.). Also, the tracking could have been better if our group allowed the motor to slow down at the corners when tracing the triangle.

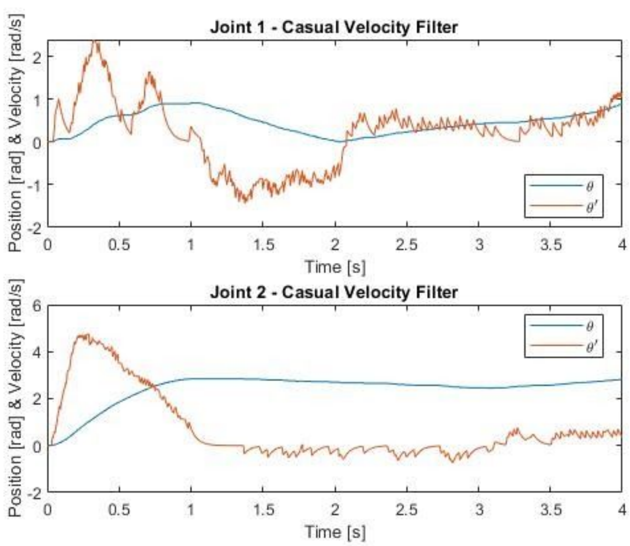

Figure 3: Casual Velocity Filter Plot

Figure 3, above, is the result of the casual velocity filter plotted with the angle position. As seen in figure 4, The casual velocity filter was built using as a function of the term f which was chosen to be 0.9. There is another term A which is a function of f, which ended up being 20. The filter resulted in a noisy signal so the value for f could be tuned to reduce the noise output. The noise could also be a result of the motors used in lab. A test to see if the velocity signal is correct would be to curve fit the position plot to obtain an equation. Next, take the derivative of the position plot's equation then plot the derivative as a function of time on a graph. Another check would be to see where the velocity is zero, which would be the flat sections of the position plot. Our velocity plot seems reasonable because the flat portions of the position graph have an output velocity value of zero.

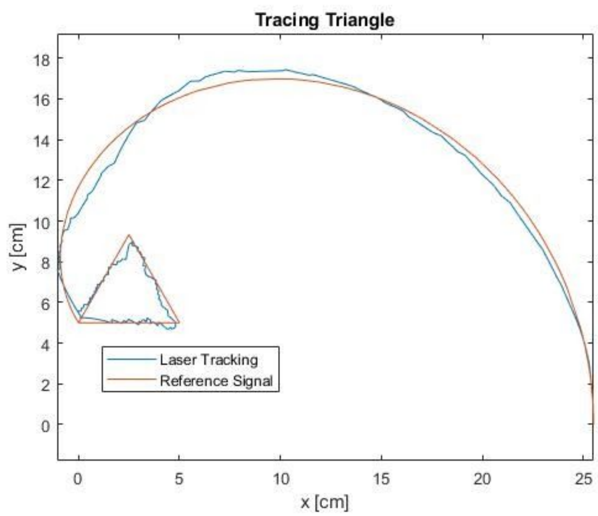

Figure 4: Plot to Trace Triangle

Figure 4, above, shows the reference signal (generated in MATLAB) with the path traced by a laser plot superimposed on it. The figure shows good tracking of the reference signal from our generated data, but in our lab (as seen the submitted video) the laser did not track the equilateral triangle well at all. The poor tracking, in the lab, could be due to wrong length specifications in our simulink model since there was no really precise way to accurately measure the linkages. Another source of error could be due to the motors used in lab, where they could be built with some finite degree of precision. Given that other groups in our lab were getting similar issues with their results, it points to a consistent issue with the motors. A way to reduce the noise is to ensure the correct lengths are measured, use a better motor, and slow down the motor at the triangle corners by tuning the proportional constants to better values. The proportional constant values used were K_{d1} = 1.3, K_{d2} = 2.25, K_{p1} = 460, and K_{p2} = 190.

MATLAB Appendix

```
clc; clear; clf;

load('lab1aDataDumpP2.mat')

%% Defining Variables
t1 = my_data.time; %time for motor 1 [s]
t2 = my_data2.time; %time for motor 2 [s]
traj1 = my_data.signals.values(:,1); %angle 1 reference signal [rad]
traj2 = my_data2.signals.values(:,1); %angle 2 reference signal [rad]
cvel1 = my_data.signals.values(:,2); %casual velocity filter for angle 1 [rad/s]
cvel2 = my_data2.signals.values(:,2); %casual velocity filter for angle 2 [rad/s]
pm1 = my_data.signals.values(:,3); %joint 1 input signal to motor
pm2 = my_data2.signals.values(:,3); %joint 2 input signal to motor
ang1 = my_data.signals.values(:,4); %tracking of joint 1 [rad]
ang2 = my_data2.signals.values(:,4); %tracking of joint 2 [rad]

%% Calculating Forward Kinematics
xtraj = L1*cos(traj1) + L2*cos(traj1 + traj2); %trajectory on x-axis [m]
ytraj = L1*sin(traj1) + L2*sin(traj1 + traj2); %trajectory on y-axis [m]
Xee = L1*cos(ang1) + L2*cos(ang1 + ang2); %reference of e.e. on x-axis [m]
Yee = L1*sin(ang1) + L2*sin(ang1 + ang2); %reference of e.e. on y-axis [m]

%% Plotting Figures
figure(1)
subplot(2,1,1)
plot(t1,ang1,t2,traj1)
title('Motor 1 Tracking')
xlabel('Time [s]')
ylabel('Angle [rad]')
legend('Joint 1 Tracking','Reference Signal','location','best')
subplot(2,1,2)
plot(t2,ang2,t2,traj2)
title('Motor 2 Tracking')
xlabel('Time [s]')
ylabel('Angle [rad]')
legend('Joint 2 Tracking','Reference Signal','location','best')
```

```
figure(2)
subplot(2,1,1)
plot(t1,traj1,t1,cvel1)
title('Joint 1 - Casual Velocity Filter')
xlabel('Time [s]')
ylabel('Position [rad] & Velocity [rad/s]')
legend('\theta','\theta^\prime','location','best')
subplot(2,1,2)
plot(t2,traj2,t2,cvel2)
title('Joint 2 - Casual Velocity Filter')
xlabel('Time [s]')
ylabel('Position [rad] & Velocity [rad/s]')
legend('\theta','\theta^\prime','location','best')

figure(3)
plot(xtraj,ytraj,Xee,Yee)
xlabel('x [cm]')
ylabel('y [cm]')
title('Tracing Triangle')
legend('Laser Tracking','Reference Signal','location','best')
axis equal
```