Week 3 Update: Wheel Control (Motors & Encoders)

4/15/22

ME/EE/CS 169

Tyler Nguyen and Lorenzo Shaikewitz

3. Encoders

(a) How many counts per motor shaft rotation (rotation of the magnet disk)?

16

(b) How many counts per wheel shaft (motor output)?

720 on both sides

(c) What is the gear reduction in the motor (hint: it is NOT 48 as labeled)?

GR = 720/16 = 45

6. Encoder Processing

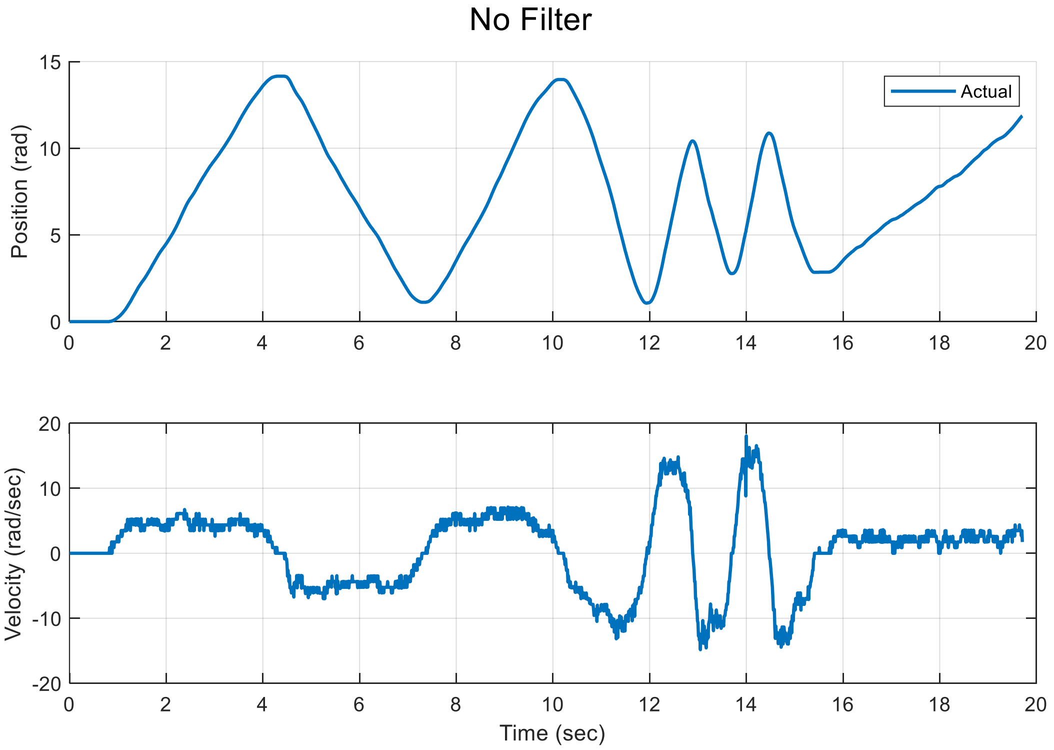
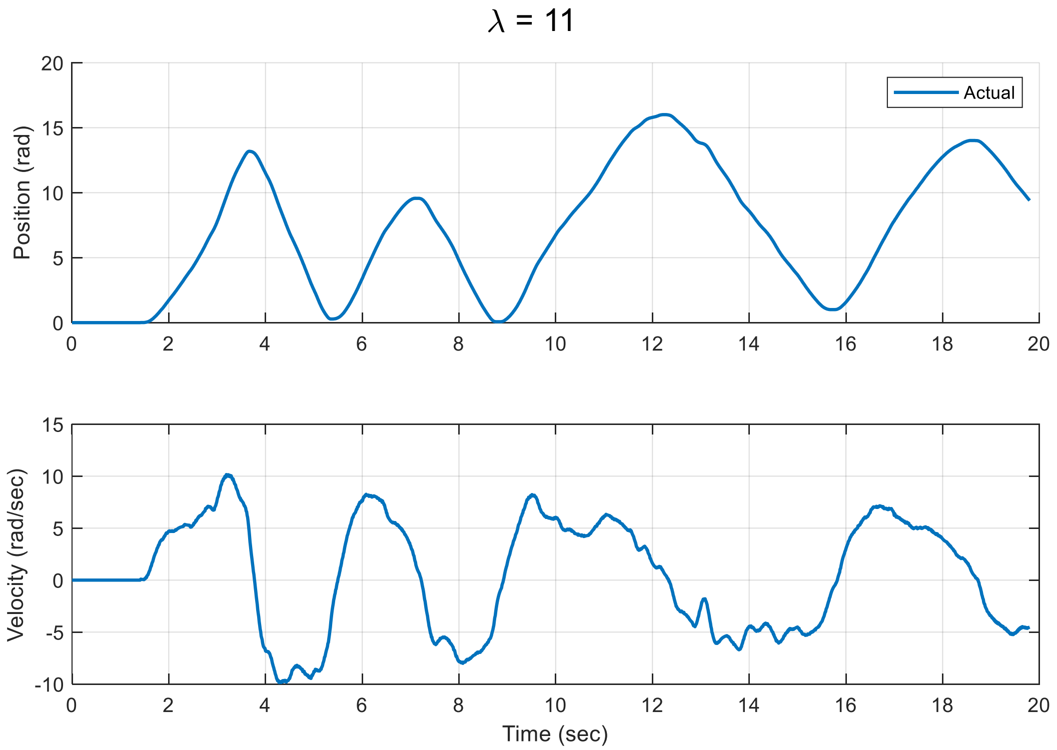
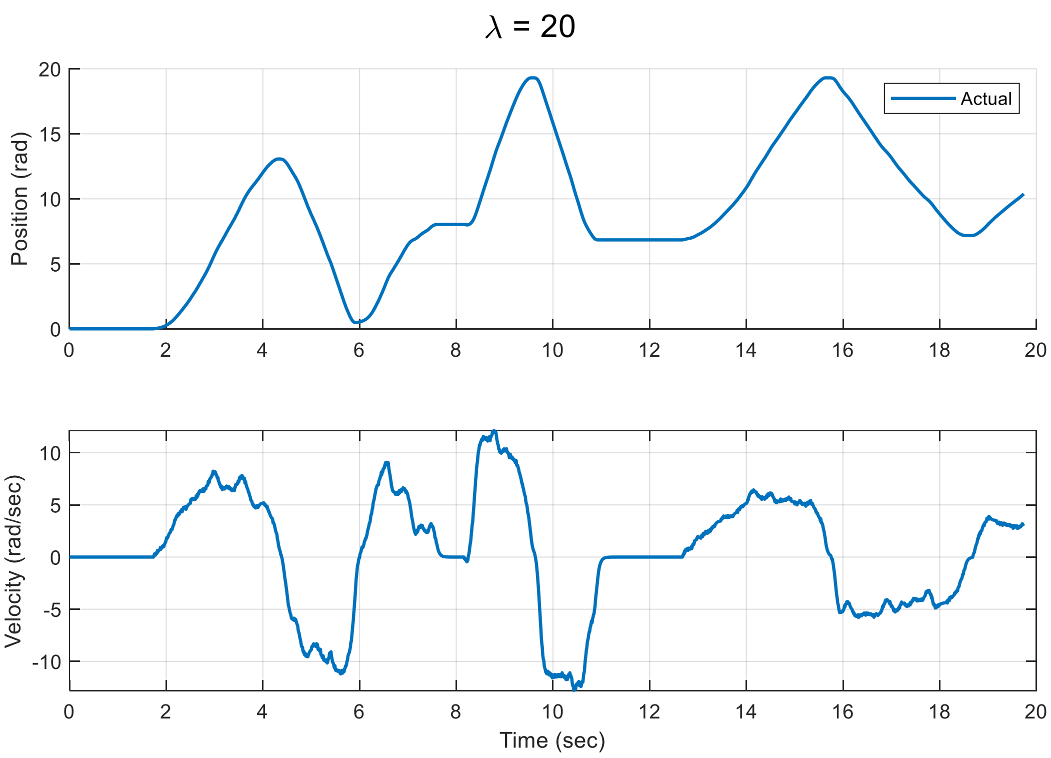


Figure 1. Left Wheel Position and Velocity Data Without Filter, Back and Forth Movement

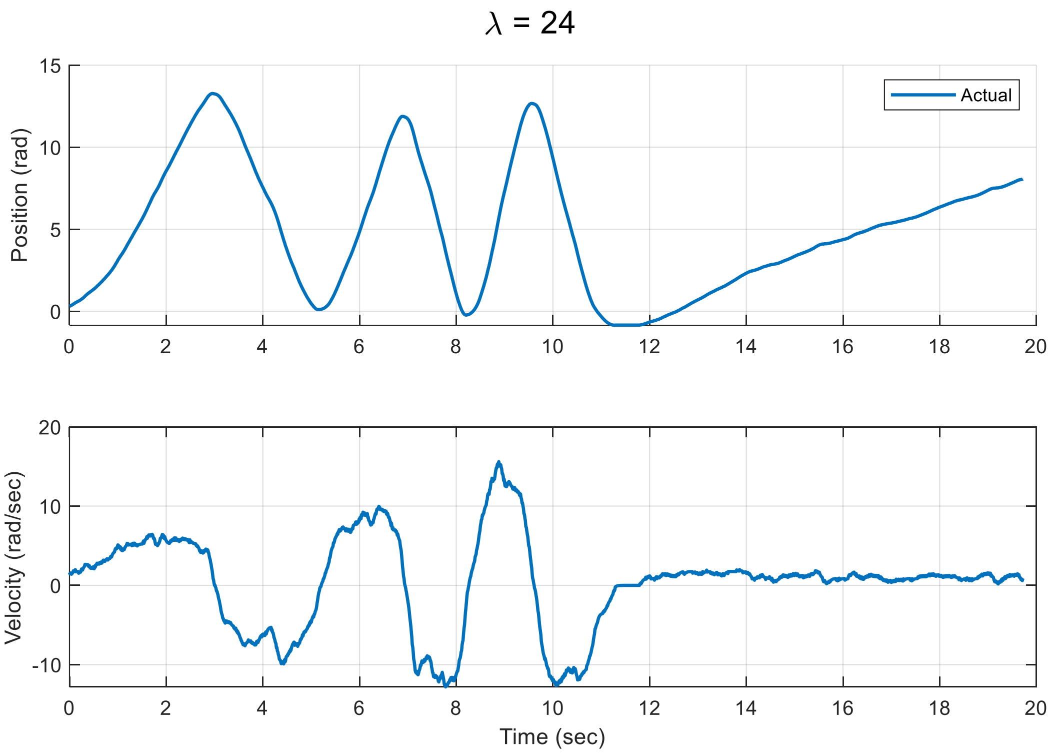
Moving the wheel by hand before filtering, we notice that the unit magnitudes appear correct, however there is a large amount of noise in the velocity graph (Fig. 1).

To avoid the noise in the velocity data, we apply a filter to the velocity measurements. We test three different time constants that correspond to timescales of three measurements ($\lambda = 33$), five measurements ($\lambda = 20$), and nine measurements ($\lambda = 11$). The results are shown below, with the wheel spun by hand and including slow motions as well as back-and-forth movements.





λ = 20



λ = 24

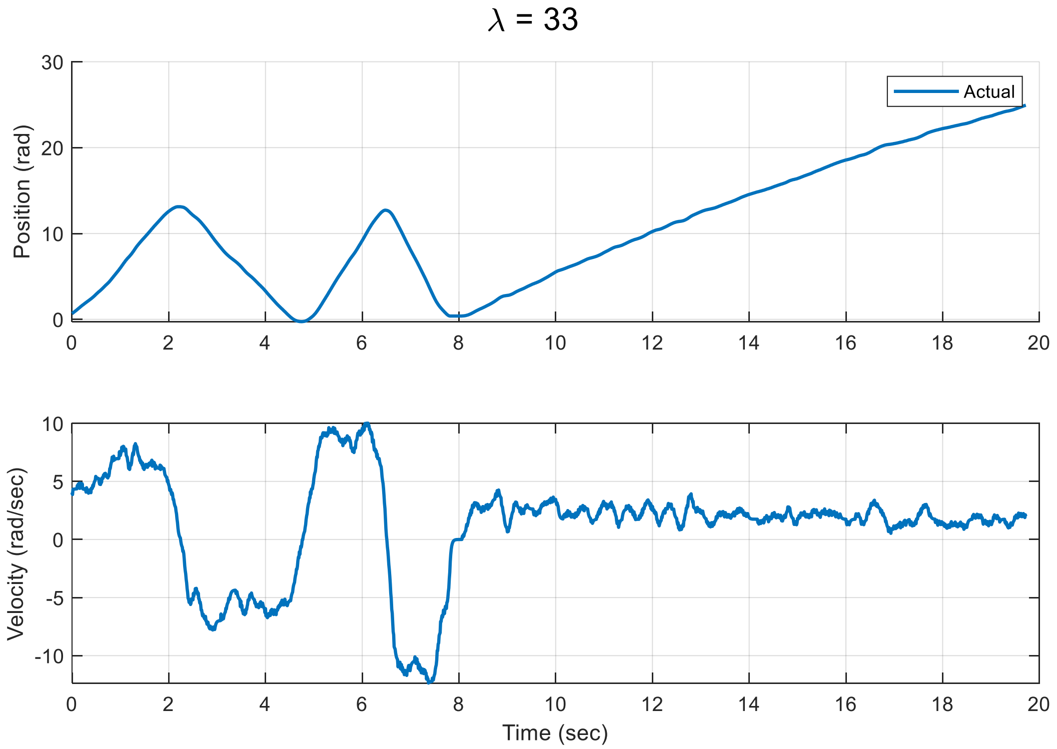


Figure 2. Filtered Velocity Readings of Left Wheel Encoder for Different Time Constants ($\lambda = 11, 20, 24,33$)

Analysis of the filtered velocity readings for different time constants show that $\lambda = 33$ has a velocity plot which is more smooth than the unfiltered, but still includes noise. With this time constant, there is little-to-no delay in the velocity readings with respect to the position data. $\lambda = 11$ data shows very little noise, but the delay in the velocity measurements is nearly 0.1s behind the position data, which is large. $\lambda = 24$ is smoother than $\lambda = 11$ and has minimal delay, but jiter is still apparent with $\pm .25\ rad/s$ spikes. $\lambda = 20$ appears to be a sweet spot between the three, having little noise while maintaining a delay less than 0.01s behind position data. Thus, we will choose $\lambda = 20$ as our time constant value for filtering.

7. PWM to Motor Speed Curve

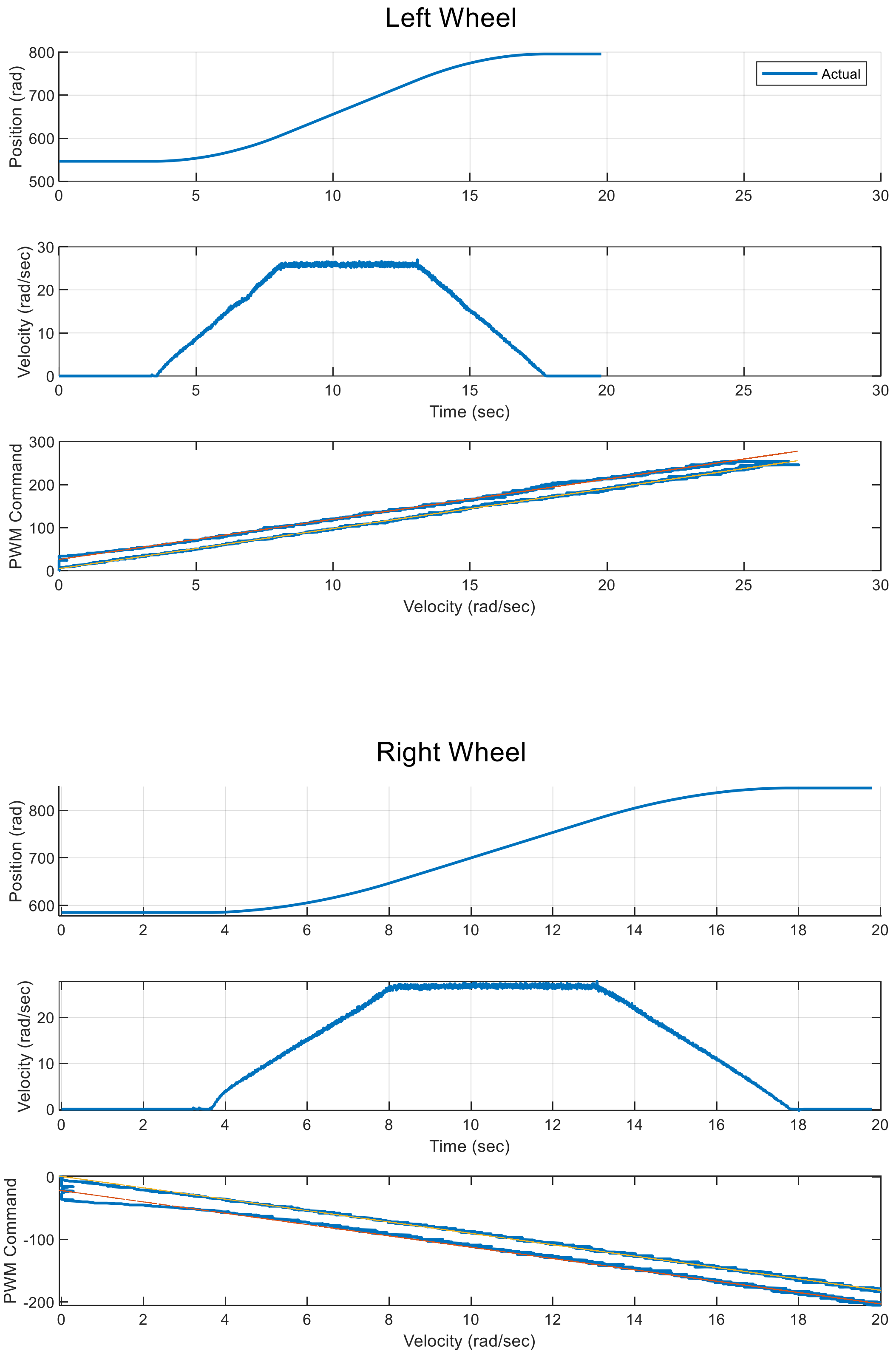


Figure 3. PWM Commands vs. Wheel Speed Curves with Plotted Linear Regression

Plotting the PWM versus wheel speed for a speed up, constant speed, and speed down, we are able to thus correlate the velocity of the wheel (Given by the encoder with filtering) and the PWM commands. For the left wheel, we found the average slope to be: 9.03114 $\frac{PWM}{rad/s}$ and the right wheel was $-9.0640\ \frac{PWM}{rad/s}$. We note that at the initial startup we see that static friction causes a non-linear relationship.

Testing the robot on the ground, we find that friction causes the robot to only start moving at PWM value of ± 55 for both wheels.

8. PWM to Motor Speed Curve

Implementing the values of the PWM to velocity found previously, we thus can perform velocity control of the motors using \wheel_command. The following was performed freespinning.

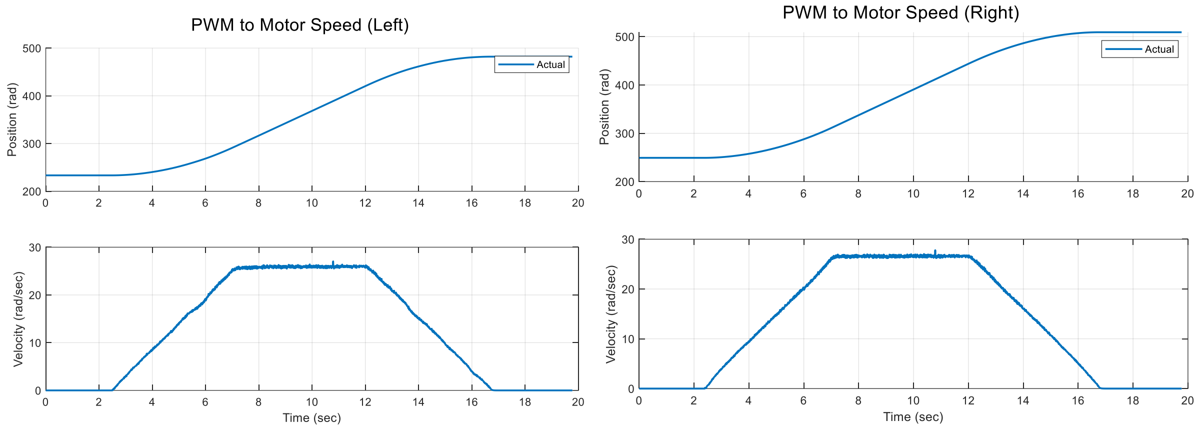
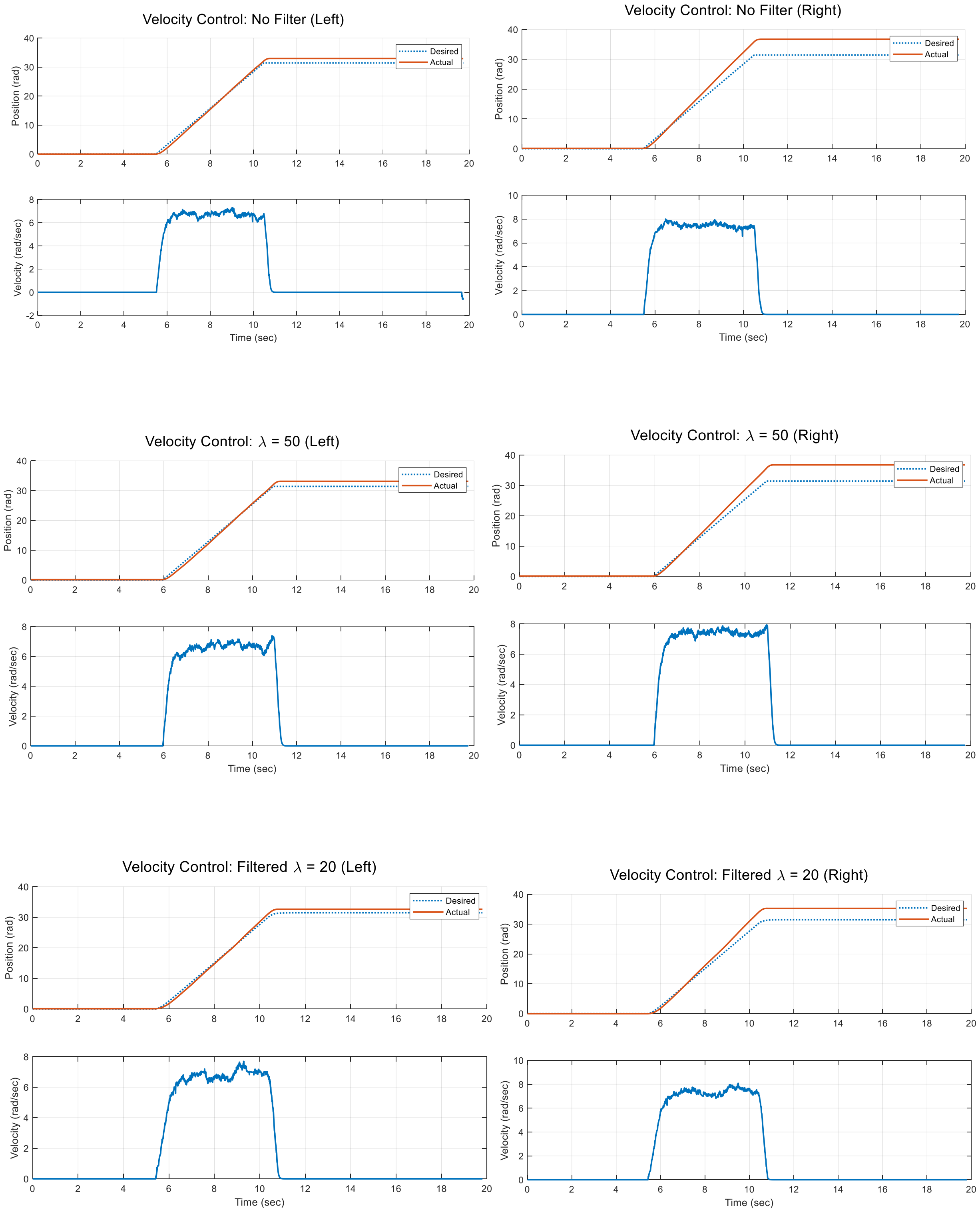


Figure 4. PWM Control of Motor Speed

9. Filtered Motor Desired Velocity

Testing the robot on the floor, we implement a filter on the motor velocity commanding. We notice that with lower values of λ, the “ramp up” and “ramp down” time increases. $\lambda = 20$ appears to be a good value for our usage, however we could even go lower if we are experiencing slipping or too much jolting.



Velocity Control: No Filter (Left)

Velocity Control: No Filter (Right)

Velocity Control: λ = 50 (Left)

Velocity Control: λ = 50 (Right)

Velocity Control: Filtered λ = 20 (Left)

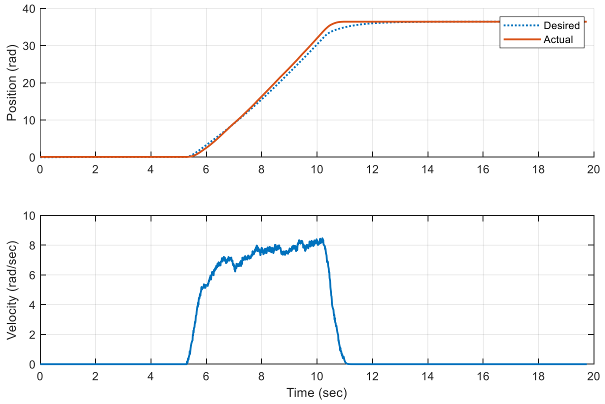
Velocity Control: Filtered λ = 20 (Right)

Figure 5. Filtered Velocity Control

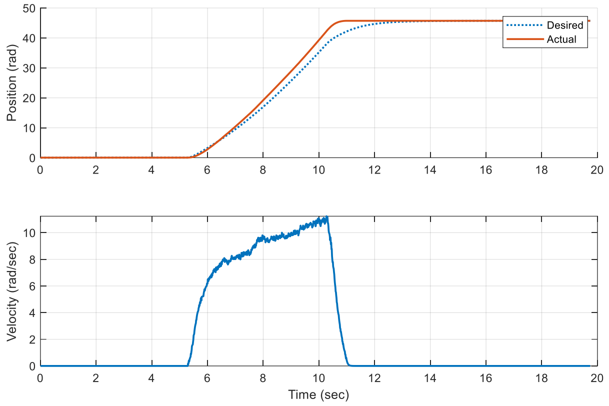
10. Corrective Velocity

Implementing a corrective velocity based upon position greatly helps the actual position to correlate with the desired position. We found that $\lambda = 30$ best helps correcting for errors in the position.

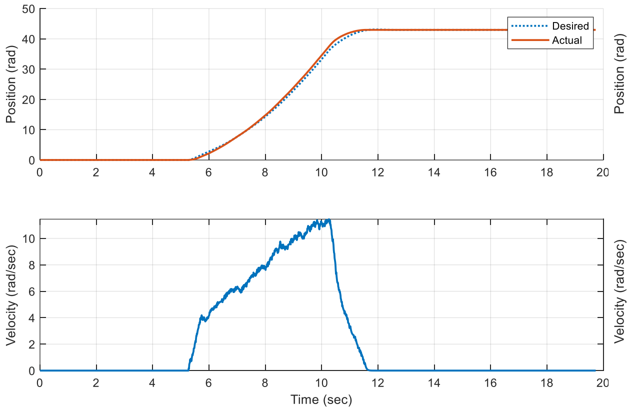
Position Correction: λ = 5 (Left)



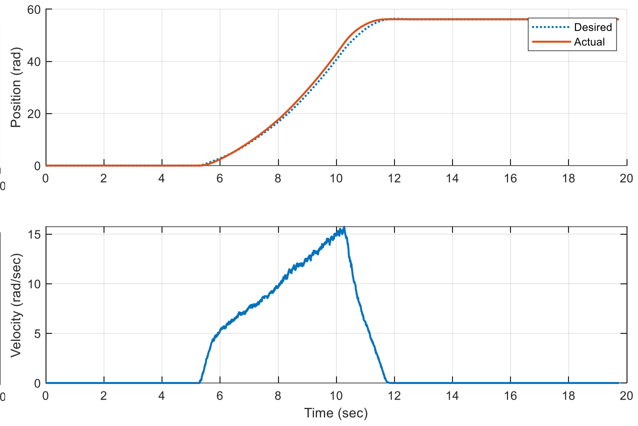
Position Correction: λ = 5 (Right)



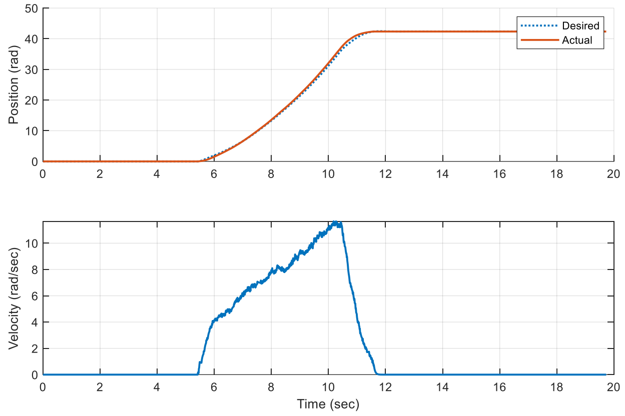
Position Correction: λ = 20 (Left)



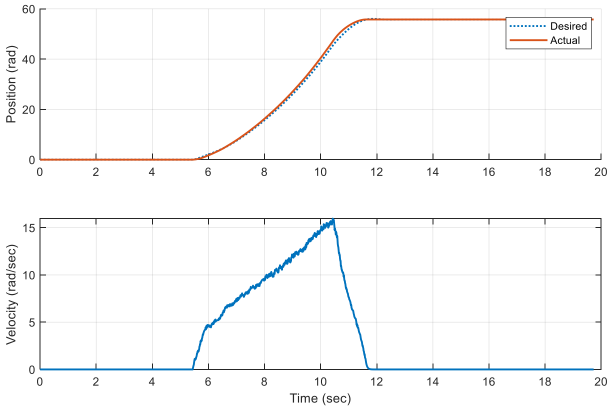
Position Correction: λ = 20 (Right)



Position Correction: λ = 30 (Left)



Position Correction: λ = 30 (Right)


Position (rad)
Desired
Actual
Velocity (rad/sec)
Time (sec)