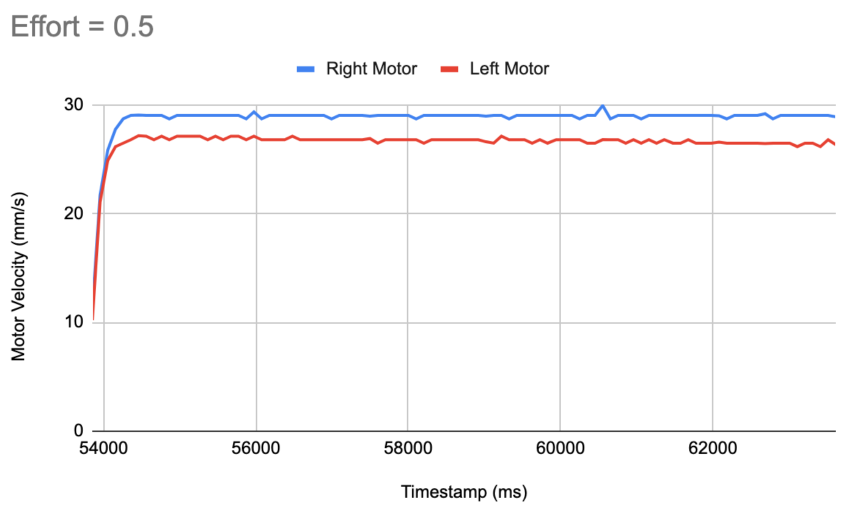
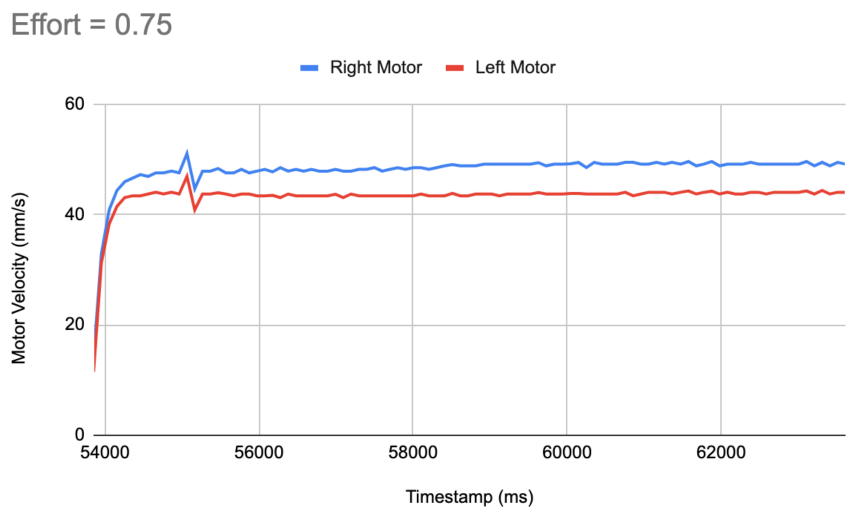
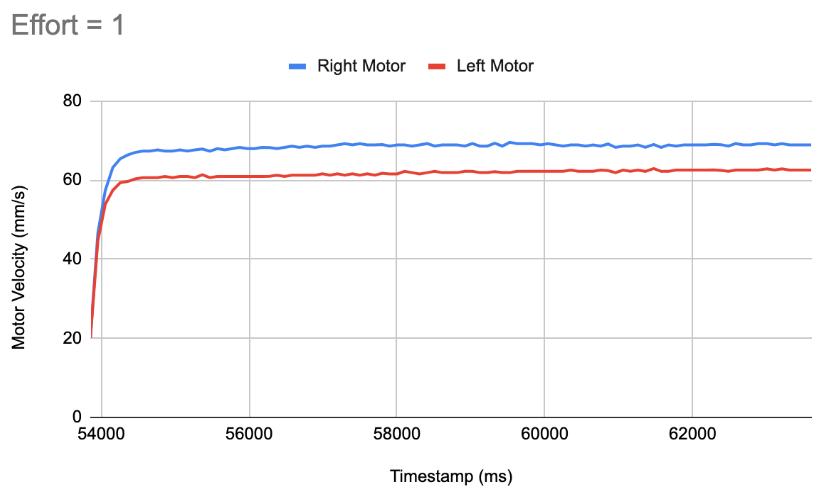
Kai Martell
February & March, 2025
ME134 - Prof. Nemitz
Lab 1 Write up

1. (With robot on the mug) Create plots of speed over time for each wheel (left and right) by applying an effort of 0.5, 0.75, and 1 for 10 seconds. Calculate the mean and standard deviation of the speeds of the two motors for a given effort.

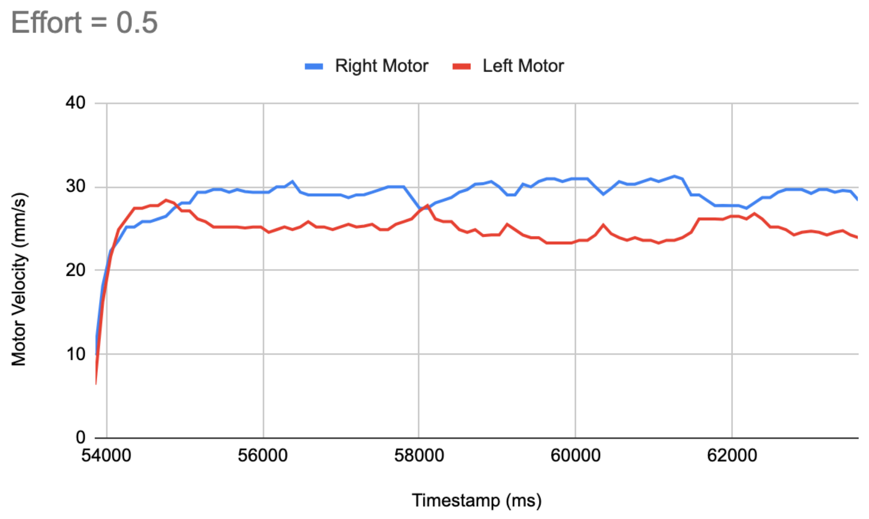


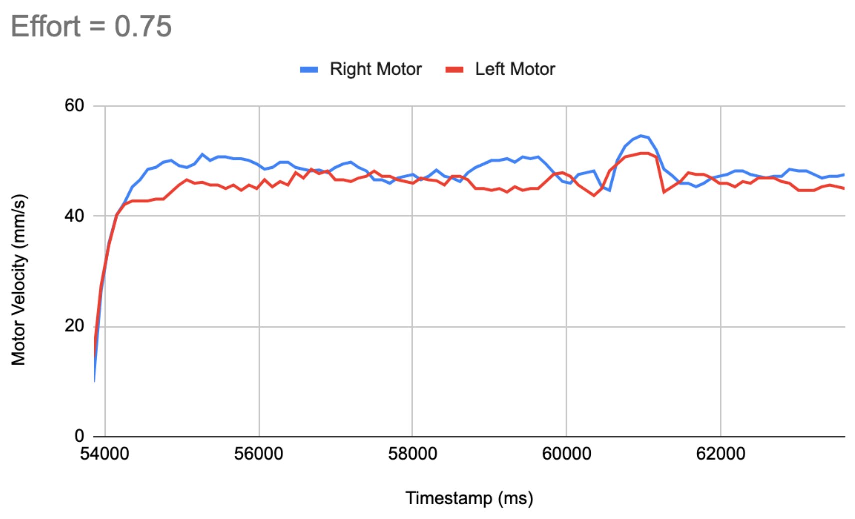


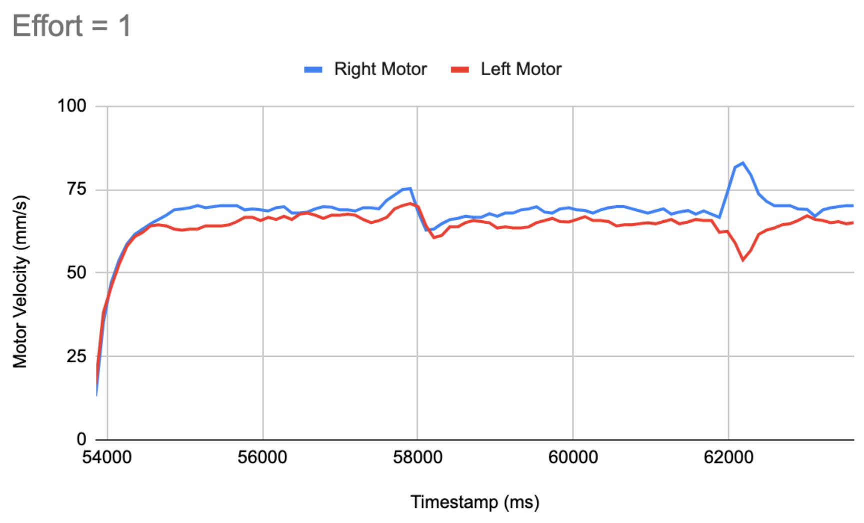


Effort:	0.5	0.5		0.75	0.75		1	1
Motor	Right	Left		Right	Left		Right	Left
Average	28.99680053	-26.45859053		49.17860316	-43.96476105		68.95295947	-62.54595474
Standard deviation	0.1112345289	0.1351707101		0.2354114379	0.2030518284		0.1819726304	0.130967552

2. (With robot on the ground) Create plots of speed over time for each wheel (left and right) by applying an effort of 0.5, 0.75, and 1 for 10 seconds. Calculate the mean and standard deviation of the speeds of the two motors for a given effort





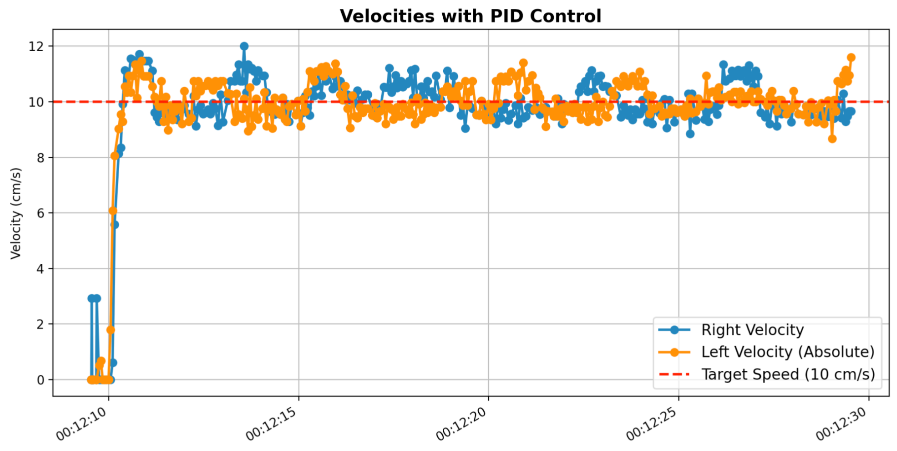


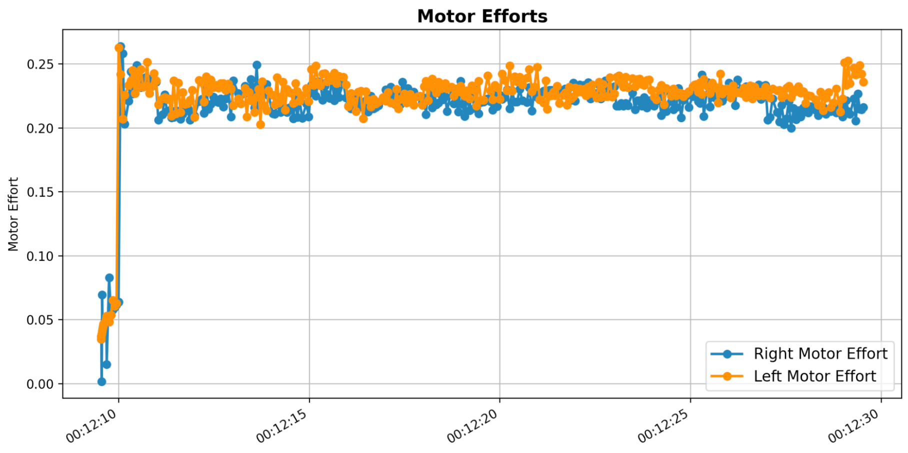
Effort:	0.5	0.5		0.75	0.75		1	1
Motor	Right	Left		Right	Left		Right	Left
Average	28.82342474	-25.23422105		47.43284632	-45.88968947		71.68687789	-63.27570789
Standard deviation	0.8313969518	0.9282811532		0.6094701891	0.8657062334		4.710884959	3.415970009

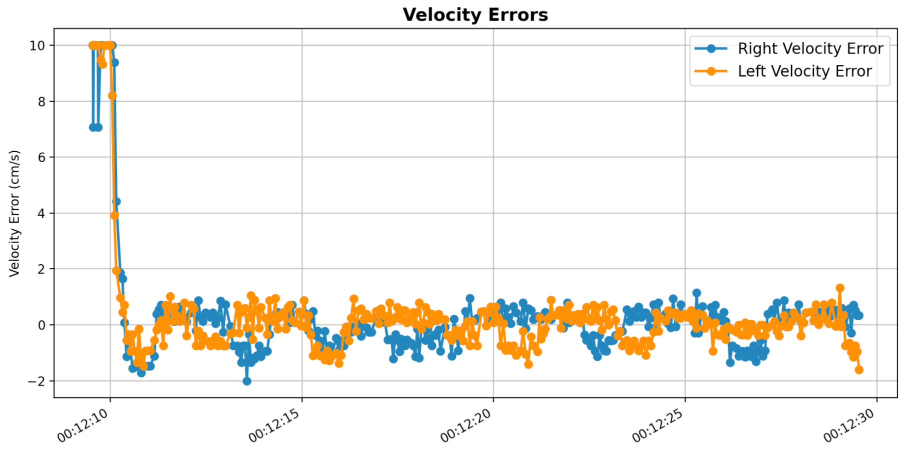
3. After tuning the PID controllers, report your Kp, Ki, and Kd values for each motor. You are not required to use proportional, integrative, and derivative terms. You likely end up with a PI or PD controller. Explain your process for finding these values.
   To tune the PID controller, I went by trial and error, gradually increasing and decreasing each value by marginal values until I thought it to be fit. I started with the I term, with both the P and D terms set to zero, since I noticed the biggest difference in oscillatory behavior when manipulating the integral term. Once oscillations appeared to reach a steady state, I tried adding in a P term to damp the oscillations. The D term was a minimal implementation since adding a massive derivative term would significantly throw off the entire effort correction. In the end, I ended up with values of:
   K_p = 0.0025
   K_i = 0.02
   K_d = 0.0001

4. (With robot on the mug) Set the target speeds to 100mm/s, 200mm/s, 300mm/s, and 400mm/s, create plots of (1) speed over time, (2) effort over time, and (3)speed error over time, for each wheel with your tuned speed controller for 10 seconds.

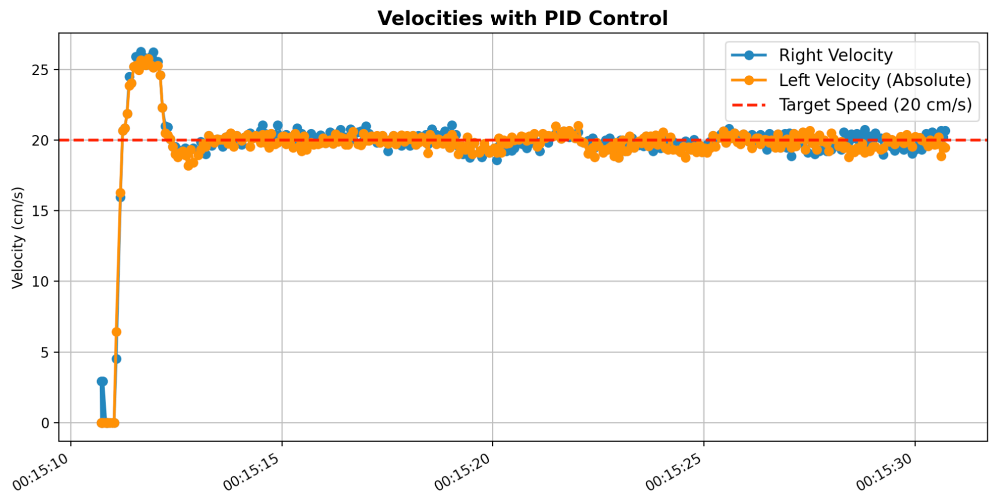
10 cm/s:

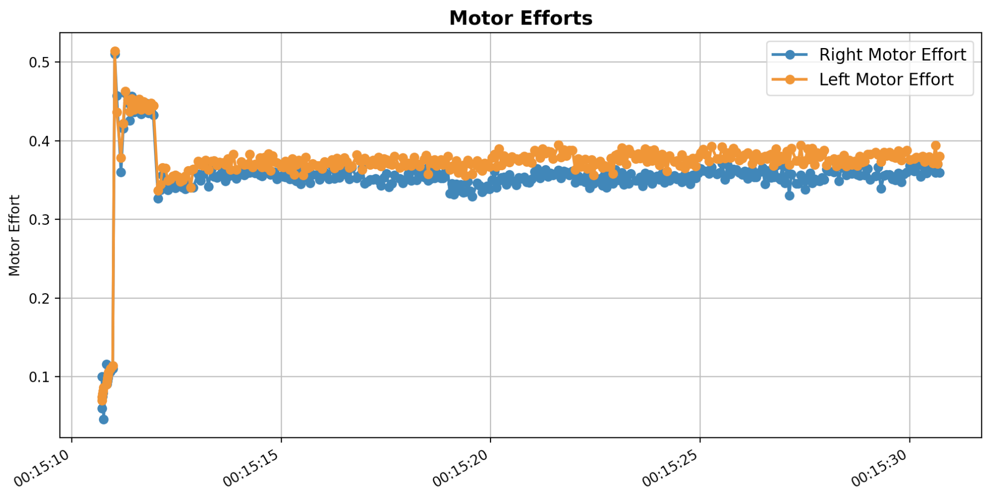

Velocities with PID Control
Velocity (cm/s)
Right Velocity
Left Velocity (Absolute)
Target Speed (10 cm/s)
00:12:10
00:12:15
00:12:20
00:12:25
00:12:30

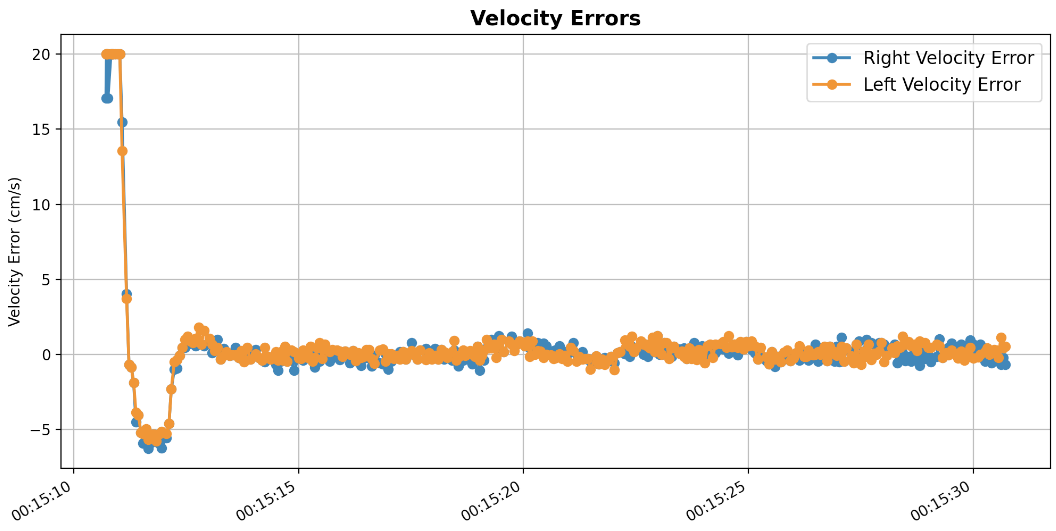

Motor Efforts
Motor Effort
Right Motor Effort
Left Motor Effort


Velocity Errors
Velocity Error (cm/s)
Right Velocity Error
Left Velocity Error

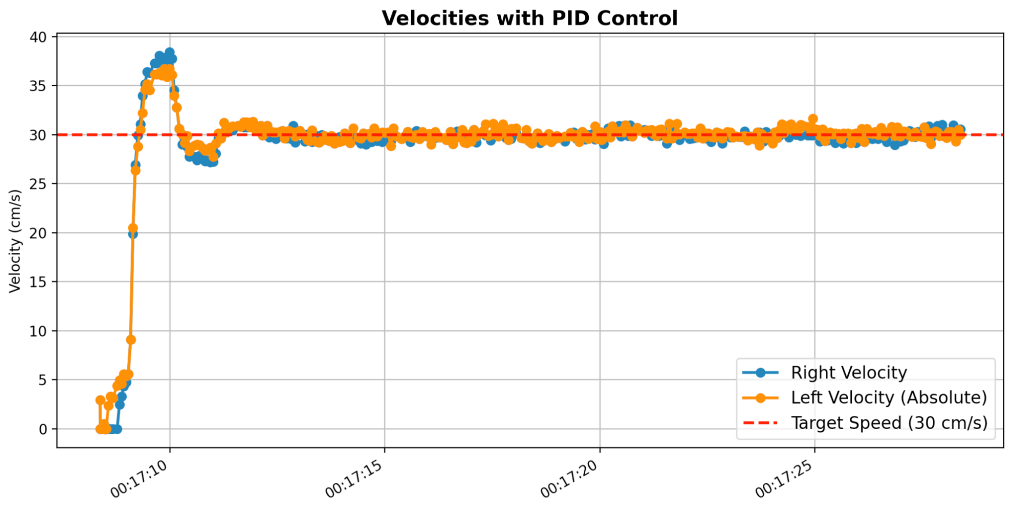
20 cm/s:

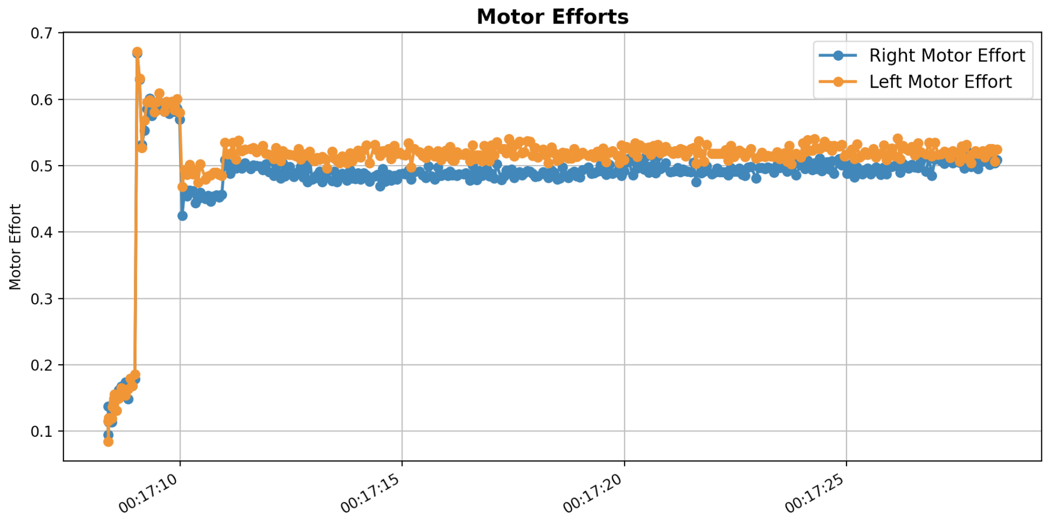

Velocities with PID Control
Right Velocity
Left Velocity (Absolute)
Target Speed (20 cm/s)
Velocity (cm/s)
0
5
10
15
20
25
00:15:10
00:15:15
00:15:20
00:15:25
00:15:30

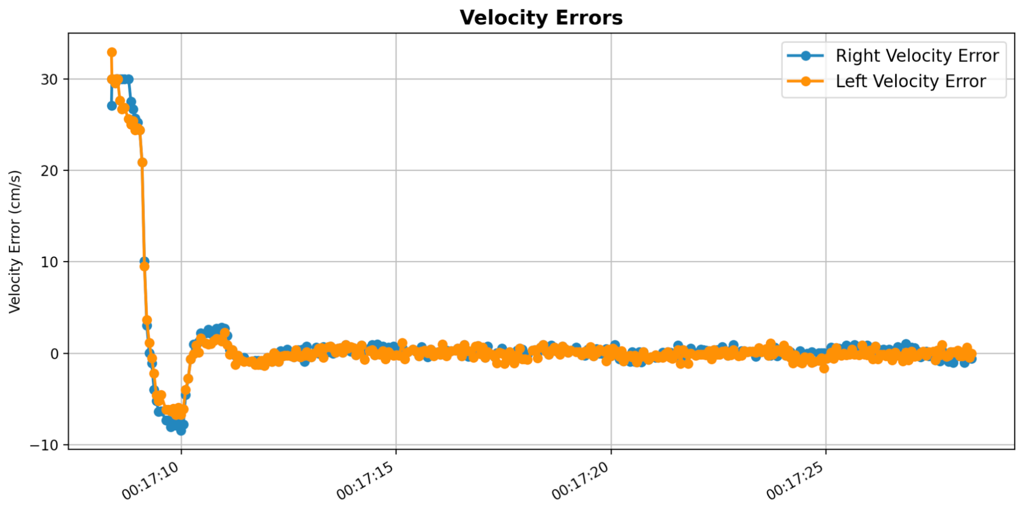

Motor Efforts
Right Motor Effort
Left Motor Effort
Motor Effort
0.1
0.2
0.3
0.4
0.5
00:15:10
00:15:15
00:15:20
00:15:25
00:15:30


Velocity Errors
Right Velocity Error
Left Velocity Error
Velocity Error (cm/s)
−5
0
5
10
15
20
00:15:10
00:15:15
00:15:20
00:15:25
00:15:30

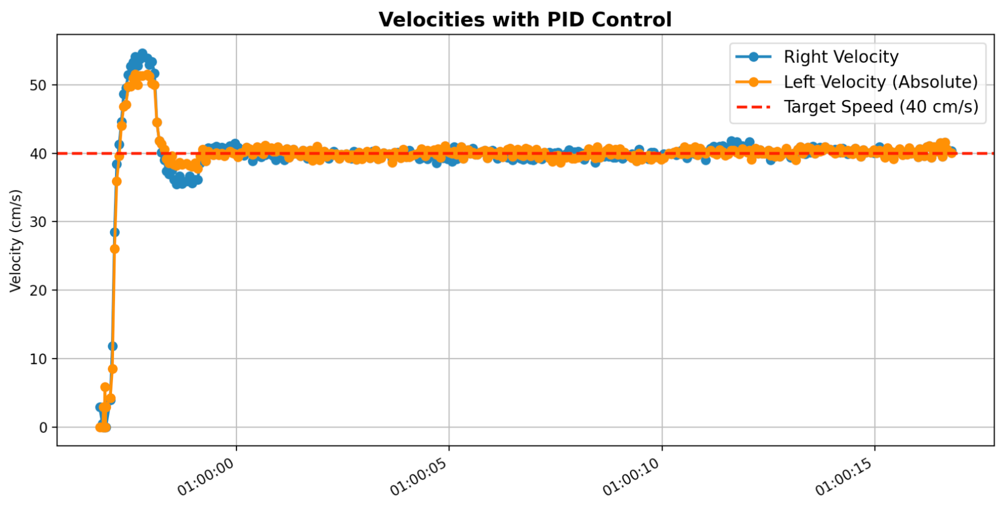
30 cm/s:

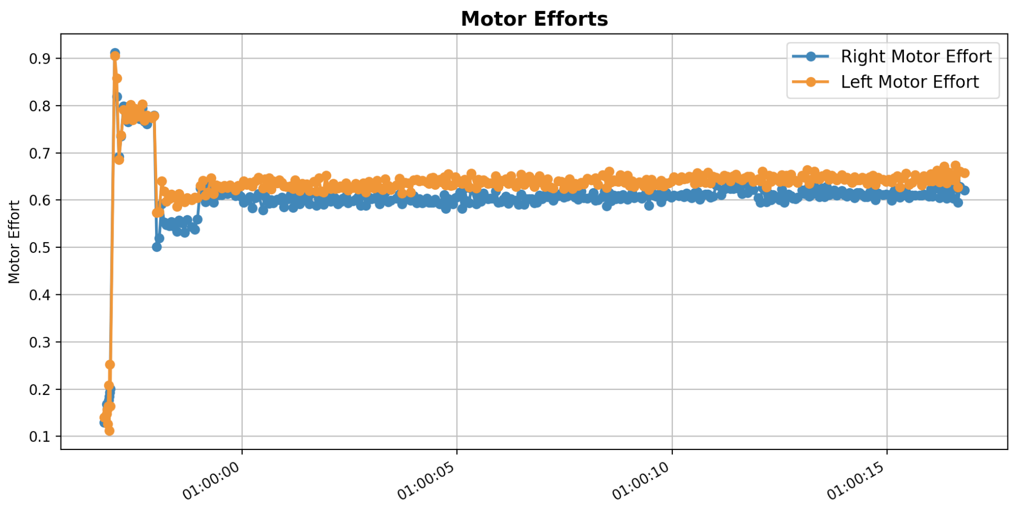

Velocities with PID Control
Velocity (cm/s)
40
35
30
25
20
15
10
5
0
00:17:10
00:17:15
00:17:20
00:17:25
Right Velocity
Left Velocity (Absolute)
Target Speed (30 cm/s)

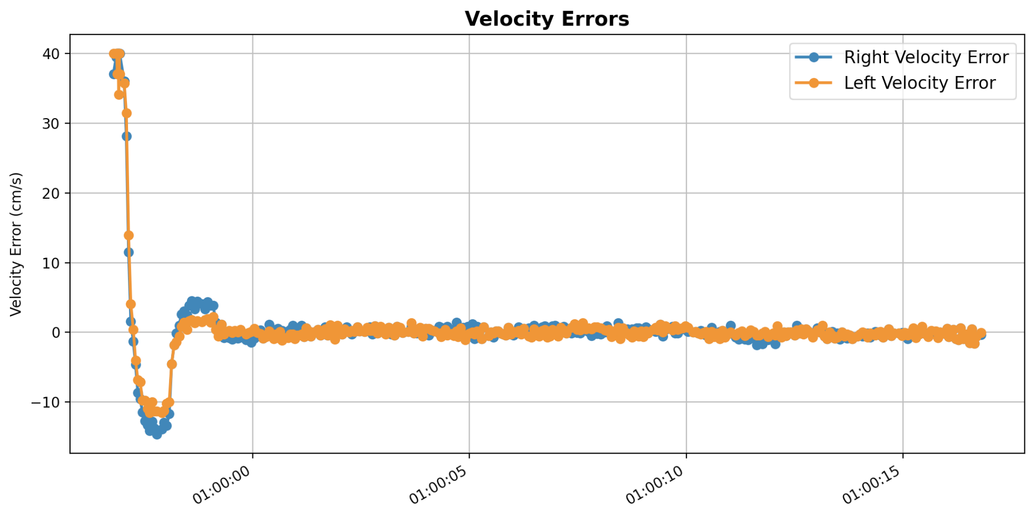

Motor Efforts
Motor Effort
0.7
0.6
0.5
0.4
0.3
0.2
0.1
00:17:10
00:17:15
00:17:20
00:17:25
Right Motor Effort
Left Motor Effort


Velocity Errors
Velocity Error (cm/s)
30
20
10
0
−10
00:17:10
00:17:15
00:17:20
00:17:25
Right Velocity Error
Left Velocity Error

40 cm/s:


Velocities with PID Control
Right Velocity
Left Velocity (Absolute)
Target Speed (40 cm/s)
Velocity (cm/s)
0
10
20
30
40
50
01:00:00
01:00:05
01:00:10
01:00:15


Motor Efforts
Right Motor Effort
Left Motor Effort
Motor Effort
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
01:00:00
01:00:05
01:00:10
01:00:15


Velocity Errors
Right Velocity Error
Left Velocity Error
Velocity Error (cm/s)
−10
0
10
20
30
40
01:00:00
01:00:05
01:00:10
01:00:15

5. (With robot on the ground) Set the target speeds of both motors to 100mm/s, 200mm/s, 300mm/s, and 400mm/s, and verify how far the robot moved after 10 seconds of locomotion. Use a measuring tape.

<u>10 cm/s:</u>
Dx = 96 cm
Dy = 8 cm

<u>20 cm/s:</u>
Dx = 188 cm
Dy = 42cm

<u>30cm/s:</u>
Dx = 300 cm
Dy = -28cm

<u>40 cm/s:</u>
Dx = 384 cm
Dy = 12 cm

6. Discuss the performance of your controller and how it can be further improved.

While my controller seemed to be a significant improvement from just inputting an effort into the robot's motors, the robot still failed to move in a completely straight line, and would veer off to one side as it moved along. This could be improved by adding personalized PID values to each motor instead of one set for both motors - I tried this however after fiddling around with the values for quite some time I decided the same values were close enough to demonstrate improvement from no control at all. Another limitation is the physical hardware; the motors we are using aren't the most expensive and robust and the encoders inside them definitely aren't the most accurate - if we were using very high quality motors with nice encoders the results might be more accurate.