MTE544 Lab 2

Group 29

Friday 3:00 PM

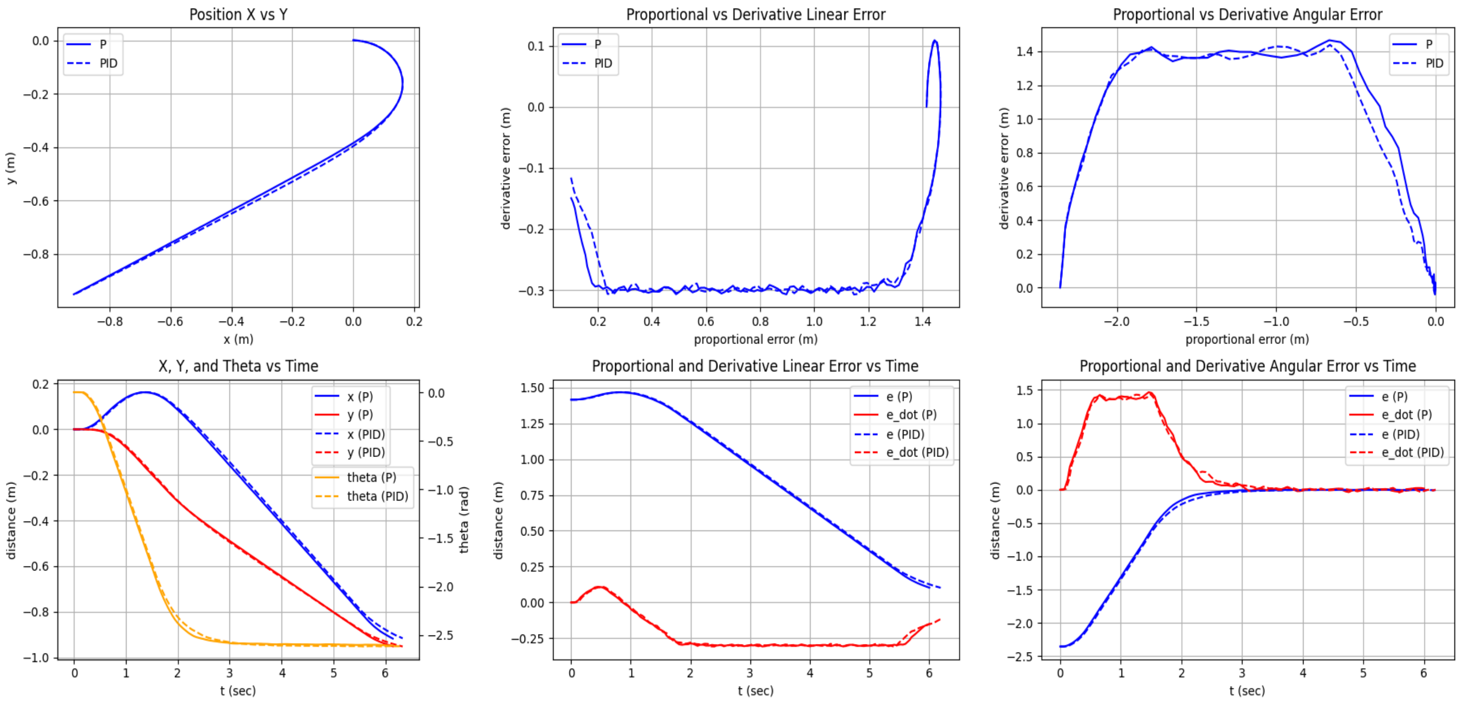
Kevin Xue (20814292)

Hunter McCullagh (20899434)

Justin Zhu (20879375)

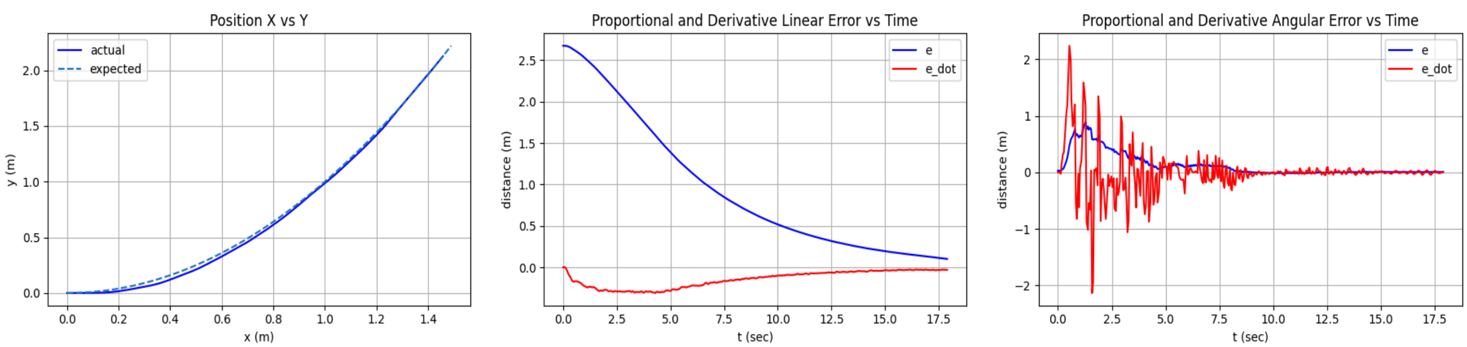
Station Number: 162

Robot Number: 12



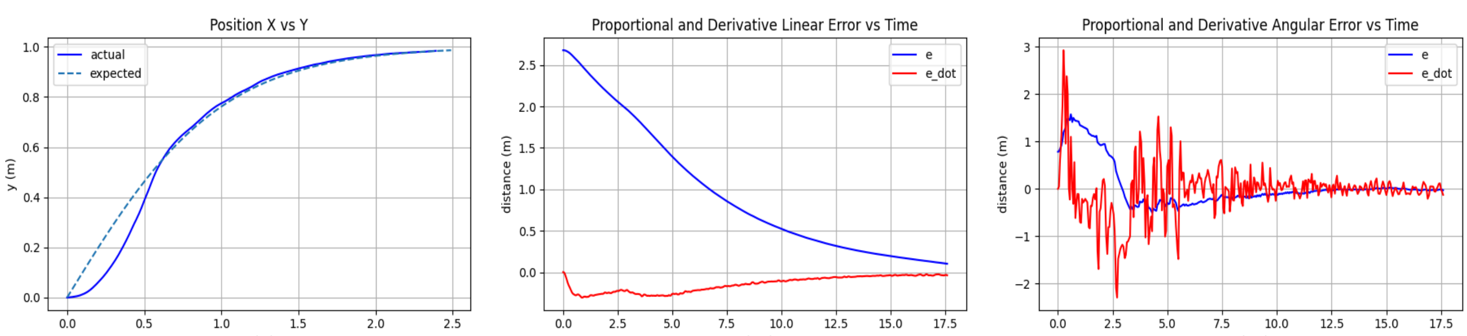
(a) (b) (c)

Figure 1: Comparison of P and PID controller for point trajectory



(b) (b) (c)

Figure 2: PID controller for parabola trajectory



(c) (b) (c)

Figure 3: PID controller for sigmoid trajectory

Section 1

Single Waypoint Trajectory

The odometry data and the P/PID error data were logged throughout the robot’s operation and are plotted in Figure 1, with every plot comparing both data collected in the P and PID controller. More specifically, the odometry data can be examined in Figure 1a), where the trajectory on a cartesian plane from (0,0) starting point to (-1,-1) end destination is illustrated in the top right plot while the bottom right overlays the state variables (x, y, theta) against time. The following two middle plots show the proportional error and the derivative error compared against each other and against time for the linear velocity controller, whereas the two leftmost plots for the angular velocity controller.

After tuning both the P and PID controller, we can see based on the selection of plots in Figure 1 that there is very minimal difference in the performance between the two controllers. It is noteworthy that in both cases, the robot makes a turn at the start with a non-zero angular velocity (and decreasing theta) until the 3-second mark where the angular velocity approaches zero with a constant theta and the remaining trajectory is a straight line with the proportional gain having a more significant influence than the derivative gain

A PID controller was selected for Part 6 because this type of controller will reduce steady-state error and oscillations. By tuning the integral gain, the steady-state error can be reduced when compared to the steady-state error of a P controller. Also, a well-tuned derivative gain will ensure less overshoot & oscillations, which also makes it a more appealing controller choice compared to a P controller.

Parabola Trajectory

Figure 2a) displays the robot trajectory with the expected parabola path overlaid. It is evident that the PID controller was tuned well, as we can see that the robot's path closely overlaps with the expected path, though in a similar fashion to the single waypoint trajectory, the linear proportional gain had a much more pronounced effect than the linear derivative gain. However, in retrospect, a higher angular derivative gain can be used to dampen the oscillations seen in Figure 2c).

Sigmoid Trajectory

Likewise to the parabola, Figure 3a) displays the robot's trajectory along with the expected sigmoid path. Similarly, the linear PID gains are well tuned as seen in Figure 3b), though a higher angular proportional and derivative gain can be used to dampen the oscillations and spikes, particularly in the first 5 seconds in Figure 3c).

Section 2

The gains for the tuned P & PID Controller can be seen in the below table.

	Proportional Gain	Integral Gain	Derivative Gain
P Linear Controller	1.2	N/A	N/A
P Angular Controller	2.2	N/A	N/A
PID Linear Controller	1.2	0.2	0.2
PID Angular Controller	1.8	0.8	0.2

The overshoot is determined by when the position of the robot surpasses the desired point. However, through the code implementation when a certain linear error threshold was met the program would terminate. Thus the P & PID controller have an overshoot of zero, this is also confirmed in Figure 1a) as there is no evident overshoot in the X & Y direction. To determine accuracy of the controller, it is a percentage ratio between the destination point & the final point of the robot. Knowing that the robot stops movement upon reaching a certain linear error as stated above, both controllers should have a very similar accuracy. To determine the agility of a controller, measuring the rise time from 10% to 90% of the steady state value is used and to determine rise time from the data, the time interval from 90% of initial error to 10% of initial error was used.

	P Controller	**PID Controller**
Accuracy (%)	92.71	92.75
Overshoot (%)	0	0
Linear Agility (seconds)	3.801	3.889
Angular Agility (seconds)	1.460	1.559

Based on the agility values in the above table, it indicates that the P & PID controller have an adequate proportional gain, but the derivative gain in the PID controller increases the rise time for the linear & angular controller. The accuracy of the controllers are similar due to how the code was implemented. The overshoot of both controllers is 0% which implies that both controllers were properly tuned.

From this analysis, we can see that the P controller was more effective overall because of its more effective agility.