Name: Sim Justin

Matric No: A0257926N

Project 1: PID Control

# Initial Conditions

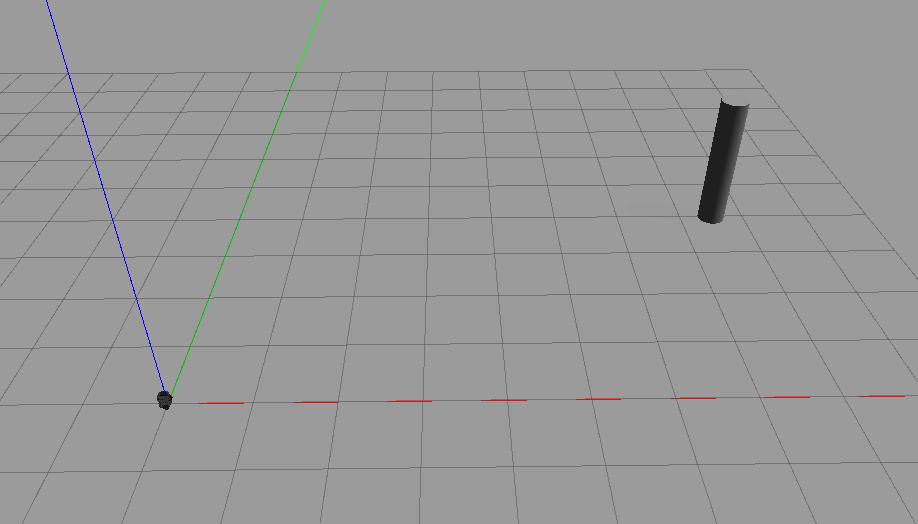


Figure I: Initial Location of Pillar and Turtlebot3

Starting Location of Pillar: (7.5, 4)

Stopping Distance: 1.1

Initial Distance: 8.5

Orientation of Pole: 0.489957326 rad

# Implementation of PID Control

Integral Term: Integral term was defined by principles of Riemann Sum, whereby the error integral is estimated by taking the summation of the product of the error at each point of time and its associated delta time. As opposed to a Classical Riemann Sum, Trapezoidal Riemann Sum was implemented to improve accuracy of integral estimation (Stephanie, n.d.) by taking the area of the trapezoid encompassed by current error, previous error, and their associated timestamps. To avoid the issue of integral windup, the integrator is only active while within a controllable region as defined by the Proportional Term.

Derivative Term: Derivative term was defined through concept of differentiation by first principles, taking the gradient between the current error and the previous error as an estimate of the gradient of the tangent of the curve at the current point.

PID Control Term: PID term was defined by taking the summation of the product between each of the proportional, integral and derivative terms and their associated constants.

Implementation can be seen as follows:

A computer screen shot of a program

Description automatically generated

Figure II: Implementation of PID Control Terms

Angular error regulation is done by normalising *error\_angle­* to the range of -PI to PI, by adding 2 PI to it if it is smaller than -PI or adding the same value if it is larger than PI, until it falls within the prescribed range. Implementation can be seen below.

A black background with white text

Description automatically generated

Figure III: Angular Error Regulation

Angular control signal is limited by imposing a limit to both the angular velocities such that if the control signal output by the PID controller exceeds this limit, the angular velocity is written to be the maximum permissible angular velocity instead.

Linear control signal limiting was performed similarly. Implementation for both can be found below.

A computer code with text

Description automatically generated

Figure IV: Control Signal Limitation for Angular and Linear Velocities

# Tuning of PID Control

## Tuning Process

## Controller Characterisation

# Performance of PID Control

# Conclusions

1. Initial conditions.
   1. Show the initial location of the pillar and the Turtlebot3 of your setting. Use the view that best show their locations.
   2. Calculate the initial distance and orientation of the pole with respect to the Turtlebot3.
2. Implementation of PID control. Discuss how the PID control is implemented in ROS with reference to your code.
   1. Describe how you (1) define the integral term, (2) define the derivative term and (3) define the PID control term
   2. Describe the purpose of the code to (1) regularize the angular error (error\_angle) and (2) limit the angular control signal (trans\_angle).
3. Tuning of PID.
   1. Discuss the tuning process, e.g., which gain is determined first, which gain is determined second, how it is determined and so on.
   2. Discuss how you would characterise the PID control (P, PI, PD or PID). Discuss the merits, demerits, and other points that you want to highlight about your design.
4. Performance of PID control.
   1. Attach the plots of errors vs time (both linear and angular errors) that represents your best design.
   2. Analyse the performance, e.g., overshoot, steady state error and settling time.
5. Conclusions and key learning points.