Assignment 2 Report

Objective:

- Trajectory generation and follow using a controller.

Requirement:

- sympy library(installing using pip : - pip install sympy)

Steps:

1. Generate trajectory by making function using **"sympy"** library.
2. Use generated points and get linear velocity and angular velocity to move robot one point to another point.
3. Plot Graphs using Point generated in **2nd Steps.**

Experiment 1:

- Generating Points at particular delay.

Trajectory Scan

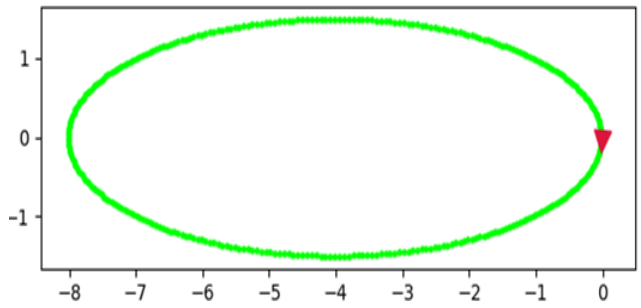

Figure 1.1

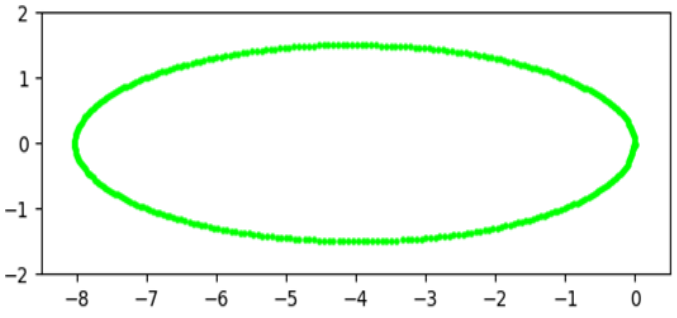

Figure 1.2

Advantages:

- Accurately following generated trajectory.

Disadvantage:

- Takes more time to complete whole trajectory.

Experiment 2:

- Continuously generate points.

Trajectory Scan

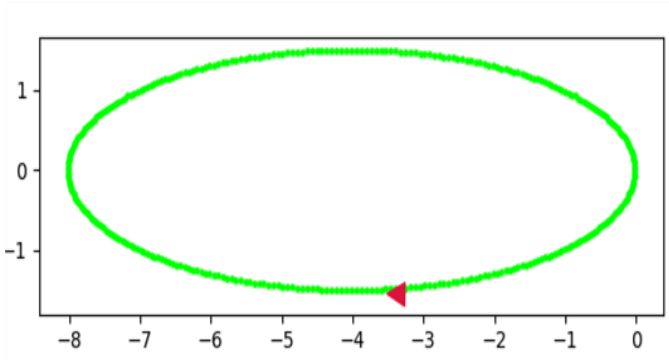

Figure 2.1

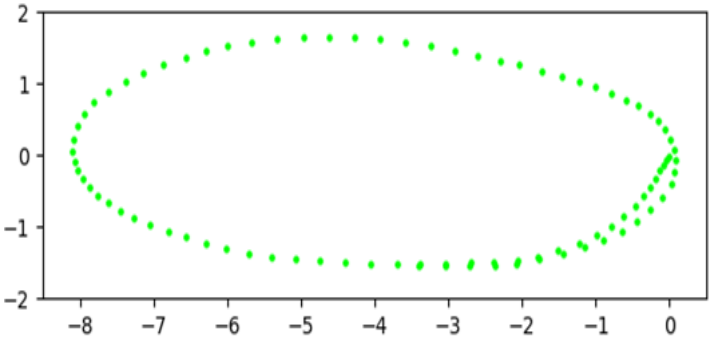

Figure 2.2

Advantages:

- Takes less time to complete the trajectory.

Disadvantage:

- Slightly inaccurate at beginning and move little more in y direction at some curves.

Results:

The results of the project showed that the Turtlebot was able to successfully follow the trajectory path for the shape specified by the equation. The PI controller was able to modulate the speed of the Turtlebot as it followed the trajectory, and the angle of the Turtlebot was adjusted according to the equation.
The plotted trajectory of the robot also closely matched the real shape as per the equation. The position difference between the actual and theoretical path was minimal, indicating that the Turtlebot was able to accurately follow the trajectory path.

Conclusion:

In conclusion, this project successfully demonstrated the ability of a Turtlebot to follow a specific shape, such as an ellipse or circle, based on a generalized equation. The PI controller was able to effectively modulate the speed of the Turtlebot, and the angle of the Turtlebot was adjusted according to the equation. The plotted trajectory of the robot also closely matched the real shape as per the equation, indicating that the Turtlebot was able to accurately follow the trajectory path. This project shows the potential for using ROS1 and Python in the control and trajectory planning of robotic systems.