SC627: Assignment 2

Report by: Atharva Mete | 190010012

Introduction

In this assignment, I implemented a potential field algorithm. The implementation was done in python and simulated on Gazebo & RVIZ using Turtlebot3. The helper function from assignment 1 is used for calculating the nearest point to the polygon.

Potential Field Algorithm:

The potential function approach directs a robot as if it were a particle moving in a gradient vector field. Gradients can be intuitively viewed as forces acting on a positively charged particle robot that is attracted to the negatively charged goal. Obstacles also have a positive charge which forms a repulsive force directing the robot away from obstacles. The combination of repulsive and attractive forces hopefully directs the robot from the start location to the goal location while avoiding obstacles

Attractive potential:

$$U_{att}(q) = \begin{cases} \frac{1}{2}\chi d^2(q, q_{goal}), d(q, q_{goal}) \leq d^{\star}_{goal} \\ d^{\star}_{goal}\chi d(q, q_{goal}) - \frac{1}{2}\chi (d^{\star}_{goal})^2, else \end{cases}$$

$$\nabla U_{\text{att}}(q) = \begin{cases} \chi(q - q_{\text{goal}}), & d(q, q_{\text{goal}}) \leq d^*_{\text{goal}} \\ \frac{d^*_{\text{goal}}\,\chi(q - q_{\text{goal}})}{d(q, q_{\text{goal}})}, & d(q, q_{\text{goal}}) > d^*_{\text{goal}} \end{cases}$$

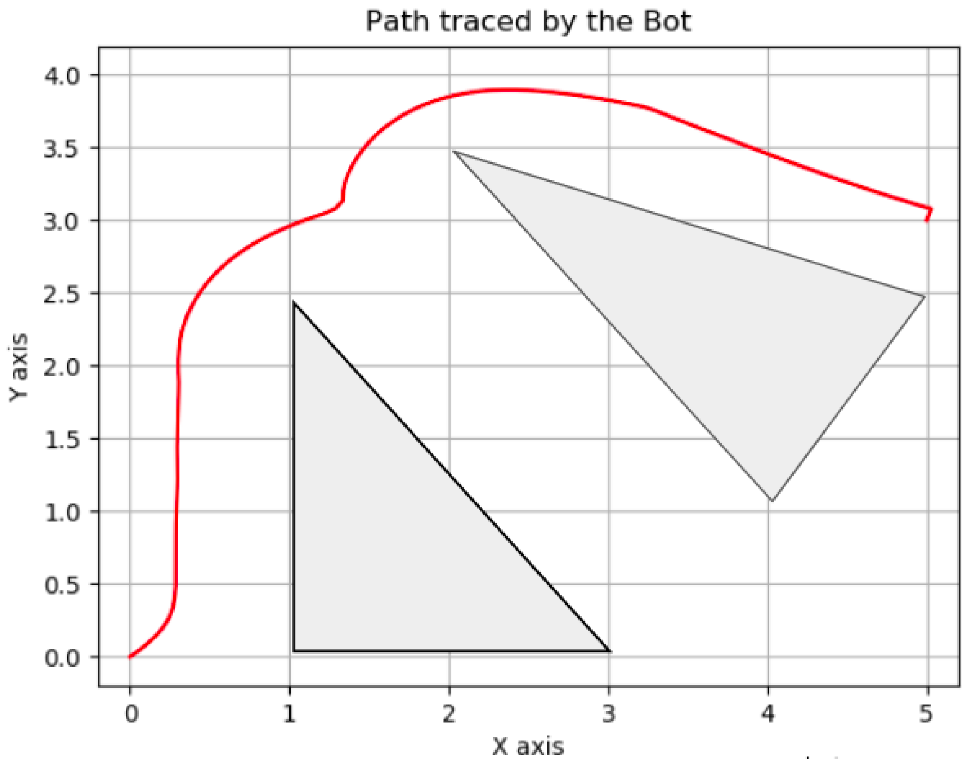
Repulsive Potential:

$$U_{rep_i}(q) = \begin{cases} \frac{1}{2}\eta \left(\frac{1}{d_i(q)} - \frac{1}{Q^{\star}_i}\right)^2, d_i(q) \leq Q^{\star}_i \\ 0, else \end{cases}$$

$$\nabla U_{\text{rep}}(q) = \begin{cases} \eta\left(\frac{1}{Q^*} - \frac{1}{d_i(q)}\right)\frac{1}{d_i{}^2(q)}\nabla d_i(q), & d_i(q) \leq Q^*_i \\ 0, & d_i(q) > Q^*_i \end{cases}$$

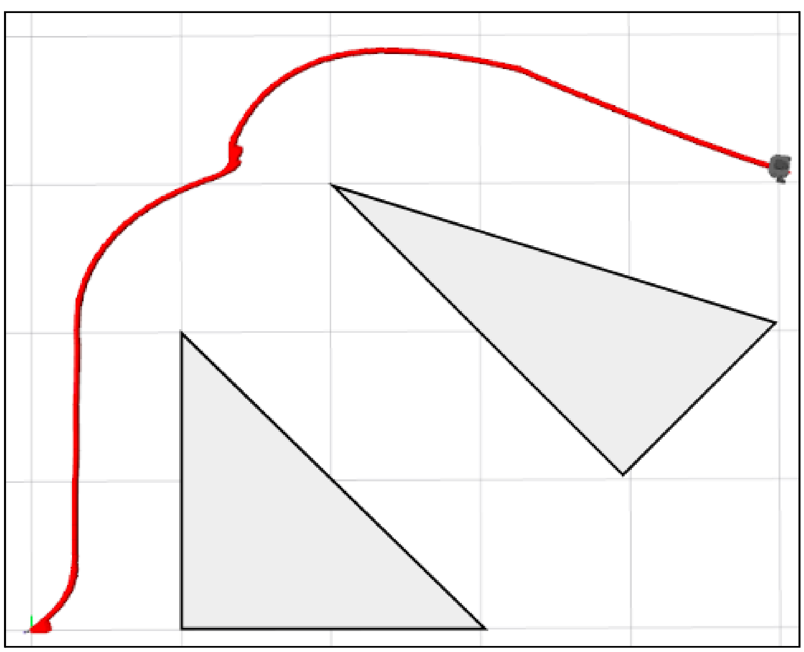
We will add attractive and repulsive gradients by all the obstacles and compute a unit vector along the net gradient vector. Then we will take a step distance along this unit vector.

The following plots show the path traversed by turtlebot:



- **Execution Time:** 194.82 sec
- **Total Path Length:** 8.055 m

Following is the path traversed in simulation (RVIZ/Gazebo):



Note: For the given values of parameters, the bot gets stuck at $(x_e, y_e) = (5.1, 3.2)$, it doesn't reach the goal. The reason behind this is that at the point (x_e, y_e), the repulsive potential becomes equal to the attractive potential, thus it keeps oscillating around that point. In order to mitigate this inconsistency, I change the value of parameter X, when it is inside the 2m(d_star) radius circle. Increasing this value to 4.5 leads the bot to the goal position.

The following is the plot of the distance of the turtlebot from the goal during its motion: