SC-627 Assignment 2

Dikshant 180040033

Potential path planner algorithm:

- Attractive Potential

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

with $\chi = 0.8$ and $d^\star_{goal} = 2$

- Repulsive Potential (for each obstacle)

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

with $\eta = 0.8$ and $Q^\star_i = 2$

The artificial potential field (APF) algorithm is used in robot path planning in which its force F_net, is the sum of the attractive potential field F_att and the repulsive potential field F_rep as shown in the following equation:

$$F_{net} = F_{att} + F_{rep}$$

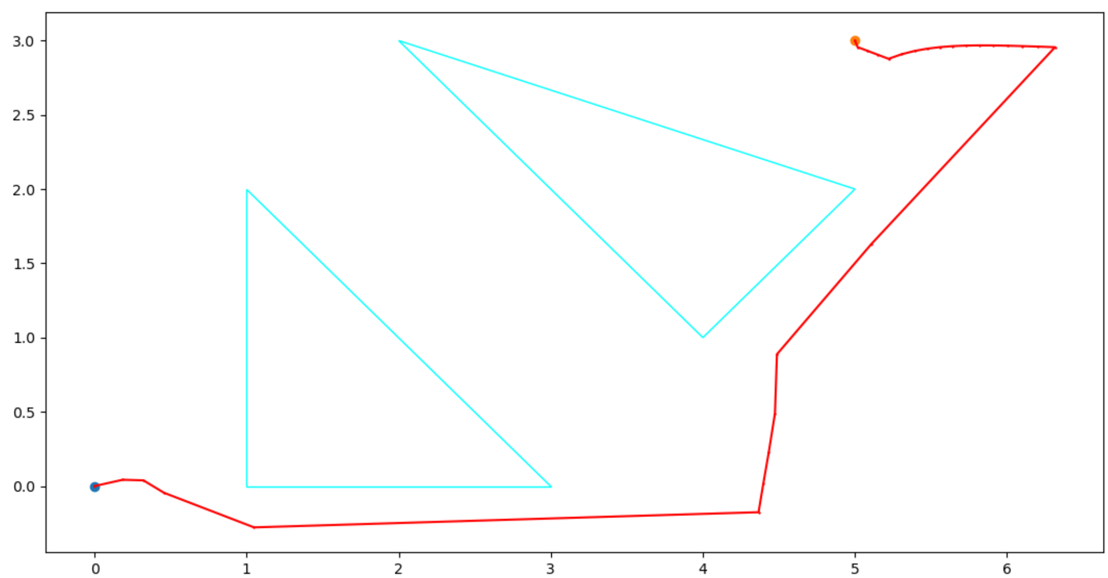
The gradient is computed with respect to d(q, q_goal). We move in the direction of steepest gradient descent using the update rule:

$$q = q - step * \nabla F_{net}$$

Its ROS implementation wasn't that perfect as it is not able to reach its end goal in a few minutes. This can be due to some delay in the processing of the PC and the actual update in the position of the bot. Thus, it will have some different attractive and repulsive fields as compared to its python implementation.

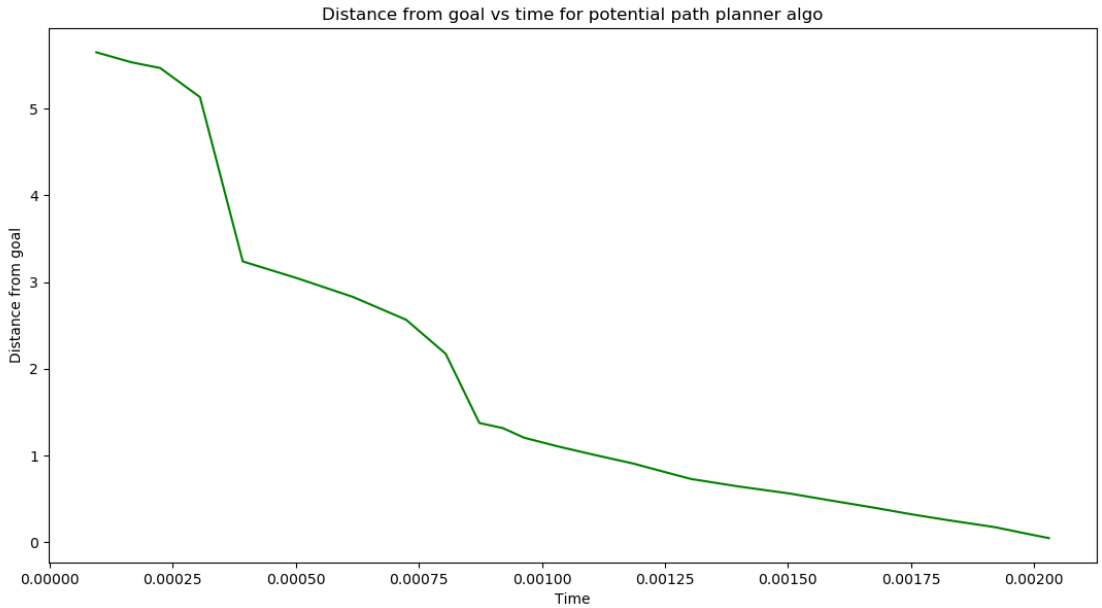
Results:

Path Traced by Potential Path Planner algorithm:



Total Path Length: 8.9 m
Computing Time: 175 sec

The plot of the distance from the bug's position to the goal as a function of time.


Distance from goal vs time for potential path planner algo
Distance from goal
Time
0
1
2
3
4
5
0.00000
0.00025
0.00050
0.00075
0.00100
0.00125
0.00150
0.00175
0.00200