[CS-475] Assignment 6: Path Planning

Michael Maravgakis
maravgakis@csd.uoc.gr

Release date: 06/05/2022
Deadline: 19/05/2022

1 Overview

In this assignment you will implement a path planning algorithm that can be used for autonomous navigation and will guide your robot to a fixed goal position. The algorithm you are going to create is the Artificial Potential Field (APF). The idea behind APF is that you assign an imaginary ("artificial") repulsive field to every obstacle and an attractive one to the goal position. The artificial forces applied to the robot guide it to the potential global minimum, i.e. the goal point. The artificial fields can be visualized as follows:

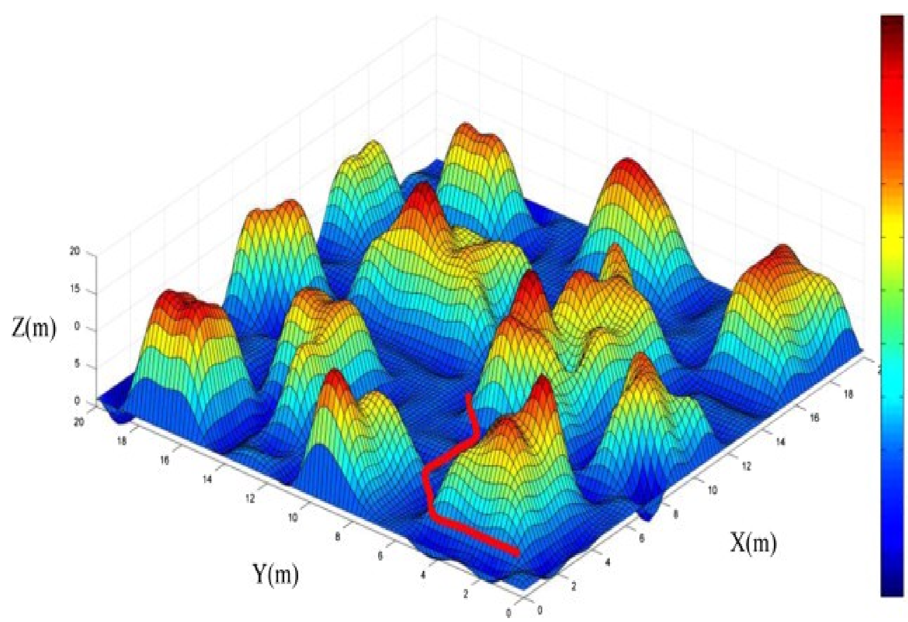

Figure 1: Artificial Potential Fields

2 Installation

In your zip file, you are provided with a ros package named assign6/. As always copy paste it to your ros workspace and compile. In case you need more information, check out the previous assignments. This package contains a launch file (*"burger.launch"*), that opens the gazebo simulation with the turtlebot and also creates a topic "/obstacles" where the position of every obstacle is published.

3 Implementation

For this assignment, you are not provided with any code and you will need to create a ROS node from scratch named *"apf.py"*. The final goal of the assignment is to visualize the path that the robot would need to follow in order to navigate to the goal point in Rviz. The actual navigation part of the robot is **BONUS 10 %**. The initial robot position is (-2,0), the goal position is arbitrary (set it to wherever you want) and poses of the obstacles are published at the topic /obstacles.

There are two types of potential fields namely attractive and repulsive. The attractive is the one placed at the goal point and practically it tries to pull the robot towards its position. The formula that describes this potential is:

$$U_{att} = \frac{1}{2} k_a r^2 \tag{1}$$

where k_a is the attractive constant (i.e. how strong is the field) and r is the distance between the robot and the goal point.

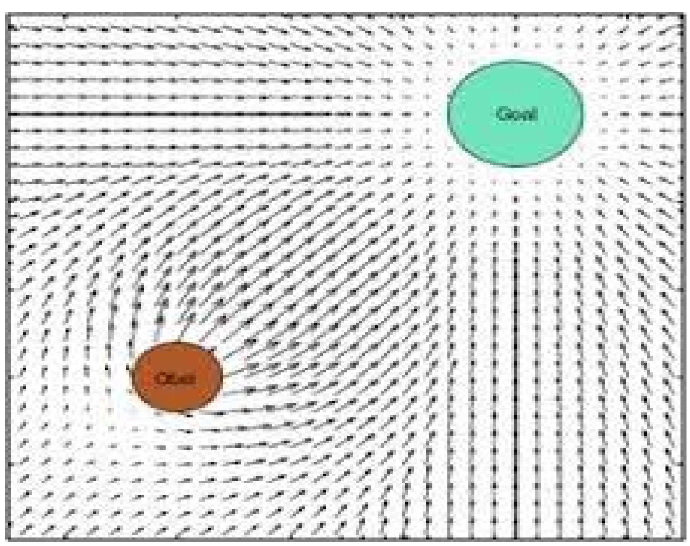

Figure 2: Attractive potential field

The repulsive potential is placed at each obstacle and can be described with the following formula:

$$U_{rep} = \begin{cases} \frac{1}{2}k_r(\frac{1}{r} - \frac{1}{r_0})^2 & r \leq r_0 \\ 0 & r > r_0 \end{cases}$$

where, k_r is the repulsive constant (how strong the repulsive field is), r_0 is the distance threshold and r is the distance between the robot and the obstacle. If this distances is less than the threshold then the repulsive field is applied.

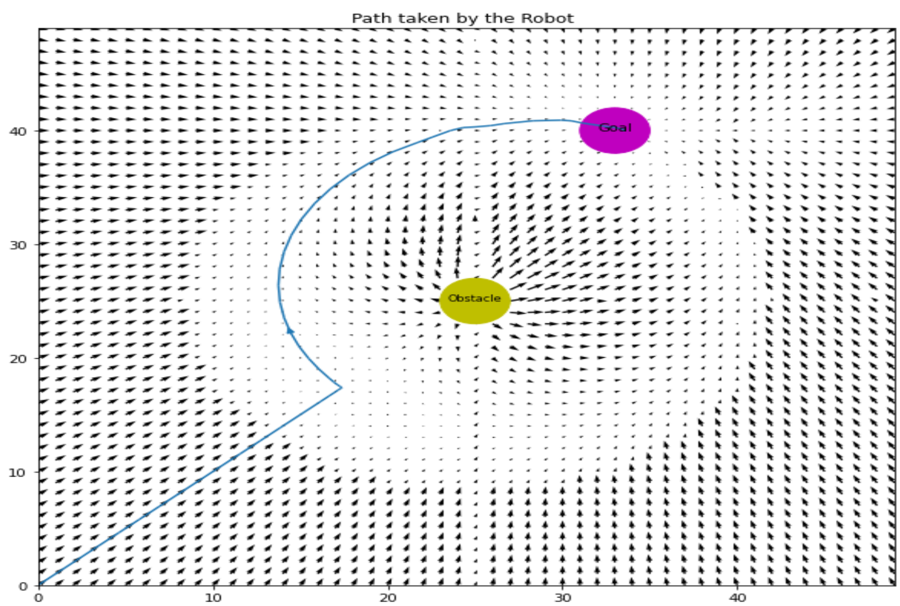

Figure 3: Total potential field and path

You can implement the algorithm by creating a grid map and calculating the total potential value at each cell. At each update step you follow the smallest value of the total potential around the current position until you reach minimum.

You can also calculate the gradient of the above potentials and each time compute the total force applied to the robot (magnitude and direction).

For the implementation, you can use the `nav_msgs/Path` for visualizing the path in rviz. For more information run:

```
$ rosmsg show nav_msgs/Path
```

To visualize the path you need to publish the aforementioned message at a new topic and open rviz:

```
$ rosrun rviz rviz
```

then click *add* and *by topic* and choose the topic you've created. You should see a line that creates the desired path.

Finally, if you want to navigate the robot (bonus part) to follow the created path, you will need to publish at the `/cmd_vel` topic after you've transformed the desired position and orientation to linear/angular velocity and time duration of the command. (You can subscribe to /clock in order to get the time)

4 Submission

Sent your node (`apf.py`) attached via email at: **maravgakis@csd.uoc.gr** with subject `"[CS-475] Assignment 6 submission"` Don't forget to mention your name and registration number. The deadline is at **19/05/2022 23:59**