9/10

Robot Design Lab
Tutor: Teena Hassan, Mihaela Popescu

WiSe 2022/2023
Group: 09

Joel Köper
Maarten Behn
Nhu Van Nguyen

Worksheet Bonus

Hand-in date: December 22, 2022

---

1 Quiz on ROS 2 Navigation Stack [5%]

Points = 4%

1. **Path Planner:** ***Which path planning algorithms does the Navfn Planner plugin support? How can you select which of these algorithms should be used? [1%]***

   The Navfn Planner plugin supports A* and Dijkstra's algorithm.

   You can select the algorithm by using:

   ```
   <name>.use_astar
   ```

   Whether to use A*. If false, it uses Dijkstra's expansion.

   This example would select A* as algorithm, because the bool data type is true:

   ```
   use_astar: True
   ```

2. **Path Planner:** ***Why does the Navfn Planner plugin include a parameter called tolerance? [1%]***

   goal pose

   Tolerance is the meters between the requested start point and end of path.

   It is necessary because the NavFn Planner will attempt to create a minimum cost plan which has to be as close to the specified goal as possible but not beyond the `default_tolerance`.

3. **Dynamic Window Approach:**

   (a) ***How would you set the following DWB plugin parameters for your TurtleBot3 Burger robot? Justify your answer. [1%]***

   - `max_speed_xy`

   Maximum translational velocity: 0.22 m/s.

   - `max_vel_theta`

   Maximum rotational velocity: 2.84 rad/s (162.72 deg/s).

   Because these are the set hardware specifications for the TurtleBot3 and it's beneficial to simulate accurately. The simulation velocity can be set higher, but the robot obviously can't go beyond it's physical boundary anyways.

(b) ***Plot the velocity space based on the values you set for the above parameters. [1%]***

Velocity (v): 0.22 m/s.

Angular velocity (ω): 2.84 rad/s

Radius (r): 0.07746 (calculated by the other two values).

The velocity space is defined by min and max values for linear velocity and angular velocity.

4. **Dynamic Window Approach:** ***Explain how TwirlingCritic and PreferForwardCritic would influence the choice of trajectories by the dynamic window based approach. [1%]***

- TwirlingCritic is a weighed scale for critic and it prevents the robot from spinning while moving to it's goal.

- PreferForwardCritic scores trajectories that move the robot forwards more highly.

These critics influence the score that the dynamic window approach (DWA) calculates, because the score is based on the set of velocities that the robot can reach during a short time interval and the chosen highest score will steer the robot without causing an collision with nearby obstacles.

To be more precise, the TwirlingCritic alters the score, because it gives a cost based on the spinning from the robot on the way to the goal and the PreferForwardCritic affects the choice of the DWA, because it can give penalties to the score if the trajectories move backwards or turn to much.

Generally speaking these types of critics can give penalties to the scores that the dynamic window approach uses to choose the velocity that will steer the robot.

2 Robot Navigates to Goal [5%]

Points = 5%

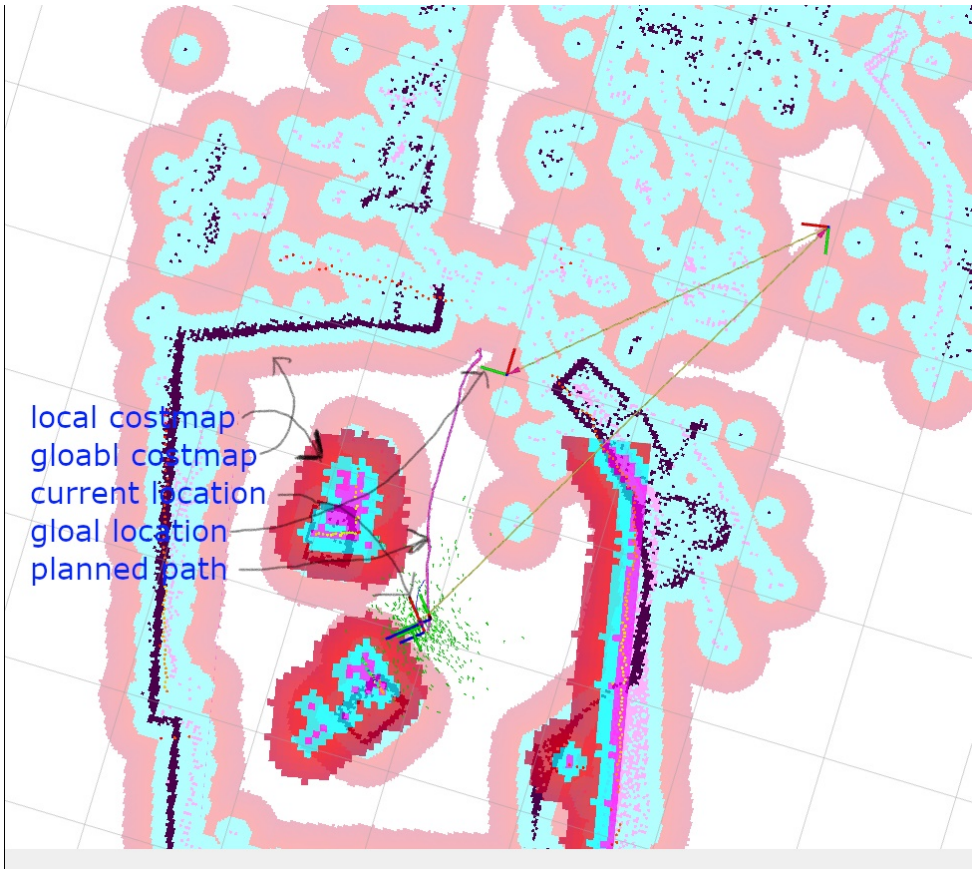

Figure 1: labeled screenshot of the Path planning visualisation.

The global cost-map is calculated from the given map and is primarily used to plan a path from the robot position to the goal position. The local cost-map is created from the online lidar data. It is used to adapt the path so that the robot doesn't run into any obstacles that are not in the given map. This way the robot can follow a path to a goal outside of its Lidar field of view without the chance of running so unexpected obstacle.