Mini Hack 2 - Writeup

Adolfo Roquero, Leticia Schettino, Kobi Green

February 2022

[Testing Robot Navigation in RVIZ]

Q1) Discuss with your team what each of the selected option displays. What are the different maps? Feel free to ask any questions to the TFs. Include these details in your write-up.

- Grid: This display shows a grid of lines (mesh), which is centered at the origin of the target frame.
- RobotModel: This display shows the illustration of the robot in the grid in its correct poses according to the tf transform.
- TF: This display shows the tf transform tree and lets you visualize the transform hierarchy. This is the display that connects the coordinate frames together.
- Map + MoveBase: Both of these displays are responsible for showing the mapping. The Map displays shows the occupancy grid with black on that map display meaning that area on the map is occupied, white meaning that that area on the map is unoccupied and gray meaning that it is unknown whether the area is occupied or not. This spatial mapping is based on the 2D lidar. The MoveBase display shows the cost map for navigation
- Camera (image): The Camera display creates a new rendering window from the perspective of the onboard camera, and overlays the image from the camera on top of it.

There are 3 different maps that we see in RViz, the 2D map, the 3D map and the cost map, which you can visualize below. The 2D map is the map generated with the data coming from the 2D lidar sensor. The 3D pointcloud map is generated using the data from the depth camera. Meanwhile, the cost map that we see in RViz is generated based on the 2D lidar map and captures obstacles around the robot, assigning a buffer region around the obstacles so that the robot can move safely without hitting the obstacles

Figure 1: 2D map

Figure 2: 3D pointcloud map

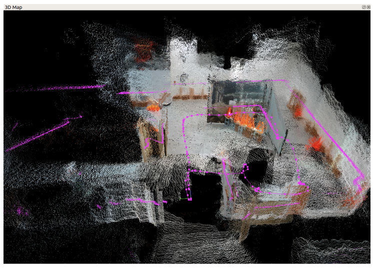
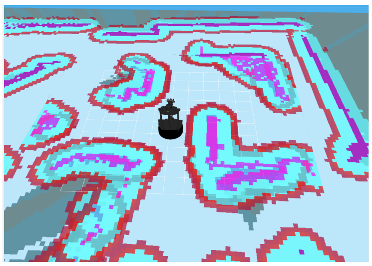


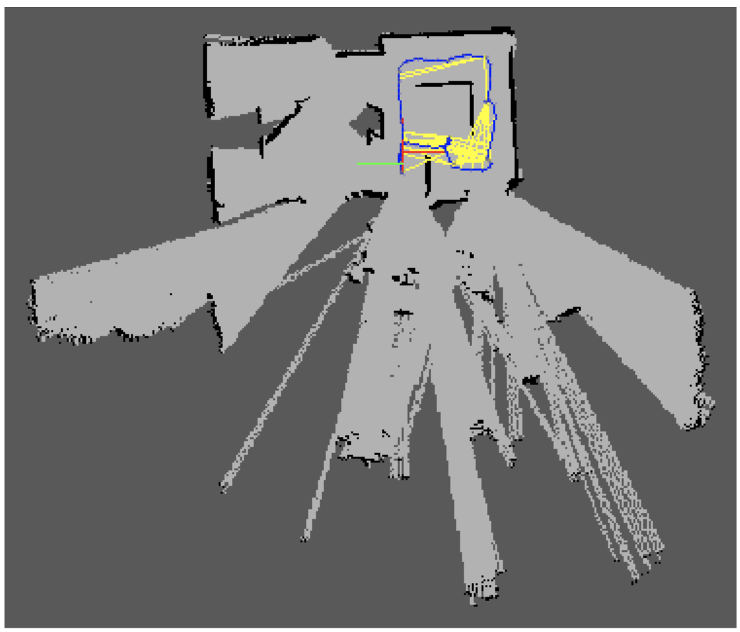
Figure 3: Cost map



Task 1

Using RVIZ, move the robot in roughly 2 x 2 m square (or any 4 locations that are not too far apart). Copy the .db file and generate the 3D and 2D map using the database viewer tool. Save the image of these maps and submit them as part of your write-up.

Figure 4: 2D map generated from robot moving in a square trajectory with Rviz navigation

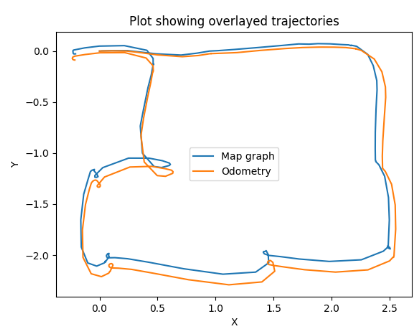


Task 2

Using the waypoint script, move the robot in a square (doesn't need to be an exact square shape) within the testbed area.) Copy the .db file to you PC. Parse the x-y positions from the two files and generate a 2D plot using python matplotlib which shows the two trajectories overlayed with each other. Add the plot to your write-up

Note: Unfortunately, I mistakenly overwrote the .db file for this task with another .db file, which is why there are a couple more waypoints. This mistake was only caught during the write up, when I was no longer in the lab and there was not enough time to return to SEC and perform this experiment again. However, since I beleive the purpose of this task is to compare the trajectory of the odometer with the trajectory according to the map graph, I hope this graph will be acceptable. The rectangular diversion the robot makes when completing the square (at coordinate 0, -1) is due to the fact that we were working in the testbed on the right and there is an obstacle at that point.

Figure 5: 2D plot of overlayed positions from robot moving in a square trajectory



Briefly explain in your write-up how the two trajectories compare in terms of their accuracy.

The two trajectories follow each other very closely, which is a good sign. The differences in the trajectory are likely a result of noise in the odometer sensor as well as noise in the lidar and depth camera sensors that are used by SLAM to create the map and the drifting that occurs in positions given by the odometer. The odometer trajectory seems slightly more accurate and it could be due to the fact that there are more sensors involved in the map graph and therefore there are more sources of noise.

Task 3

Q3) Submit the generated 3D map, 2D map and the 2D plot that shows the overlay of the optimized positions and the odometry positions

Figure 6: 2D cost map

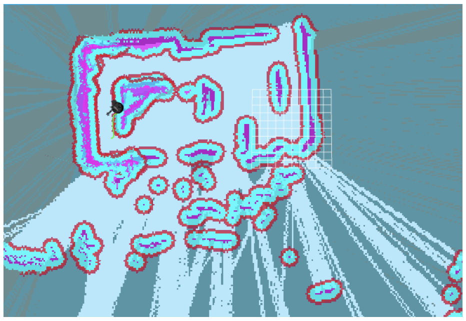


Figure 7: 2D map

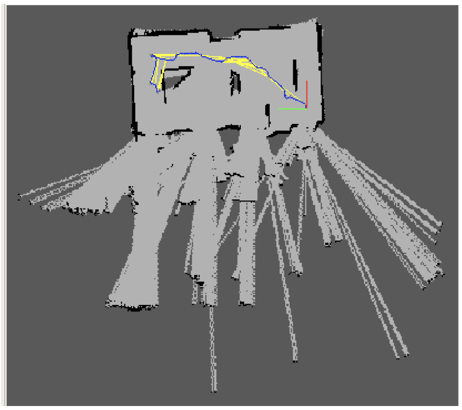


Figure 8: 3D map - bird eye's view

Figure 9: 2D overlay positions plot