University of Bremen

German Research Center for Artificial Intelligence

Worksheet 03: Mapping and Localization

Deadline: 30.11.2022
Robot Design Lab
03-IBGA-FI-RDL
WiSe 22-23

Universität Bremen
FB 3 – Mathematik und Informatik
Arbeitsgruppe Robotik
Prof. Dr. Dr. h.c. Frank Kirchner
Dr. rer. nat. Teena Hassan
M. Sc. Mihaela Popescu

1 Mapping

1.1 Quiz on Mapping 15%

Please provide the answer to the following theoretical questions:

1. What is mapping in robotics? (3%)
2. What is the purpose of loop closure in localization and mapping? (3%)
3. What is the meaning of the following occupancy grid mapping algorithm parameters? (3%)
   - map resolution (~delta)
   - occupancy threshold (~occ_thresh)
   - map update rate (~map_update_interval)

   Hint: You can use the official documentation of SLAM Gmapping[1], but you should formulate the answers with your own words.
4. Explain the laser scanner inverse sensor model used for occupancy grid mapping. (4%)
5. Probabilistic maps: If the robot senses an obstacle 2 meters ahead, why don't we directly enter this information into the map? Why do we need a model based on probability? (2%)

1.2 Occupancy Grid Mapping 30%

In this section you will create a map of the environment for the TurtleBot. You will use the generated map to localize the robot, plan a path to a goal in the environment and navigate autonomously.

First, connect to your turtlebot robot and launch the bring up file. In order to visualize the laser scanner data used for mapping and localization, install and configure the Rviz visualizer following the next steps on your PC:

```
sudo apt update
sudo apt install ros-humble-rviz2
source /opt/ros/humble/setup.bash
rviz2
```

Once Rviz is open, activate the laser scanner point cloud visualization:

- Fixed Frame: `odom` or `base_scan`.
- Add → By topic → LaserScan.
- LaserScan → Topic → Reliability Policy → Best effort.
- LaserScan → Size → 0.05.

[1] SLAM Gmapping Documentation: http://wiki.ros.org/gmapping

For the mapping task, you will use the `slam_gmapping` library for ROS2[1]. This package is already installed on your TurtleBot. Follow the next steps to start the mapping node:

1. Connect to the TurtleBot using `ssh` in a new terminal window.
2. Run the mapping node on the robot and setup the resolution parameter delta:

```
source /home/ubuntu/rdl_ws/install/setup.bash
ros2 run slam_gmapping slam_gmapping --ros-args -p delta:=0.05
```

3. Visualize the map in Rviz: Add → By topic → Map.
4. In order to update the map, move your robot around using the `rdl_teleop_keyboard`, `wheel_velocity_controller` and `rdl_move_base` from `worksheet 02`.
5. You should now see in Rviz that the map is updating. Check that the map is published on the topic `/map` with the message type `nav_msgs/msg/OccupancyGrid` using `ros2` commands.
6. In order to save the map, install the navigation package on your PC and start the `map_saver_cli` node:

```
sudo apt install ros-humble-navigation2
ros2 run nav2_map_server map_saver_cli -f map
```

7. You should now see a file called map.pgm in the current path. You can open the map with the image viewer:

```
eog map.pgm
```

Mapping Task

1. Create a map of the maze using the following values for the map resolution parameter `delta`: **0.01, 0.05, 0.5**. Save the three maps. (15%)
2. Tune the parameter `delta` until you obtain a map that looks desirable to you in terms of minimising the chances of hitting obstacles and minimising the storage space. Save this map. (5%)
3. Describe your observations and findings during mapping. Tell which value you found for the `delta` parameter in subtask 2 and motivate your choice. (10%)
4. Include the `map.pgm` and `map.yaml` files from subtasks 1 and 2 in your `.zip` submission. Convert the `map.pgm` images to `.pdf` format and insert them into your PDF report.

2 Localization

2.1 Quiz on Localization 15%

Please provide the answer to the following theoretical questions:

1. What is robot localization? (2%)
2. What is the kidnapped robot problem and which strategy can be applied to cope with it? (2%)
3. Monte Carlo Localization can work without being adaptive. What does *Adaptive* in AMCL mean and why is it useful to have an *Adaptive* Monte Carlo Localization? (3%)
4. Explain the following sensor model parameters of the AMCL algorithm (8%):
   - measurement noise (`z_hit`)
   - maximum range (`z_max`)
   - random measurement (`z_rand`)
   - unexpected obstacles (`z_short`)

   Hint: You can use the documentation of AMCL package[2], but you should formulate your own answer.

[2] AMCL Documentation: https://navigation.ros.org/configuration/packages/configuring-amcl.html?highlight=amcl

2.2 Adaptive Monte Carlo Localization 30%

In this section, the robot will perform localization into an occupancy grid map similar to the one from section 1.2. For this purpose, you will use the Adaptive Monte Carlo Localization package `nav2_amcl`[2].

In order to start the localization node, follow the next steps:

1. Connect to your TurtleBot robot and launch the bring up file.
2. Download the map files from StudIP (alternatively, use your own map files), navigate to the path where the files `map.pgm` and `map.yaml` are on your computer, copy them on the robot and build the package:

   ```
   scp map.pgm ubuntu@192.168.1.xx:/home/ubuntu/rdl_ws/src/slam_gmapping/maps/
   scp map.yaml ubuntu@192.168.1.xx:/home/ubuntu/rdl_ws/src/slam_gmapping/maps/
   ssh ubuntu@192.168.1.30+no.of the robot
   cd ~/rdl_ws/
   colcon build
   ```

   Remark: Your map should be named `map`, otherwise it will not be found by the `amcl` launch file.
3. Open `rviz2` on your computer and add the map visualisation by selecting: Add → By topic → Map. Make sure to select Fixed Frame: map.
4. Open a new terminal window, connect to the TurtleBot and start the localization launch file:

   ```
   source /home/ubuntu/rdl_ws/install/setup.bash
   ros2 launch slam_gmapping amcl.launch.py
   ```

5. Run `ros2 topic list` on your PC. You should see the following topics:
   - `/map` of type `nav_msgs/msg/OccupancyGrid`. The map server publishes the static map on this topic and makes it available by request to the subscribers.
   - `/particle_cloud` of type `nav2_msgs/msg/ParticleCloud`. Here is where all the particles and their weights are published by the AMCL algorithm. In order to visualize them in `rviz2`, click Add → By display type → `nav2_rviz_plugins` → ParticleCloud, then on the left menu select Topic: `/particle_cloud`.
   - `/amcl_pose` of type `geometry_msgs/msg/PoseWithCovarianceStamped`. This is the robot pose estimated by the particle distribution. In order to visualize it in `rviz2`, click Add → By topic → PoseWithCovariance.

   Remark: For `/particle_cloud` and `/amcl_pose`, please select Topic → Reliability Policy → Best effort.
6. Use the global localization service of the `amcl` package to initialize the particles randomly and uniformly across the entire map. Run the following command in a new terminal:

   ```
   ros2 service call /reinitialize_global_localization std_srvs/srv/Empty
   ```

7. Teleoperate the robot using `rdl_teleop_keyboard`, `wheel_velocity_controller` and `rdl_move_base` from `worksheet 02` and see how the poses are converging in `rviz2`.

Localization Task 1: Navigate in the map until all particles converge to a small area. Take screenshots of the particles and the `amcl` pose visualization in `rviz2` at a few seconds interval. Include in your solution 4 screenshots that show the progress of the AMCL algorithm from the initial robot state until the final state. Describe the evolution of the uncertainty ellipses for the robot position and orientation. What do they represent? (15%)

Localization Task 2: You will now kidnap the robot! After the AMCL algorithm converges to a final pose, move the robot to another part of the map. Navigate around and see whether the AMCL algorithm is able to recover, namely to show the correct robot pose again. When the localization fails, try out the following three methods to recover from kidnapping and briefly explain the advantages and disadvantages of each method. (15%)

1. The `amcl` node has two parameters `recovery_alpha_fast` and `recovery_alpha_slow` used in deciding when to recover from kidnapping by adding random poses. By default, these parameters are zero, which means that no random particles are added. Relaunch the localization and set the recovery parameters as follows:

```
source /home/ubuntu/rdl_ws/install/setup.bash
ros2 launch slam_gmapping amcl.launch.py recovery_alpha_slow:=0.001 recovery_alpha_fast:=0.1
```

2. Provide an initial guess of the robot pose. Go to Rviz, click on the `2D Pose Estimate` arrow, then click on the map where the robot is and hold to select the orientation of the robot.

3. Reinitialize the global localization and redistribute all particles randomly on the map:

```
ros2 service call /reinitialize_global_localization std_srvs/srv/Empty
```

3 Quiz: Battery Handling 10%

Please provide answers to the following questions:

1. Batteries lose capacity over time. What do you need to do in order to ensure a long lifetime of the battery? List two measures you can take. (2%)
2. You want to build a new, powerful wheeled robot from scratch! Name two safety precautions you would consider: (i) when it comes to selecting an appropriate placement for the batteries on the robot, and (ii) while mounting the battery onto the robot. (2%)
3. List two types of fire extinguishers. Which one can be used in the case of a fire caused by LiPo batteries? (2%)
4. Name two differences between LiPo and LiFe batteries. (2%)
5. On which ROS 2 topic can you check the status of the battery of your TurtleBot 3? What is the type of message used to report the battery status on this topic? (2%)
Hint: You can use commands such as `ros2 topic list`, `ros2 topic info`.

4 Feedback

Your feedback is very important to us. Please briefly answer the following questions:

1. How much time did you spend on doing this sheet per person? Anonymize your answer!
2. Was it too easy, easy, ok, hard, too hard?
3. Please tell us what you liked in this exercise sheet.
4. Did you face any difficulties? What should be improved?
5. Any other general remarks?
6. ROS 2 Humble installation:
   (a) In the first tutorials, did you face any issues during the installation of ROS 2 Humble on your notebook? If yes, please let us know what issues you faced, and how they were resolved.
   (b) Do you still have any unresolved issues?

5 Submission Procedure

- Please use the LaTeX template provided in *StudIP/Wiki* to write your solutions. Upload the PDF file together with source code and other additional materials as a .zip file in StudIP.
- The naming style of your submission should follow the pattern **Gxx_0y_lastname1_lastname2_lastname3.zip**, where *xx* stands for the group number and *y* stands for the exercise sheet number.