ROBT 403 Laboratory Report 5 & 6

Daniil Filimonov
Robotics Engineering, SEDS
Nazarbayev University
Astana, Kazakhstan
daniil.filimonov@nu.edu.kz

Abdirakhman Onabek
Robotics Engineering, SEDS
Nazarbayev University
Astana, Kazakhstan
abdirakhman.onabek@nu.edu.kz

Kir Smolyarchuk
Robotics Engineering, SEDS
Nazarbayev University
Astana, Kazakhstan
kir.smolyarchuk@nu.edu.kz

***Abstract*—This report demonstrates the outcome of tasks implemented for Laboratory works 5 & 6. In Lab 5, the robot was assembled and configured for teleoperation, while a custom labyrinth was constructed based on predefined specifications, ensuring compatibility with the robot's LiDAR and navigation requirements. Lab 6 extended these efforts by implementing simultaneous localization and mapping (SLAM) techniques to enable the robot to autonomously explore and navigate the labyrinth. Key tasks included parameter tuning for SLAM accuracy, integration of teleoperation and path-planning algorithms, and validating the robot's performance in a structured environment. The experiments highlighted the challenges and solutions in achieving autonomous mobile robot navigation, providing valuable insights into the interplay between hardware, software, and algorithms. The findings contribute to a deeper understanding of mobile robotics and its potential real-world applications.**

I. Introduction

Mobile robots are increasingly being employed in diverse applications ranging from warehouse automation to autonomous navigation in hazardous environments. This report outlines the activities undertaken in Labs 5 and 6 of the Robotics II course, centered on the TurtleBot 3 Burger, a compact differential-drive robot designed for research and education. The TurtleBot 3 Burger features a Raspberry Pi-based control unit, an OpenCR board for motor control, and a 360-degree LiDAR sensor for environment mapping and navigation. Its lightweight, modular design and compatibility with the Robot Operating System (ROS) make it an ideal platform for understanding the principles of robotics and automation.

LiDAR (Light Detection and Ranging) is a pivotal sensor in mobile robotics, enabling precise distance measurements by emitting laser pulses and detecting their reflections. This technology is critical for obstacle detection, mapping, and navigation. In these labs, LiDAR was utilized in tandem with ROS packages to implement Simultaneous Localization and Mapping (SLAM), a computational technique that allows a robot to build a map of its environment while tracking its location within the map. SLAM is fundamental for autonomous navigation in unstructured environments and integrates sensor data, odometry, and probabilistic algorithms such as particle filters and Extended Kalman Filters [1].

In Lab 5, the primary focus was on robot assembly and configuration, including setting up the Raspberry Pi, installing ROS Noetic, and testing teleoperation capabilities. A custom robot labyrinth was constructed to evaluate the robot's navigation capabilities. The specifications for the labyrinth ensured the walls exceeded the LiDAR height, and the corridors were wide enough to accommodate the robot's differential drive. Teleoperation using ROS commands facilitated initial exploration and system validation.

Lab 6 advanced the capabilities of the TurtleBot by incorporating SLAM for real-time mapping and localization. The `gmapping` package was employed to create 2D occupancy grids of the labyrinth, enabling the robot to autonomously navigate through defined paths. Parameters such as angular update rates and map resolution were tuned to optimize SLAM performance. Additionally, path-planning algorithms such as Dynamic Window Approach (DWA) were utilized for generating local paths aligned with the global map, ensuring smooth and collision-free navigation [2].

The integration of these techniques demonstrated the challenges and intricacies of mobile robotics, particularly in coordinating hardware, software, and algorithms for autonomous operation. The following sections of this report provide a detailed account of the methodologies, experimental setup, results, and insights gained through these exercises.

II. Methods

This section outlines the methodologies for the two labs. Lab 5 focused on constructing a physical labyrinth for robot navigation, while Lab 6 transitioned to simulation-based experiments using Gazebo and ROS to implement teleoperation, localization, and mapping.

A. Lab 5: Physical Labyrinth Construction

The aim of Lab 5 was to construct a physical labyrinth designed for testing mobile robot navigation in later simulations. The process involved:

- **Specifications Design:** The labyrinth specifications ensured compatibility with the TurtleBot 3 Burger. Key considerations included:
  - Minimum wall height exceeding the LiDAR sensor height (10 cm).
  - Corridor widths greater than the robot's base width (13.8 cm).
- **Material Preparation and Assembly:** Materials were provided by the lab coordinators, including cardboard

and wooden supports. Measurements were taken to ensure walls and corridors met the design specifications.

- **Verification:** The labyrinth dimensions were cross-verified with other teams to ensure consistency across different setups.

Photographs and schematics of the constructed labyrinth were documented for reference in the simulation phase.

B. Lab 6: Simulation-Based Localization and Mapping

Lab 6 focused on simulating the TurtleBot 3 Burger's navigation capabilities within the labyrinth using Gazebo and ROS. The methodology included:

1) Simulation Environment Setup: A virtual representation of the constructed labyrinth was created in Gazebo to mimic real-world constraints. The TurtleBot 3 Burger simulation model, including its LiDAR sensor and differential drive, was used. The simulation setup included:

1) Installing ROS Noetic and Gazebo on an Ubuntu 20.04 system.
2) Loading the TurtleBot 3 Burger model and configuring its sensor and movement parameters.
3) Simulating the labyrinth based on the physical design from Lab 5, ensuring similar dimensions and obstacle placement.

2) SLAM and Path Planning Simulation: Simultaneous Localization and Mapping (SLAM) was implemented using the `gmapping` package:

- **SLAM Execution:** The robot autonomously explored the simulated labyrinth, creating a 2D occupancy grid map in real time.
- **Parameter Tuning:** Key SLAM parameters such as `map_update_interval`, `linear_update`, and `angular_update` were adjusted to improve map accuracy and resolution.
- **Path Planning:** Navigation goals were set in RViz, and the robot followed these goals using the Dynamic Window Approach (DWA) local planner, which ensured collision-free navigation through narrow corridors.

3) Teleoperation Testing: Before implementing autonomous navigation, the robot's teleoperation was tested using the `turtlebot3_teleop` package to validate movement and control within the simulated labyrinth. This pro

III. Lab 5

The labyrinth was initially modeled in SolidWorks, using the specified parameters to ensure compatibility with the TurtleBot 3 Burger's dimensions and capabilities. Key considerations included maintaining a minimum wall height greater than the LiDAR sensor's height (10 cm) and setting corridor widths wider than the robot's base width (13.8 cm) to facilitate safe navigation. The SolidWorks model provided a precise and visual representation of the labyrinth layout, allowing for iterative adjustments and validation (figure 1).

Once the design was finalized, the labyrinth was physically assembled using cardboard and wooden supports provided by the lab coordinators. Measurements were taken during assembly to ensure adherence to the modeled specifications, and adjustments were made where necessary. The assembled labyrinth served as a testing environment for subsequent robot simulations and navigation experiments. Resulting labyrinth is shown on Figure 2.

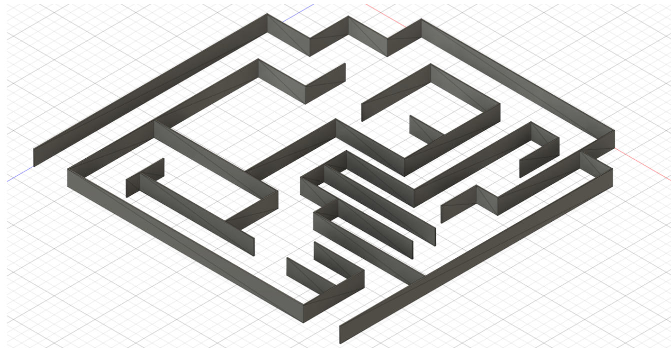

Fig. 1. 3D-Model of the labyrinth made in SolidWorks

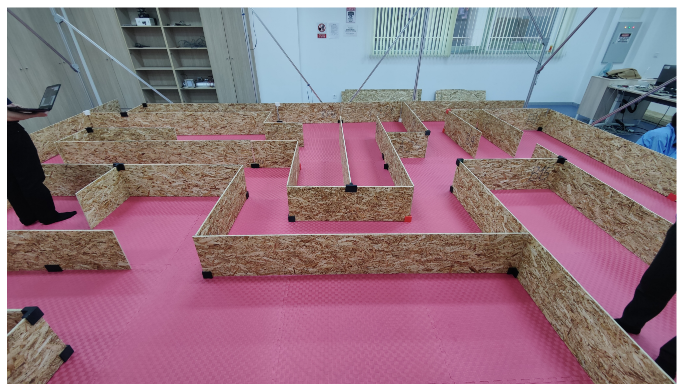

Fig. 2. Assembled labyrinth based on the CAD drawings

IV. Lab 6

In Lab 6, the focus shifted to simulating and testing the TurtleBot 3 Burger's navigation and mapping capabilities within the labyrinth. The labyrinth designed in SolidWorks during Lab 5 was exported as an STL file and imported into the Gazebo simulation environment, ensuring an accurate virtual representation of the physical setup. The TurtleBot 3 Burger model, complete with a simulated LiDAR sensor, was loaded into the environment for further experimentation (figure 3).

The first step involved mapping the labyrinth using Simultaneous Localization and Mapping (SLAM) techniques. The `gmapping` ROS package was employed to enable the robot to construct a 2D occupancy grid of the environment. Initially, manual manipulation of the robot was performed using the `turtlebot3_teleop` package to explore the labyrinth (figure 4). This step allowed the virtual LiDAR to capture distance measurements and generate an accurate map of the labyrinth. Key SLAM parameters, such as `map_update_interval` and `linear_update`, were tuned iteratively to ensure the generated map aligned closely with the modeled environment.

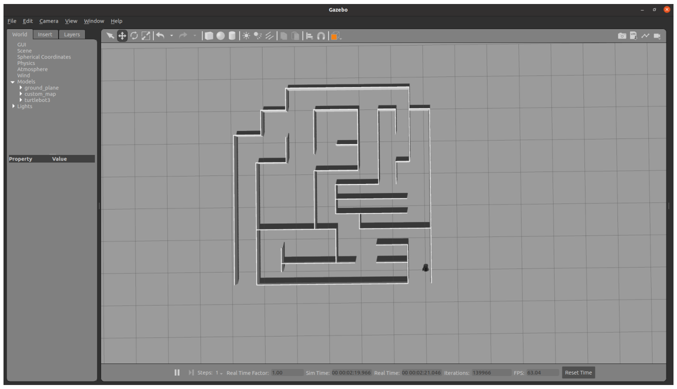

Fig. 3. Labyrinth loaded to Gazebo

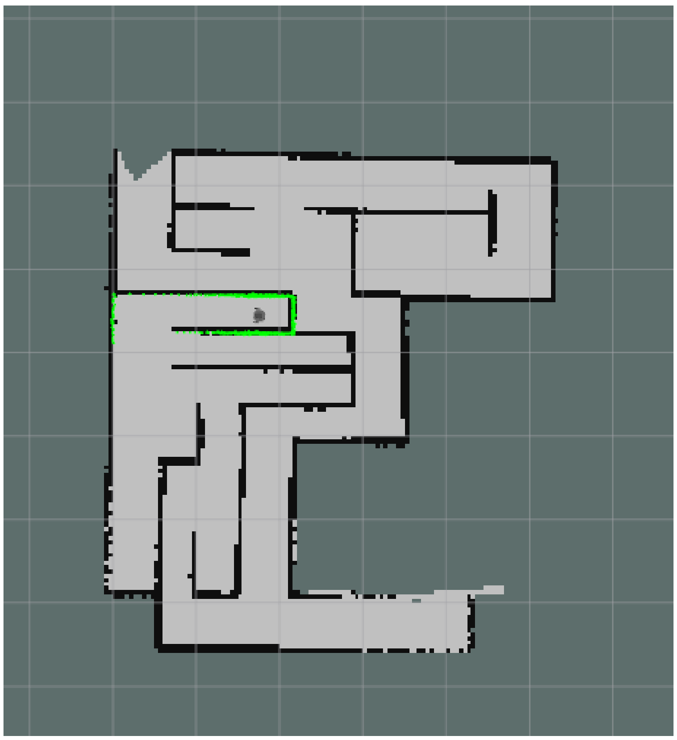

Fig. 4. First map of the labyrinth based on virtual lidar

Once the mapping was completed and validated (figure 5), the robot was tasked with autonomously navigating to a predefined goal position within the labyrinth. Using the map generated during SLAM, the `move_base` ROS package was employed to integrate global and local path-planning algorithms. The Dynamic Window Approach (DWA) planner was configured to handle local navigation, utilizing sensor readings to avoid obstacles and refine movement paths. RViz was used to set navigation goals and visualize the robot's trajectory in real time.

As shown in Figure 6, the robot successfully navigated the labyrinth by combining its sensor data and the pre-mapped environment, demonstrating effective integration of mapping and navigation capabilities. This simulation process

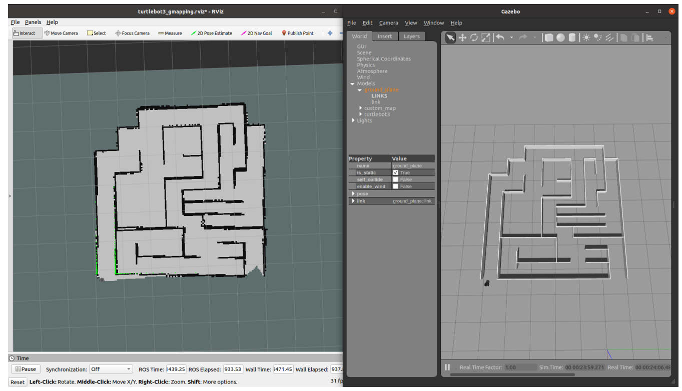

Fig. 5. Final map of the labyrinth based on virtual lidar

highlighted the interplay between hardware abstraction, sensor data, and algorithmic decision-making in mobile robotics.

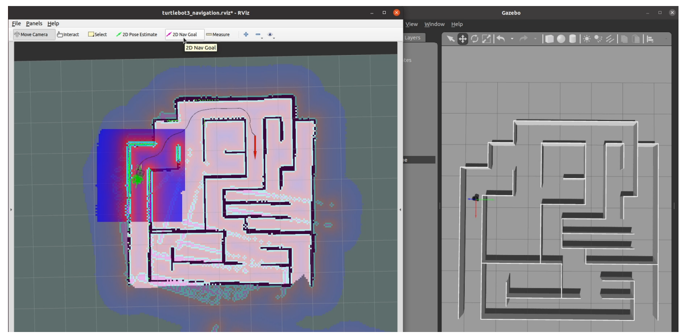

Fig. 6. Robot moves autonomously to the goal position

V. Conclusion

In this lab, we successfully simulated a TurtleBot3 navigating a maze using Gazebo. We began by designing the maze map in SolidWorks and converting it into a compatible world file. Following the detailed instructions on the TurtleBot3 e-manual, we established the necessary ROS environment and controlled the robot in the simulation. This hands-on experience allowed us to understand the process of configuring mobile robots, from creating maps to implementing teleoperation and SLAM techniques. The task enhanced our skills in integrating software tools like ROS and SolidWorks while also reinforcing concepts such as localization, mapping, and robot control. This exercise forms a foundational step towards more complex tasks, such as autonomous navigation and path planning.

References

[1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.

[2] "SLAM gmapping," http://wiki.ros.org/gmapping, accessed: Nov. 19, 2024.