Minor Exam

JRL7000 - Robotics Laboratory

Course Instructor :

Prof. Sunil Jha

IIT Delhi

Instructions

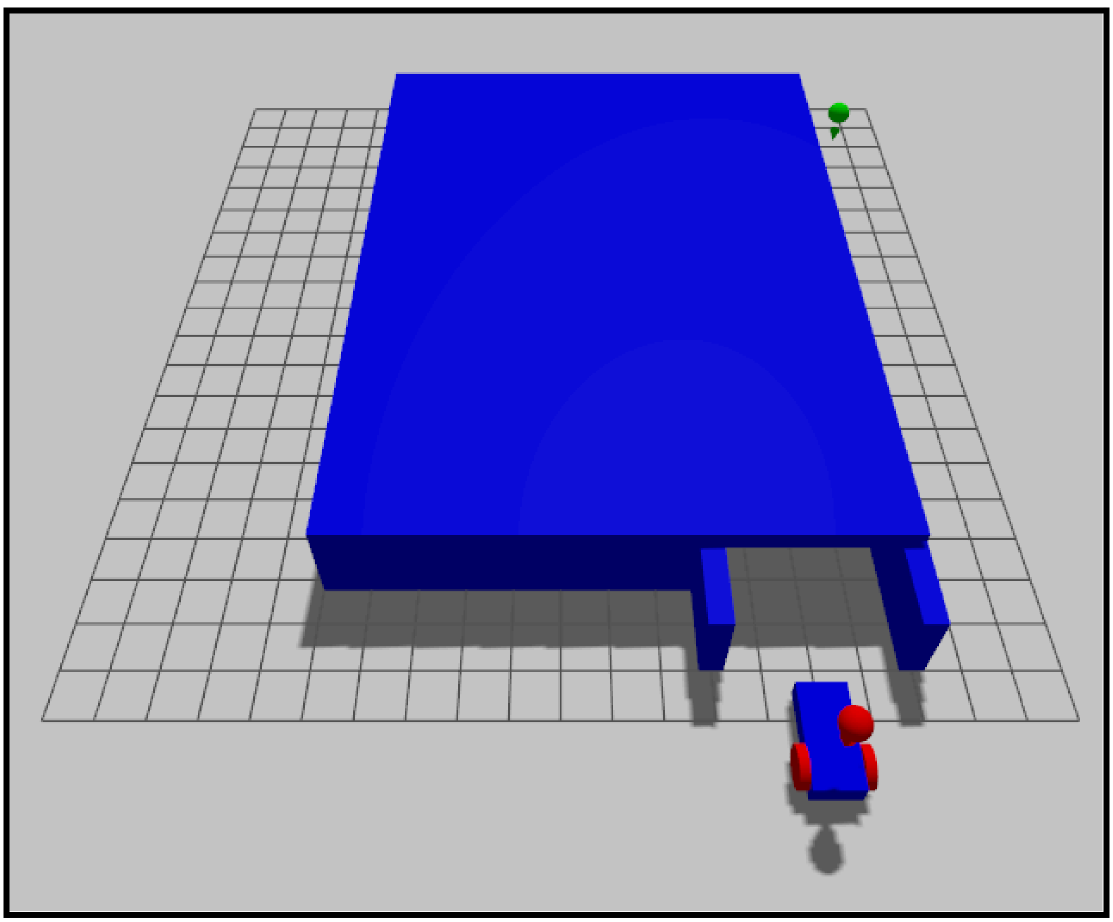
- *You should not use AI generated code however web resources are allowed.*
- *You should be able to explain the code.*

Best of Luck!

Maze Runner

World Description:

Unknown Maze World



The simulation world consists of a simple maze enclosed by blue walls. The maze is built on a flat grid surface and contains narrow corridors with one central divider wall that creates a winding path from the starting point to the goal.

- The robot is a minimal **two-wheeled differential drive robot with a caster wheel** for support.
- The robot is equipped with a **2D LiDAR sensor** as its only sensing modality. No cameras, GPS, or odometry are provided beyond wheel encoders and LiDAR.
- The **red pin** marks the **starting position** of the robot, located at the bottom-right corner of the maze.
- The **green pin** marks the **goal location**, positioned at the top-right corridor of the maze.
- The maze requires the robot to navigate through several turns and avoid collisions with the walls to successfully reach the goal.

Task Description:

1. Visualize complete Maze using **LiDAR** data in **Rviz** by doing **Teleop.**
2. Your task is to implement a **navigation strategy using only LiDAR data** to guide the robot from the **start (red pin)** to the **goal (green pin)**.

Specifically, you are required to:

1. Teleop to visualize the Maze in Rviz and get the idea of the environment.
2. Develop a ROS2 node (or set of nodes) that processes LiDAR data to detect obstacles and free space.
3. Implement an ***obstacle avoidance algorithm*** that drives the differential drive robot safely through the maze.
4. Ensure that the **robot does not collide with the walls** of the maze.
5. Successfully reach the goal location marked by the green pin.

Constraints:

- You may not use pre-built SLAM or navigation stacks.
- You must rely only on LiDAR scan data for perception and obstacle avoidance.
- The robot should demonstrate **autonomous navigation** without manual teleoperation.

Setup Instructions:

You have been provided with a **ROS2 package named** **`minor`**.
To set up the simulation environment:

1. **Use** https://app.theconstruct.ai/ inside the `src` folder of your ROS2 workspace.
2. Open **webshell**
    a. Change directory to ros2_ws/src
    `cd ros2_ws/src`
    b. Clone the Repository
    `git clone https://github.com/NishantWankhade/MazeRunner---ROS2.git`
    c. Rename the folder "***MazeRunner—ROS2***" to "***minor***"
    Rename it as it is, with the CASES.

3. **Build the package** using:
    `colcon build`
4. **Run the launch file** provided in the package:
    `ros2 launch minor <launch_file_name>.launch.py`
5. This will start both the **Gazebo simulator** (with the maze world and robot) and the **RViz tool** (for visualization).

Expected Folder Structure:

```
src
└── minor……………………………………………………………………Package Folder
    ├── launch…………………………………………………………………Launch Folder
    │   └── world.launch.py
    ├── LICENSE
    ├── minor
    │   ├── __init__.py
    │   └── solution.py…………………………………………………………Your Logic
    ├── package.xml
    ├── resource
    │   └── minor
    ├── setup.cfg
    ├── setup.py
    └── world
```

- ***Make changes to all necessary files required to run your implementation.***

Submission Details:

*You have to submit **2 videos**: Align the two tabs side by side, one from Gazebo and other from Rviz.*

1. Record the ***whole screen*** first for Rviz visualizations of LiDAR data - ***Using Teleop***
2. Again Record ***Using your Logic***

Submit all those files where you have made the changes.

- ***Submit as a single zip file. – Moodle***
- ***Submit the Video on the Google Form.***