Autonomous Robot with PID-Controlled Path Following, Obstacle Avoidance, and Real-Time Re-planning

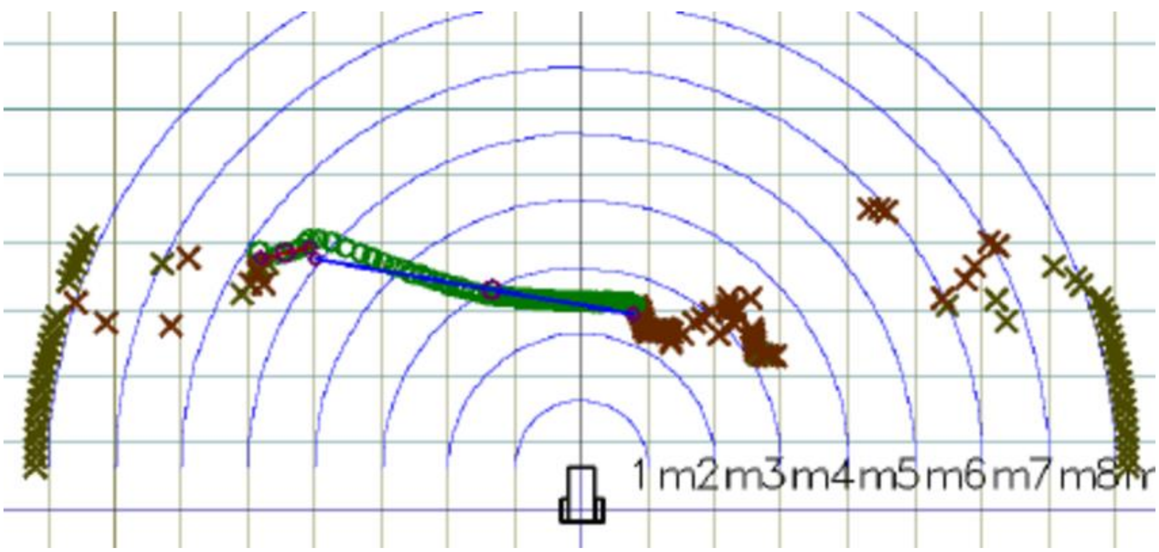
1. Introduction

This report details the integration of Simultaneous Localization and Mapping (SLAM), the A* path planning algorithm, and various sensors to create a complete autonomous robot navigation system. The system leverages LiDAR, IMU, and ultrasonic sensors to enable the robot to map its environment, plan optimal paths, and avoid obstacles in real-time. This is achieved using the Robot Operating System (ROS).

2. System Components

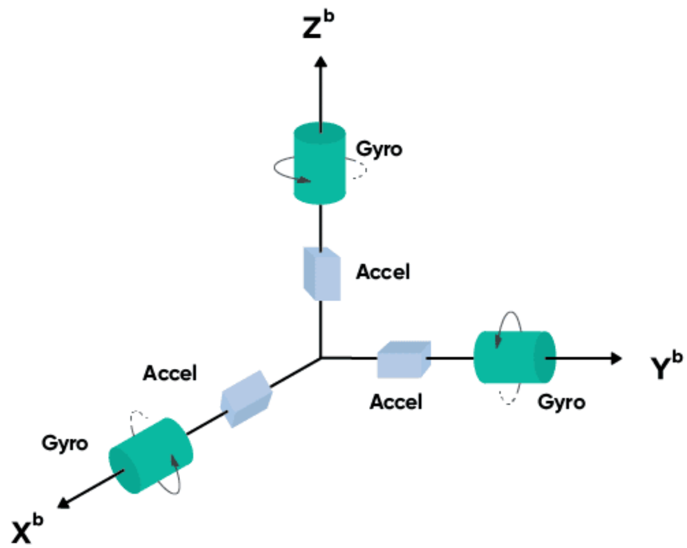
LiDAR (Laser Scanner)

The LiDAR sensor provides 360-degree scans of the environment, enabling the robot to detect objects and map its surroundings. The LiDAR generates distance measurements to nearby objects, which can be used to build a real-time map of the robot's environment. It is crucial for accurate SLAM performance.



IMU (Inertial Measurement Unit)

The IMU tracks the robot's orientation and movement by measuring acceleration and angular velocity. It helps stabilize the robot's motion, particularly during turns or rapid movements, and provides additional data to complement the LiDAR scans in SLAM.



Ultrasonic Sensors

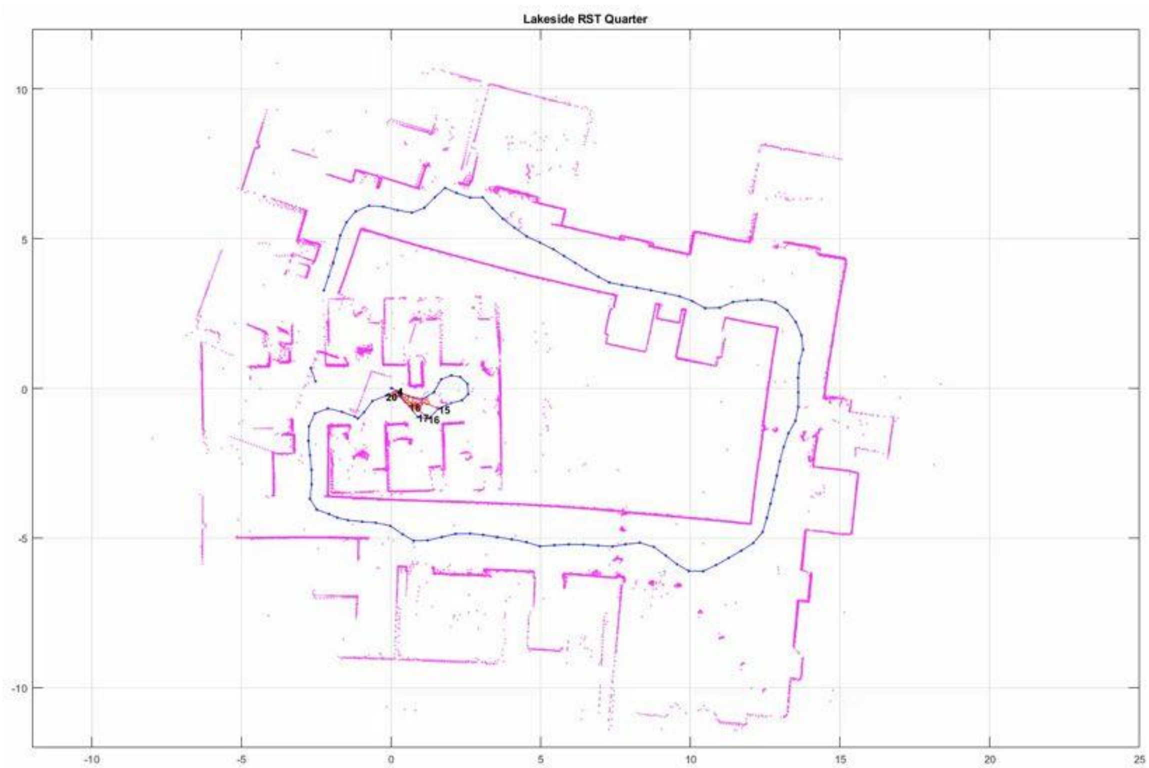
Ultrasonic sensors are used to detect close-range obstacles. They measure the time it takes for sound waves to bounce back from nearby objects, providing short-range distance readings. These sensors are particularly useful for detecting obstacles in the robot's immediate path that might be missed by the LiDAR, especially in blind spots.

Motors and Actuators

The motors control the robot's wheels and are responsible for its movement. PID (Proportional-Integral-Derivative) control is used to precisely control motor speed and direction, allowing the robot to follow its planned path smoothly. The actuators respond to control signals generated by the PID controller based on sensor inputs and navigation goals.

3. Simultaneous Localization and Mapping (SLAM)

SLAM is the process of creating a map of an unknown environment while simultaneously determining the robot's position within that map. In this project, SLAM is implemented using data from the LiDAR and IMU. As the robot moves through its environment, the LiDAR collects data on nearby obstacles, while the IMU helps stabilize the map by tracking the robot's movement. The ROS GMapping package is used to handle the SLAM process, and the map is continuously updated as the robot moves.



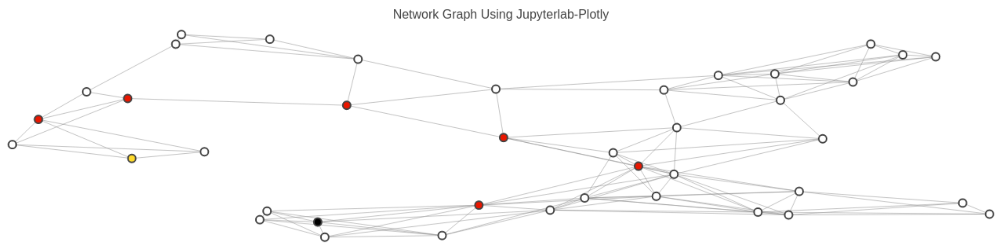
4. Path Planning with A* Algorithm

A* is an algorithm that searches for the shortest path between the robot's current location and a target destination. The robot uses the map generated by SLAM to identify obstacles and calculate a safe, efficient path to its goal. The A* algorithm takes into account both the cost of moving to each new location and the estimated distance to the goal, ensuring that the robot finds the most optimal route. The ROS Move Base package handles path planning and navigation, using A* to generate paths and avoid obstacles in real-time.

```
[11]: start = 35
goal = 33
test_path = PathPlanner(map_40, start, goal).path

print("Path: ", test_path)
show_map(map_40, start, goal, test_path)
```

```
Path:  [35, 15, 17, 12, 37, 16, 14, 33]
```

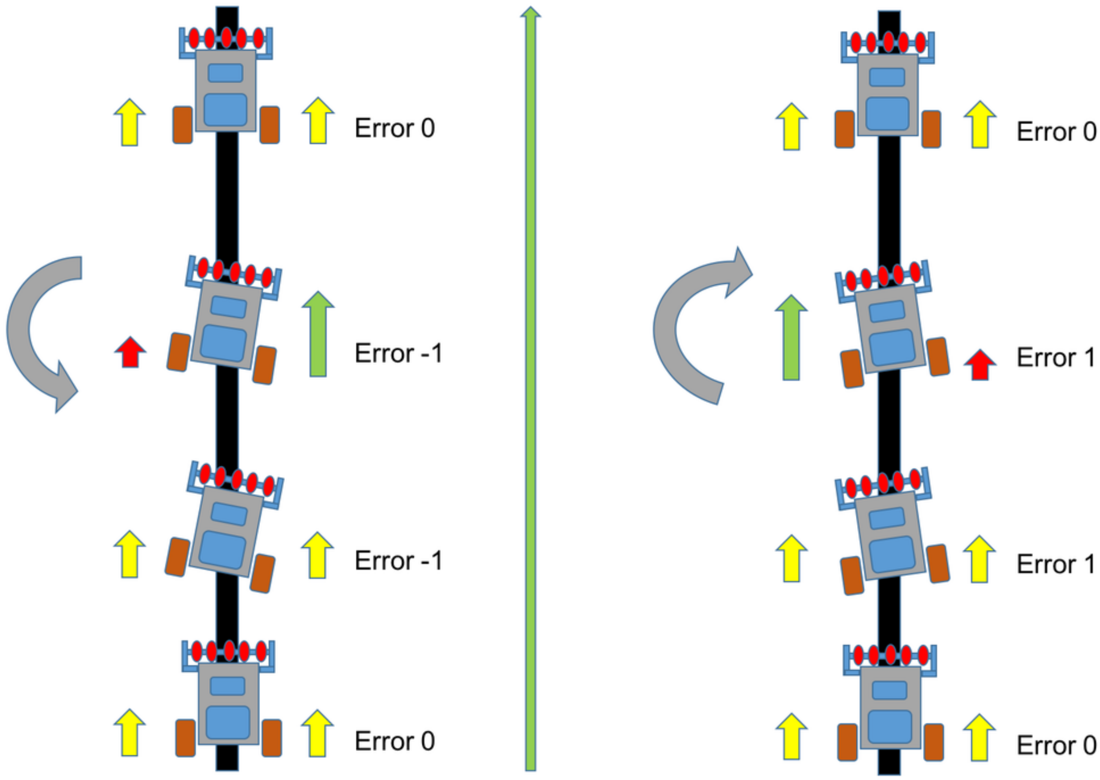


5. Real-Time Obstacle Avoidance

In addition to planning a global path with A*, the robot needs to react to obstacles that may not have been present when the map was first generated. This is where the ultrasonic sensors come into play. They continuously scan for nearby objects and provide real-time distance data to the robot. This data is fed into the local costmap, allowing the robot to make quick adjustments to its planned path and avoid collisions.

6. Robot Movement and PID Control

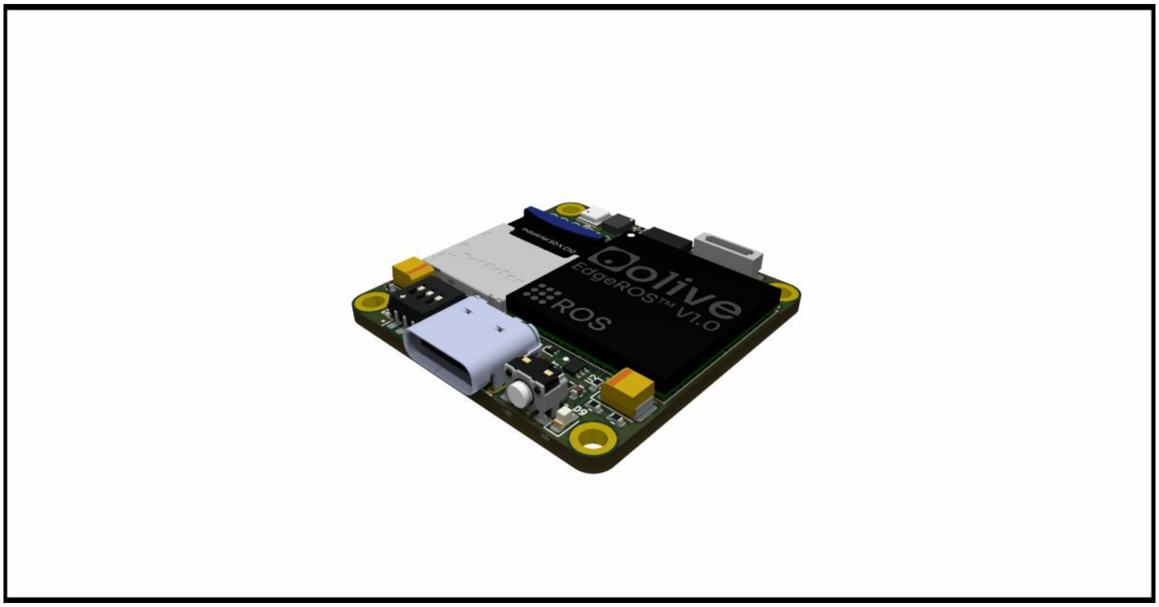
PID control ensures that the robot moves smoothly and accurately along its planned path. The PID controller takes input from the sensors (such as LiDAR and ultrasonic sensors), calculates the necessary adjustments, and sends control signals to the motors. By adjusting the motor speeds, the PID controller helps the robot maintain a steady course, whether it's navigating straight or making turns.



7. ROS Integration

ROS (Robot Operating System) acts as the middleware that ties all the components together. In this project, several key ROS packages are used:

• GMapping: Provides SLAM capabilities, allowing the robot to create a map and localize itself.
• Move Base: Manages path planning and obstacle avoidance, using A* for global path planning and real-time obstacle avoidance through sensor input.
• RViz: A visualization tool that allows real-time monitoring of the robot's map, position, and navigation goals.



8. Conclusion

In this robot navigation system, SLAM, A* path planning, ultrasonic sensors, and PID control are successfully integrated to enable autonomous navigation. The robot continuously builds a map of its environment, plans efficient paths to target locations, and dynamically avoids obstacles as they appear. ROS facilitates the integration of all these components, making the system highly flexible and scalable for future improvements.