ENPM690 Homework 3 Report – Spring 2025

Part 1: Robot Teleoperation

For the teleoperation mode, I deployed the robot in a simulation environment using **Gazebo with ROS 2**. The robot was controlled via keyboard inputs, allowing for manual navigation. The teleoperation implementation included:

- **Keyboard Input Capture:** The robot responded to real-time user inputs from the terminal.
- **Console Logging:** The terminal displayed each keypress along with the corresponding movement command.
- **Visual Feedback:** The simulation environment provided real-time text-based confirmation of movement actions.

Part 2: Autonomous Behavior

I implemented an **obstacle avoidance behavior** using sensor feedback from a **LIDAR sensor**. The robot dynamically adjusted its movement based on proximity to obstacles. The behavior included:

- **LIDAR Sensor Integration:** Real-time obstacle detection data was displayed in the simulation.
- **Algorithm Implementation:** The robot followed a simple **Bug algorithm**, modifying its path when an obstacle was detected.
- **Reactive Navigation:** If an obstacle was detected within **0.5 meters**, the robot performed a left turn to avoid collision; otherwise, it moved forward.

Tunable Parameter Analysis

The tunable parameter for this implementation was **speed**, which was tested at two values: **0.1 m/s** and **0.2 m/s**. The key observations were:

- **Speed = 0.1 m/s:** The robot moved cautiously, reacting more accurately to obstacles but taking longer to navigate the environment.
- **Speed = 0.2 m/s:** The robot covered more distance in less time but exhibited slightly delayed reactions to obstacles, leading to more frequent stops and direction adjustments.

Comparison: Teleoperation vs. Autonomous Behavior

- **Teleoperation:** Required manual control and provided **immediate responsiveness**, but lacked adaptability in complex environments.

- **Autonomous Mode:** Enabled self-navigation with **dynamic obstacle avoidance**, reducing the need for human intervention but requiring careful tuning for optimal performance.

Observations & Conclusion

- A lower speed enhanced precision but reduced efficiency.
- A higher speed improved traversal time but required fine-tuned reaction handling.
- Teleoperation was effective for direct control, while autonomy allowed for **adaptive navigation** in unpredictable environments.
- The tunable parameter provided flexibility in adjusting the behavior based on the environment.