Differential Drive Robot Navigation in Gazebo using ROS2

Team Members

Robert Hogan, Kevin Wu, Jacob Cassady, Shanshan Lin

Challenges and Reflections

1. Integrating Gazebo with ROS2

Challenge:
At first, the robot did not respond to velocity commands sent through ROS2 topics. The Gazebo simulation ran correctly, but no movement occurred when using teleop_twist_keyboard.

Solution:
We discovered that the issue was due to incorrect topic names between the Gazebo simulation and ROS2. After confirming the Gazebo-ROS bridge topics, we remapped /cmd_vel properly in both the launch file and the controller node. Once corrected, the robot moved as expected when receiving commands from ROS2.

2. PID Controller Instability

Challenge:
When we repurposed our PID controller from the midterm, the robot oscillated around the target point and sometimes overshot the goal due to aggressive tuning parameters.

Solution:
We performed systematic tuning of the PID gains. We reduced the proportional gain (Kp) to prevent overshooting, added a small derivative term (Kd) to dampen oscillations, and fine-tuned the integral term (Ki) to eliminate steady-state error. After several iterations, the robot achieved smooth and accurate navigation toward the goal point.

3. Launch File Configuration

Challenge:
Our initial launch file failed to start all necessary components together. Gazebo launched successfully, but the ROS2 bridge and PID node did not start automatically.

Solution:
We revised the launch file to include each process in separate Node and IncludeLaunchDescription blocks, ensuring all dependencies were loaded in the correct order. We also used relative package paths (via FindPackageShare) to improve portability. This allowed us to launch Gazebo, RViz, and the controller with a single command.

4. Bridging Gazebo Transforms
Challenge: Populating the /tf topic in ROS2 ecosystem was challenging. Our first attempt at leveraging the tf_topic parameter from gz-sim-diff-drive-system was unsuccessful. It did not

provide enough information for RViz to display the robot. Additionally, it was vulnerable to drift during collisions.

Solution: The PosePublisher and JointStatePublisher were leveraged to populate /tf. A static transform publisher was used to merge the gazebo robot chassis frame with the ros2 side. Remapping was required in the gazebo bridge to map from poses to tf. Additionally, the QoS flag for the /tf and /tf_static publishers had to be modified to be transient_local.