

MCE550 – Robotics & Intelligent Systems
ROS – Assignment 1
Spring 2023

In this assignment, you will create a ROS package that contains multiple nodes, a custom message, and a launch file. The package will be used to simulate a simple robot that moves in a straight line and detects obstacles.

Requirements

1. Create a new ROS package named **"simple_robot"** using **catkin_create_pkg**.
2. Create a new custom message type named **"Obstacle"** in the package. The message should have two fields: a **float32** named **"distance"** and a string named **"name"**.
3. Create a node named **"motion_controller"** in the package. This node should publish messages to the **"/cmd_vel"** topic to control the robot's motion. The node should subscribe to the **"/obstacle_detection"** topic to detect obstacles.
4. Create a node named **"obstacle_detector"** in the package. This node should subscribe to the **"/scan"** topic to detect obstacles and publish Obstacle messages to the **"/obstacle_detection"** topic.
5. Create a launch file named **"robot.launch"** in the package. This launch file should launch the **"motion_controller"** and **"obstacle_detector"** nodes.

After completing the above requirements you can test the **"obstacle_detector"** and **"motion_controller"** by launching a TurtleBot3 in an example world. The **"motion_controller"** will command robot to keep moving forward and the **"obstacle_detector"** should log the detected obstacles along the way. Use the following command to launch a TurtleBot3 robot in a non-empty world.

➤ **roslaunch turtlebot3_gazebo turtlebot3_world.launch**

You should get the below Gazebo environment shown in Fig. 1.

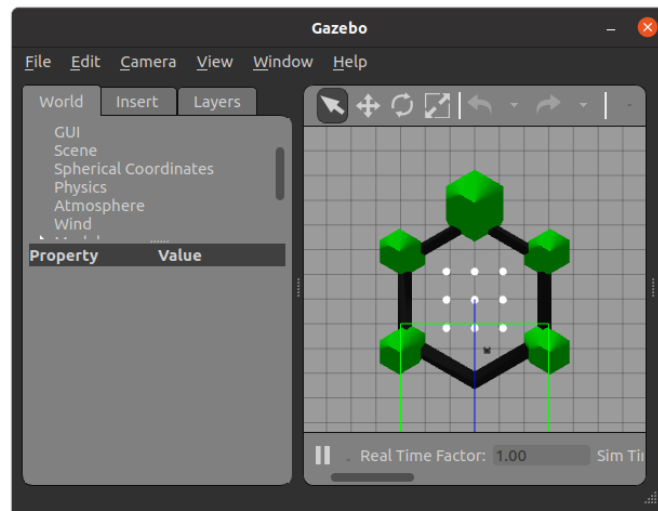


Figure 1: TurtleBot3 Sample World

Grading

Your assignment will be graded based on the following criteria:

- Proper use of ROS packages, nodes, and topics.
- Correct implementation of the custom "**Obstacle**" message type.
- Correct implementation of the "**motion_controller**" node to control the robot's motion and detect obstacles.
- Correct implementation of the "**obstacle_detector**" node to detect obstacles and publish Obstacle messages.
- Correct implementation of the "**robot.launch**" launch file to launch the necessary nodes.

Submission

Assignments are to be submitted **individually on Blackboard**. Submit a compressed folder containing the "**simple_robot**" package directory, along with a README file that includes instructions on how to build and run the package.

Due Date: Thursday, April 6, 2023.