Homework 5 **Student Name: _____**

AuE 8930: Computing and Simulation for Autonomy, Fall 2021

Instructor: Prof. Bing Li, Clemson University, Department of Automotive Engineering

* Refer to Syllabus for homework (late) submission, grading and plagiarism policies;

* Submission due 11/22/2021 11:59 pm via Canvas, include:

- This document (with answers), and with your program results/visualization;
- A .zip file of (modified) source code and data if any, which the TA might run.

Purpose

The goal of this homework is to get you started writing software that controls a (simulated) robot using ROS.

In this homework you will create a ROS node that will drive the robot around with a simple wanderer algorithm, much like a mobile robot. More specifically, the robot should move forward until it reaches an obstacle, then rotate in place until the way ahead is clear, then move forward again and repeat.

Preparation

Before you start this assignment, you should make sure that you understand the concepts in ROS tutorials 1-6, 8, 11 or 12, and 13. In addition, go over all the code samples from class and make sure you understand them thoroughly. Also make sure that the turtlebot_gazebo packages are installed on your (virtual) Ubuntu system.

Assignment

1. Make a new ROS package called wander_bot, with the appropriate dependencies.
2. Fill in your name and ID in the package manifest file.
3. Copy this provided launch file into a launch sub-directory of your package. This launch file runs the Gazebo simulator and the wander_bot node that you are going to write.
4. Your task is now to write the wander_bot node. The node should implement a simple algorithm: if the robot is moving sufficiently close to an obstacle in front of it, then rotate it in the direction that is away from obstacles until the way

ahead is clear; otherwise move forward by default. You can start from the Stopper node that we have developed in class.

5. Make sure that the wander_bot node works, by running it and watching the robot in the simulator.

The Rules

1. Make sure that your code is tidy and well-commented.
2. All parameter values (such as the rotation speed of the robot) should be configurable via the launch file.
3. All the work you turn in should be yours, and not done in collaboration with anyone else. If you use any external sources of inspiration, other than ros.org, then let us know in a README file.

What to Hand In

You should hand in everything that someone else needs to run your code. For this assignment, that means your source code (either Python or C++), manifest file, CMakeLists.txt (if you're using C++), and launch files. You should *not* hand in executable files, or any other files that can be regenerated.

Your code should be easy to run. After getting a copy of your code, running:

```
catkin-make --pkg wander_bot
roslaunch wander_bot wander_bot.launch
```

should be sufficient to start up Gazebo and make the robot move.

Finally, zip all the necessary files and submit the zip file to Canvas.