CSCE 574

Section 001

Fall 2017

Project 1

Ian Davis

**Description**

The most notable aspect of my project is that I run all three robots but each take turns going through the scan-drive-turn sequence. I got the robots moving by subscribing to two topics and publishing to one. I subscribed to the topics in “nav\_msgs/Odometry to update my robots’ positions in the world and in sensor\_msgs/LaserScan to take readings of the robot’s laser sensor. With these two packages and their respective topics I was able keep the robots’ position in relation to the walls and ground thus, accurately guide the robots around. In regards to the odometry callback function, the way I found to get the theta in addition to the x and y positions was to take the odom topic and turn its pose message from a quaternion into an RPY allowing me to get the yaw (theta angle). The way I made the robots move was to publish to their respective geometry\_msgs/Twist topics. I was able to input and adjust their velocities in both linear and angular fashion. One odd thing I was noticing was that my laser\_dist variable wouldn’t be updated till I already started driving and so in addition to having the normal subscription to the laser sensor I also used the waitForMessage() function to get an initial scan for each robot. By using that I just grabbed a single message and go on my way like normal. I also put the driving and turn processes in their own functions. The driving function simply takes the laser reading and drives forward while checking the distance between itself and the distance it needs to travel. The turning function works similarly but replaces the distance it needs to move with the angle it needs to reach. I was able to find equations online for finding the euclidean distance as well as the normalized angle. Both the drive and turn functions publish a 0 velocity message at the end to stop the robot. Each robot has separate driving and turning functions since the odometry data they need is specific to them. I have a loop in the main function that runs the scan-drive-turn sequence 4 times for each robot then ends as they have done each edge. Lastly I shutdown the node as it has finished it’s job.

**Evaluation**

As of writing this report all three robots go through through the sequence without crashing or freezing. The only aspect which I don’t think works is that the laser scan will never give a reading larger than 5 back. I don’t know why this happens but I never works no matter the robot’s position. The scans still give smaller readings as you get closer to the wall but when a wall is farther than 5 meters away it won’t give a reading larger than that. I tried using different positions in the LaserScan msg->ranges array as well as the max\_range and min\_range but nothing will go higher than 5. Because of this issue my robots won’t go down the full extent of the boxes in some cases. Only other problem I had was that I couldn’t figure out a way to make my node work without using the 12 global sensor variables for the robots. Excluding these problems my robots appear to be working correctly.

**Allocation of Effort**

I wrote my code and implemented it as this was a solo project.