Lab 1

Introduction to ROS

Objective

In this experiment, students will use ROS to achieve following goals:

1. Understand the basics of ROS
2. Implement simple nodes to perform publish/subscribe messages
3. Use the Turtlebot-Stage tool to simulate the iRobot Roomba.

Reference Materials


Lab Procedure

This lab consists of the following three parts:

Part 1: Publish and Subscribe a string message

1. Create a package for a simple publish/subscribe task.
2. Create two C/C++ nodes and name them as talker.cpp and listener.cpp. The former publishes a string message “Hello ROS” with a message id number. The latter program subscribe the published message and print on the PC screen.

Part 2: Use Turtlebot and Stage simulator to run a demo program

With the next two steps, you will learn how to use a simulator to test your programs.

1. Open a terminal and run “roscore”.
2. Open another tap in the existing terminal and start Stage simulator by executing a command: “roslaunch turtlebot_stage. turtlebot_in_stage.launch”. You might want to comment out the last line of “turtlebot_in_stage.launch” file since we are not using rviz in this experiment.
3. Confirm that a 2D simulator with a simple map and a box shaped robot in it.
4. Open another tap and run “roslaunch turtlebot_teleop keyboard_teleop.launch –screen”
5. Use keyboard to control the robot.
Part 3: Create a simple node to control the turtlebot in Stage simulator

You might want to look at turtlebot_joy.cpp before starting your programming.

1. Create a node to control turtlebot: Make turtlebot to circle around a small region.

2. Close all terminals. Repeat 2-1, 2-2, and 2-3 in order to have a simulator ready.

3. Execute your node.

Lab grading

Your grade is based upon three criteria: 1) Pre-lab, 2) Program test, and 3) Post-lab report with emphasis upon the performance of your program and post-lab report. For this reason, your program must be checked by TA before submitting your report.