Lab 1a  
Introduction to ROS and TurtleBot 2

OVERVIEW

In this lab, you will learn about the Open Source Robotics Foundation ROS software, and get to know your TurtleBot 2 robot.

TASKS

Computer setup. This semester, your group will be assigned one of our TurtleBot 2 robots to work with. Each robot has a laptop associated with it, with the robot name labeled on the lid. From time to time, it will be useful to have another computer to work with while you are completing the labs, in order to run software remotely on the robots. Your personal laptops will probably suffice (but you won’t need them today).

Robots will be shared between two teams. Your group should choose a login password, and share it with me. I will let you know which robot to use, and also whether to use the “team1” or “team2” login account on your robot. Once you supply me with your login password, I’ll get you up and running with the computer.

You will be using the Linux command line extensively when operating your TurtleBot, so you might want to familiarize yourself with it before the next lab. One tutorial is at http://www.ee.surrey.ac.uk/Teaching/Unix/. We will be using the terminator software to access the command line. The icon (on the left of your laptop’s screen) looks like this:

As the icon suggests, terminator conveniently allows you to split its window into a bunch of tiled terminal sessions. The shortcut keys to do this are Ctrl + Shift + O to split the current pane vertically, and Ctrl + Shift + E to split it horizontally. When the tutorials below ask you to open a new window, consider just splitting the current one instead.

Other tips: you can often use tab completion in terminal windows by typing the first few letters of a command or a command line argument and then hitting tab to automatically complete the rest. Also, the robot laptops screens are touch-enabled.

ROS Tutorials. Load the ROS tutorial page at http://www.ros.org/wiki/ROS/Tutorials. Along with your group, and taking time to make sure that everyone is following what is going on, complete at
least the following tutorials:

- 2. Navigating the ROS Filesystem
- 3. Creating a ROS Package *use catkin, not rosbuild here*
- 5. Understanding ROS Nodes
- 6. Understanding ROS Topics
- 7. Understanding ROS Services and Parameters
- 8. Using rqt_console and roslaunch
- 12. Writing a Simple Publisher and Subscriber (Python)
- 13. Examining the Simple Publisher and Subscriber

For tutorial 2, part 3, you will not need to follow the “Creating a workspace for catkin” tutorial, since your account was already set up with a workspace.

It’s probably a good idea to take turns reading over each tutorial, performing the steps, and explaining the material to your group members. *There will be an in-class quiz on the material in the tutorials*, so make sure that everyone understands what’s going on!

**Modify the simple publisher and subscriber.** In the same directory you created for tutorial 12, make a new program, *reverser*, that subscribes to messages on the `chatter` topic, and publishes the same message, backwards, on a topic named `rchatter`. For instance, if the `talker` program publishes the message

```
hello world 123
```

the *reverser* program should publish the message

```
321 dlrow olleh
```

You should be able to verify that the `reverser` program is working by running

```
rostopic echo /rchatter
```

**Get to know your TurtleBot.** Once you’re familiar with ROS, you should get to know your TurtleBot. Make sure the robot is powered up, make sure the robot and Kinect are plugged into the USB ports, and then start the robot control software by typing

```
roslaunch turtlebot_bringup minimal.launch
```

You should hear a rising series of beeps from the robot to indicate that it is connected and ready to go. In a separate terminal, run
roslaunch turtlebot_bringup 3dsensor.launch

You may see some warnings about missing calibration files – that’s fine. Finally, in a third terminal, run

roslaunch turtlebot_rviz_launchers view_robot.launch

If the rviz program crashes (this sometimes happens, sadly), just try the command again. When it launches, you will see the rviz interface. Maximize the window, and then try bringing up the “Image” and “Registered DepthCloud” displays by clicking the check boxes on the left. You might need to wait a second for either to appear. Finally, you might want to rotate the 3D display (by clicking and dragging) to see what’s going on with the DepthCloud.

Finally, close the rviz window (pick the “Close without saving” option if prompted whether to save the settings), and go back to the terminal and run

roslaunch turtlebot_teleop keyboard_teleop.launch

Gently place the robot on the floor, with the laptop on top of it and the lid still open. Being careful to catch the laptop if it falls, try using the keyboard to control the robot and drive it around for a bit. Please do not increase the speeds from the default.

To safely shut down your robot, hit Ctrl+C in terminal windows with running ROS processes, in the reverse order from which you started them (i.e. kill the minimal.launch process last). When the robot control software is shut down, you will hear a series of falling beeps from the robot to let you know it’s ready to be safely powered off.

WHAT TO TURN IN

You should schedule a demonstration with me to show:

- completion of the ROS tutorials, and knowledge of their contents
- your finished reverser node