COMS 4733, Computational Aspects of Robotics, Fall 2019

Due (electronically via courseworks):   Monday, Sep. 23, 11:59 PM

Lab 1:   Getting Started

The purpose of this lab is to make sure you 1) can get a running ROS Indigo system up and 2) learn how to rotate and translate a mobile robot base.

1. Read chapters 1-7 of the book ROS By Example, A Do-It-Yourself Guide to the Robot Operating System, by R. Patrick Goebel. Make sure you are using the version of the book for ROS Indigo – this is important!
2. You need to set up a working Ubuntu 14.04 machine, either using a Virtual Machine or direct boot into your personal machine.   Your best option is to create a native Ubuntu 14.04 USB boot stick and run a native Ubuntu machine. This video describes how – just change Ubuntu 16.04 in the video with 14.04.   Here ar links to Virtual Box and Ubuntu:
    a. VirtualBox
    b. Ubuntu 14.04.5 LTS (Trusty Tahr)
3. Also, Instructions on doing this can be found in chapter 3 of the book.
4. If using Mac, you can purchase parallels software that will give you access to a Ubuntu 14.04 image. This video is helpful and explains what to do to get Ubuntu 14.04 set up. Also, some notes on the setup.
5. **If neither of these options work for you, then talk to us about using other computing resources.**
6. Next, install ROS Indigo. Chapter 4 explains how to install ROS Indigo.
7. Chapter 5 explains how to install the Ros By Example book code.
8. Chapter 6 explains how to install the Arbotix python simulator.
9. You should go through the official ROS tutorials , Beginner Level, to acquaint yourself with ROS. Make sure you actually run the tutorial examples on your machine.
10. Modify the code in timed_out_and_back.py (see section 7.6 of the book) to control the robot from a terminal. Your program should do the following:
    - Set up a turtlebot environment using **roslaunch rbx1_bringup fake_turtlebot.launch**
    - Start an rviz simulator: **rosrun rviz rviz -d `rospack find rbx1_nav`/sim.rviz**
    - Prompt the user to input a **T** for translation, **R** for rotation, or **Q** for quit.
    - If **T** or **R,** prompt for the distance to translate or angles to rotate. Make sure your program handles positive and negative translations (forward and back movement) and positive and negative rotations (counterclockwise and clockwise rotation).
    - Keep prompting until the user hits **Q**
11. What to submit: Instructions for submitting lab 1 are here.