

CS/ME/ECE/AE/BME 7785

Lab 0

Before you get started...

Read this entire section before starting any installations to understand all of your options. Always feel free to reach out to the instructors of the course if you have concerns or questions.

When released, the robots we work with in this course (Turtlebot 3), originally used Ubuntu 16.04 and ROS 1 Kinetic Kame but, since those are reaching their EOL, we will be switching to Ubuntu 20.04 and ROS 2 Foxy Fitzroy. Be careful when following Turtlebot3 guides blindly as they were originally written for ROS 1 Kinetic Kame.

Ubuntu is the standard operating system used with ROS, and the one we advise you use in this course. We will be using **Ubuntu 20.04**. If you don't have this Ubuntu version installed already, then you have several options:

- dual-boot your laptop to run both Ubuntu and your current OS.
- install Ubuntu on an external hard drive and boot from that.
- set up Ubuntu on a virtual machine. There will be significant challenges to making this work well. For MacOS some have reported good results using [Parallels](#), but again, use this approach with care, our two GUI tools (rviz and Gazebo) may not work well.
- use one of the lab desktops we provide in CCB 050. This is the least flexible solution since it relies on a shared machine.

Note 1: ROS 2 has functionality on Windows, however, the instructors have not tested this thoroughly with the Turtlebots and it does not appear that the Turtlebot3 packages have been updated for ROS 2 with Windows.

Note 2: ROS 2 has functionality on MAC OS, however, the instructors have not tested this thoroughly with the Turtlebots. Beyond this, there may be issues with the M1/M2 chipset and it does not appear that the Turtlebot3 packages have been updated for ROS 2 with Mac OS.

Lab

1. If you are new to ROS and to a lesser extent ROS2, information for installing and understanding ROS2 Foxy Fitzroy can be found at: <http://wiki.ros.org/ROS/Introduction>.
2. There are two guides to install ROS2 on Linux and Windows (they are the same presented in different ways),

- (a) Follow the guide from the official ROS 2 documentation at <http://docs.ros.org/en/foxy/Installation.html>
 - i. This course recommends you perform a Desktop-Full Install so you include all the GUI options available with ROS 2.
- (b) Follow the guide from the Turtlebot3 documentation at <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>
 - i. **Note this page defaults to ROS Kinetic, make sure you click the button at the top for Foxy/Windows and follow the correct Foxy/Windows directions.** You'll know you're on the right page if your instructions include commands like,

```
sudo apt-get install ros-foxy-gazebo-*
```
3. Complete the Beginner ROS tutorials found at <http://docs.ros.org/en/foxy/Tutorials.html>. Make sure you are using the **colcon** build system. There are tutorials in both C++ and Python. Feel free to use either language, however note that ROS Python code is typically easier to understand and debug (for your instructors as well as in general). C++ is typically performs faster and has some advantages (like Nodelets), but we will not be doing anything requiring that kind of optimization.
4. Next install the Turtlebot3 Packages. The directions are in section 1.4 of <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>. **Note this page defaults to ROS Kinetic, make sure you click the button at the top for Foxy/Windows and follow the correct Foxy/Windows directions.** You'll know you're on the right page if your instructions include commands like,

```
sudo apt-get install ros-foxy-turtlebot3-msgs
```

The next two steps involve simulation. If your laptop cannot handle the simulation environment or there are other errors let us know.
5. Now that you're starting to grasp at ROS, it is useful to be able to test scripts you write in a simulation environment before deploying them on the hardware in the lab. The people working on the Turtlebot3 have released a great simulation package for Gazebo (<http://gazebosim.org/>). Follow the tutorials in Step 6 of the quick start guide found at <https://emanual.robotis.com/docs/en/platform/turtlebot3/simulation/> make sure to install and test the Gazebo simulator (step 6.1) and *fake_node* (step 6.4). **Note again these pages defaults to ROS Kinetic, make sure you click the button at the top for Noetic and follow the correct Noetic directions.**
6. Make sure you can run the *turtlebot3_teleop* example using the Turtlebot3 fake node and Gazebo simulator. The simulated robot should move around based on your keyboard command!

Submission

There is nothing to submit for this lab, but please post any issues or questions you run into during this process in Slack. While this lab does not count towards your grade, it is the foundation for the rest of the course. If you are having problems, please avoid waiting until Lab 1 to address them.

Note, after reading over the beginner tutorials for ROS you should be able to answer the following questions.

- What is a ROS_DOMAIN_ID?
- What is a node?
- What is a topic?
- What is a message?
- What is a subscriber?
- Write the syntax to create a subscriber in Python.
- What is a publisher?
- Write the syntax to create a publisher in Python.
- Can a node have multiple subscribers? Can a node have multiple publishers?