RBE 502 — ROBOT CONTROL

Instructor: Siavash Farzan

Fall 2022

Programming Assignment 0:

Setting Up Robot Operating System (ROS) and MATLAB ROS Toolbox

Note: This assignment must be completed before the start of Programming Assignment 1 as you may not be able to begin work on Assignment 1 if your Linux machine and MATLAB are not properly set up.

0.1 Objectives

By the end of this assignment, you should have Robot Operating System (ROS) Noetic Ninjemys and MATLAB installed on a Linux Ubuntu 20.04 machine. Ubuntu could be installed as a virtual machine if needed.

0.2 Overview

For some of the Programming Assignments in this course we will be using a Linux Ubuntu machine because Robot Operating System (ROS) only works on Linux machines. If you currently have a computer with the Ubuntu 20.04 operating system, or you can dual boot with Ubuntu 20.04 operating system, that's great! This is the best way to work on robots. In that case, you can skip the next two sections of this assignment and continue from Section 0.5.

However, if you have only a Windows or macOS operating system, you have two options:

- Install a virtual machine on your Windows/macOS. This may be a little slow while performing heavy simulation tasks in Gazebo, but will still do the job. (Gazebo is a robot simulator that we are going to use throughout the course, together with ROS.)

- Dual installation (e.g. installing Windows/macOS and Linux alongside each other). There are many resources on the internet on how to do this. (Please be careful about dual boot, since you have to take potential risks.)

In the following, we will go through some basic steps to get our development environment ready.

0.3 Download the Virtual Machine

Virtual machines (VMs) allow users to run an operating system that behaves like a completely separate computer in an app window on a desktop. One or more virtual "guest" machines run

on a physical "host" machine. Each virtual machine runs its own operating system and functions separately from the other VMs, even when they are all running on the same hosts*. In this course, you can use a VM to run an instance of Ubuntu 20.04 in order to use ROS even if you don't have a Linux computer.

When you create a virtual machine, you can shutdown and start the same VM with the same files still stored on it; however, if you delete a VM, you also delete any files you have stored on it.

If you have a Windows laptop, you can download and install VMware Workstation Player, available at:
`https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html`.
Note that VMware Workstation 16 Player is available for free to students for non-commercial use.

If you have a Mac laptop, you can go to VMware Fusion webpage: register under "Use for Free with a Personal Use License" tab and download VMware Fusion Player using a free personal license:
`https://www.vmware.com/products/fusion.html`
For macOS you can use VMware Fusion 12.1.2 or higher.

0.4   Install Linux on VMware

Once you have your VMware installed, let's create a new VM and install Ubuntu 20.04. First, download Ubuntu 20.04 disc image from official website (64-bit PC (AMD64) desktop image):
`http://releases.ubuntu.com/20.04/`
You should now have a `.iso` file.

For Windows:

- Open VMware, and select "Create a New Virtual Machine"
    - Choose the `.iso` disc image you downloaded as the "Installer disc image file"
    - Enter some information about this Linux, including Name, User name, and Password
    - Enter a Virtual machine name, e.g. "Ubuntu 20.04"
    - Allocate at least 30GB (preferred 50GB or more) for the Maximum disk size
    - Store virtual disk as a "single" file
    - Under the "Customize Hardware" settings: use the slider to allocate at least 5GB of RAM (i.e. 5120 MB) for better performance
    - Select "Finish"

For macOS:

i) Open VMware Fusion, and select the "Install from disc or image" option.
    - Select "Continue", and use the Ubuntu `.iso` file you downloaded above as the disc image.
    - Select "Continue", and use the "easy install" set up with the default settings. Enter a password for your Ubuntu user account, and select "Continue" again.

---

*`https://www.vmware.com/topics/glossary/content/virtual-machine`

– Select "Finish" to launch the virtual machine.

It will take a few minutes for the Ubuntu 20.04 to be installed on the virtual machine. Once installed, log in to your user account using the password you specified above.

ii) Now shut down the virtual machine (shut down, not pause. Please be sure you are aware of the difference).

iii) Next we want to allocate an appropriate amount of RAM to the VM. With the Ubuntu shut down, from the VMware menu, go to Virtual Machine → Settings...

– Click on the "Processors & Memory" icon.
– Under the "Memory" section, use the slider to allocate at least 5GB of RAM (i.e. 5120 MB). You should be able to allocate 5GB of RAM in 8GB systems. On a 16GB machine, you could easily allocate over 8GB (5GB should be sufficient for our course assignments). You will not be able to edit this if your VMware is not shut down.

iv) Finally, we want to allocate at least 30GB (preferred 50GB or more) of Hard Disk to Ubuntu. Under the Settings:

– Click on the "Hard Disc" icon.
– Use the slider to increase the Disk size to 30GB or more.

Note that the disk size 30GB will not be allocated instantly, but will grow gradually as you add more stuff.
Close the Settings window.

Great. Now you have a (virtual) Linux Ubuntu computer. Take your time and get familiar with it!

0.5   Install ROS

Once you are familiar with Linux, you can start installing Robot Operating System (ROS). In this class, we will be using ROS version Noetic. You will learn more about ROS in the next assignment.

First, in Ubuntu, open up a terminal window (right click anywhere → Open Terminal) and in it run

```
echo "export SVGA_VGPU10=0" >> ~/.profile
```

In general, to install ROS, we need to follow ROS installation tutorial:
http://wiki.ros.org/noetic/Installation/Ubuntu

The main steps are:

i) Set up your computer to accept software from packages.ros.org
(**note: the commands here are supposed to be one line, with no break in between. Be careful when copying and pasting into the terminal.**)

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc)
    main" > /etc/apt/sources.list.d/ros-latest.list'
```

ii) Set up your keys

```
sudo apt install curl
curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc | sudo
    apt-key add -
```

iii) Update package index

```
sudo apt update
```

iv) Install ROS desktop full

```
sudo apt install ros-noetic-desktop-full
```

v) Make sure your environment variables get correctly setup every time a new shell is launched by adding to your `.bashrc` file. Run the following commands

```
echo "source /opt/ros/noetic/setup.bash" >> ~/.bashrc
source ~/.bashrc
```

vi) Install more dependencies for building packages

```
sudo apt install python3-rosdep python3-rosinstall python3-rosinstall-
    generator python3-wstool build-essential
```

vii) Initialize rosdep

```
sudo rosdep init
rosdep update
```

0.6   Create a ROS Workspace

From now on, we assume that you have already installed Ubuntu 20.04 and ROS Noetic.

Open a new terminal, and create a new ROS workspace by the following commands (run them line by line).

```
mkdir -p ~/rbe502_ros/src
cd ~/rbe502_ros
catkin_make
echo "source ~/rbe502_ros/devel/setup.bash" >> ~/.bashrc
source ~/rbe502_ros/devel/setup.bash
```

We will use the RRBOT robot for some of our assignments. To set up the RRBOT robot in Gazebo, first let's download the ROS package for the robot.

```
cd ~/rbe502_ros/src
git clone https://github.com/ros-simulation/gazebo_ros_demos.git
```

We can install the dependencies by the following commands.

```
sudo apt-get update
sudo apt-get upgrade
sudo apt update
```

```
sudo apt-get install ros-noetic-ros-control ros-noetic-ros-controllers
sudo apt-get install ros-noetic-gazebo-ros-pkgs ros-noetic-gazebo-ros-control
```

With all dependencies ready, we can build the ROS package by the following commands.

```
cd ~/rbe502_ros
catkin_make
```

To visualize the robot in Gazebo, you can run the following command in the Ubuntu terminal:

```
roslaunch rrbot_gazebo rrbot_world.launch
```

0.7   Common VMware Issues and Solutions

- If you get the error below every time you boot with VMware:
  `'VMware Cannot Connect the Virtual Device sata0:1'`
  Go to your VMware settings: in the Hardware tab choose CD/DVD, and in Device Status, uncheck the option "Connect at power on".

- (For Mac computers) if you get the error below when booting your virtual machine for the first time:
  `'Could not open /dev/vmmon: Broken pipe.'`
  On your computer (not in your VM) go to Systems Preferences → Security & Privacy → General. Under the section "Allow apps downloaded from", press allow for VMware. Then restart VMware.

- If Gazebo is repeatedly crashing and shutting down, try shutting down the VM and with the VM still shut down, go to Virtual Machine → Settings → Display and uncheck "Accelerate 3D Graphics". This may cause Gazebo to have more lag but it should also stop it from crashing.

- If you cannot connect to the internet, click the wifi icon at the top right and make sure that "Enable Networking" is checked. If it is already checked, try unchecking and rechecking it. When you are connected, the VM should recognize your connection as an ethernet connection and the wifi icon should be replaced with two arrows pointing in opposite directions (one up, one down). If it still will not connect, try restarting your VM.

0.8   Install MATLAB

Download and install the latest version of MATLAB on your Ubuntu machine by following the instructions available at:
`https://hub.wpi.edu/article/805/matlab-install-guide`

After you downloaded the installer file in Ubuntu, unzip the installer file from the terminal using the command:

```
sudo unzip -X -K matlab_R2022a_glnxa64.zip -d matlab_2022a_installer
```

Then, open a terminal and go to the `matalab_2022a_installer` directory and install MATLAB using the commands:

```
xhost +SI:localuser:root
sudo ./install
```

Now the MATLAB Installer is on, sign in, follow the steps and choose the product that you need. Make sure to select at least the following products to be installed:

- MATLAB

- Control System Toolbox

- Robotics System Toolbox

- ROS Toolbox

- Symbolic Math Toolbox

Also make sure to select the "Create symbolic links to MATLAB scripts in" option during the installation process.

After MATLAB is installed on your Ubuntu machine, you can start MATLAB by typing the command below in a terminal.

```
matlab
```

0.9 Learn from ROS and MATLAB Tutorials

Once you have ROS Noetic and MATLAB installed, you can follow the tutorials on ROS Wiki as well as the MATLAB ROS Toolbox documentation:

```
http://wiki.ros.org/ROS/Tutorials
https://www.mathworks.com/help/ros/
```

If you have no prior MATLAB experience, you can take the MATLAB Onramp course (about 2 hours) at:
```
https://www.mathworks.com/learn/tutorials/matlab-onramp.html
```

You will need to use your WPI email to create a Mathworks account in order to access the course.

Have fun!

0.10 Submission

- Submission: individual submission via Gradescope. Submit a screenshot of the RRBOT robot in the Gazebo window with MATLAB open in the background.

- Due: as specified on Canvas.

- Files to submit: (please use the exact file name, file type could be .png, .jpg, or .pdf)

    - LastName_ProgAssignment0_screenshot.png