Instructions on Building Robot and Running Monte Carlo Localization Assignment

Zuozhi Yang and Todd W. Neller

Gettysburg College, Gettysburg Pennsylvania 17325

This is the full instruction on building and setting up the robot model for our Monte Carlo Localization (MCL) assignment.

1 Hardware

The robot model is originally designed by Dr. Paul Ruvolo at Olin's College. We slightly adapted it so that the connection is directly between the robot and the computer instead of the sharing Wi-Fi connection.

Neato XV Vacuum and Mini USB Cable The Neato XV Vacuum is the main body of the robot with actuator and sensor (laser range finder). We uses StarTech Mini USB Cable - A to Right Angle Mini B. We remove the vacuum brush and, in newer models, open the dirt bin and toggle the ON/OFF switch to ON.

Raspberry Pi 2 Model B and Memory Card We install a Raspberry Pi on the robot to host an wireless connection to allow communication between robot and Linux machine. A SanDisk Ultra 16GB Memory Card is also needed by Raspberry Pi to install operating system. We cloned our current install http://cs.gettysburg.edu/ tneller/cs371/mcl/pisdcard.img as follows:

- Read the instructions for cloning a Raspberry Pi SD Card from http://lifehacker.com/how-to-clone-your-raspberry-pi-sd-card-for-super-easy-r-1261113524.
- Install win32diskimager from https://sourceforge.net/projects/win32diskimager/.
- Read image from our current Raspberry Pi SD card using win32diskimager. (link above)
- Format new card.
- Write image to our new Raspberry Pi SD card using win32diskimager.

Adafruit Industries - 1115 - 16*2 LCD and Keypad Kit The Adafruit LCD will indicate IP address of the wireless connection hosted by Raspberry Pi. A tutorial on assembling it is https://learn.adafruit.com/adafruit-16x2-character-lcd-plus-keypad- for-raspberry-pi/assembly

Two CanaKit Raspberry Pi WiFi Wireless Adapters / Dongles (802.11 n/g/b 150 Mbps) Two wireless adapters are used for communication. One is installed on the Linux machine, and the other is installed on the Raspberry Pi.

External Battery for Raspberry Pi We used a 6000 mAh 5V/2.1A USB external battery pack.

2 Software

On the Raspberry Pi, we use a Linux operation system. It can be downloaded here. (This is if you cannot use the cloned Raspberry Pi Linux install from the previous steps.) To clone the image, one can use the instructions here: http://elinux.org/RPi_Easy_SD_Card_Setup For setup of a Raspberry Pi, you'll need a USB keyboard, and an HDMI display with an HDMI cable to plug into the Pi.

Here are some configuration need to be done:

1. Plug in the USB wireless adapter and boot the system. Login using the default username `pi` and password `raspberry`.
2. Use `ifcongig -a` to check and record the MAC address of the adapter. Your advice is called `wlan0` in the system.
3. Modify line 7 of `~/get_pi_specific.py`, change the key of the first entry in variable `data` to the MAC address of your adapter. You can also change the name of this connection by changing `rpi-adhoc-222`.
4. If you have multiple robots, you can give them different IP address by changing line 9 of `~/make_interfaces_adhoc.sh`

Also, we need Robot Operating System (ROS) installed on student's machine. Dr. Ruvolo has a very detailed instruction for installing ROS[1].

3 Assignment Base Code

First, one needs to create a workspace. Catkin is the official build system of ROS, so for here we call it `catkin_ws`. For more details, look at

`http://wiki.ros.org/catkin/Tutorials/create_a_workspace`

```
$ source /opt/ros/indigo/setup.bash
$ mkdir -p ~/catkin_ws/src
$ cd ~/catkin_ws/src
$ catkin_init_workspace
$ cd ..
$ catkin_make
```

Edit your /.bashrc file so that the following line appears last in that file:

```
$ source ~/catkin_ws/devel/setup.bash
```

Then copy all the files in the supplied zip file into the `src` directory under workspace.

[1] https://sites.google.com/site/comprobo15/how-to/setting-up-your-environment

4 Connecting with the Robot

This section gives running down list of how to boot the robot and connect the robot with student machine with the wireless connection.

- Before the use of the robot, you should make sure that both the robot and the battery for Raspberry Pi is sufficiently charged for using.
- Make sure every cable is properly plugged except the Raspberry Pi battery cable.
- Plug in the battery for Raspberry Pi and it should take about one minute to boot. When you see the LCD screen is flashing on and off, the Raspberry Pi is successfully booted. Then turn on the Neato vacuum and the LCD should be on.
- There are four button on the keypad forming a diamond shape. Press the `down` button until `ad-hoc` shows on the LCD screen, the press right. After a short time of configuring, an IP address should be on the LCD screen. The IP address is customizable through the OS on the Raspberry Pi.
- On the student Linux machine, connect to the wireless connection hosted by the Raspberry Pi. The host name should start with `rpi-`

5 Controlling the Robot and Running the Assignment

This section gives running down list of how to start the ROS, control the robot, and run particle filter algorithm.

- `roscore`
  Bring up ROS core services. (optional, roslaunch will bring up its own)
- `rosrun teleop_twist_keyboard teleop_twist_keyboard.py`
  Bring up keyboard control, instruction are printed out at initiation (needs new terminal)
- `roslaunch turtlebot_rviz_launchers view_navigation.launch`
  Bring up visualization. (needs new terminal)
- Build a map if you do not have one. See commands below.
- `roslaunch my_pf test_my_pf.launch map_file:=MAP_DIRECTORY`
  Launch Particle Filter using the given `.yaml` map (needs new terminal)

Here is how to build a map using ROS:

- `roslaunch neato_2dnav gmapping_demo.launch`
  Start SLAM algorithm. (needs new terminal)
- Drive the robot around until you have a satisfying map
- `rosrun map_server map_saver -f ~/MAP_SAVING_DIRECTORY`
  Save the map (needs new terminal)
- Terminate mapping algorithm.
- (Optional) If you have a noisy map, you can also use image editing software to edit out some noise manually.

Note that whenever you want to drive the robot around, you should put the controlling terminal at the very top.