Advanced Robotics

Introduction to ROS

Pôle

Numérique

Author	: Guillaume GIBERT
Version	: 1 R 0
Promotion	: 2026
Program	: EENG4
Date	: October 8th 2024

1. Lab description

The aim of this lab session is to apply the ROS knowledge introduced during last week's lectures and tutorials. The students will experiment using the mobile robot Turtlebot that is fully ROS-compatible. This lab session is composed of different phases:

- set up the hardware components and the network;
- launch the compulsory ROS components;
- find various information on ROS nodes and topics;
- create a map of the environment using SLAM and teleoperation;
- perform navigation in the known environment thanks to the map.

2. Setup

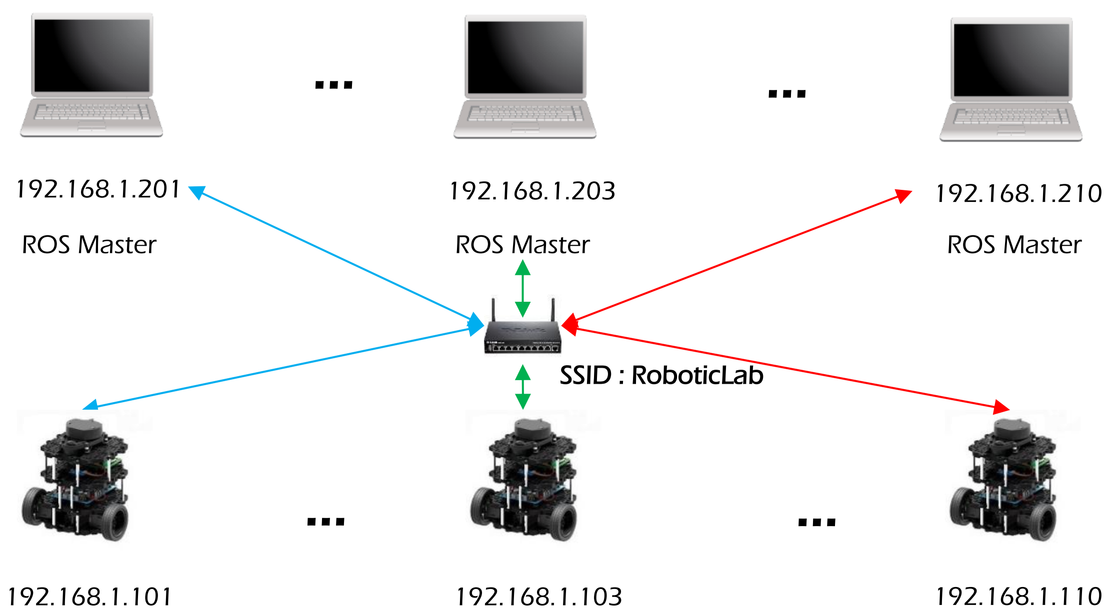

Figure 1 : Architecture of the connections of the laptops and robots in the local network

2.1. Network settings

One Turtlebot and one laptop are available for each pair of students. They are all connected to a local network whose SSID is **RoboticLab**.

Each laptop will run a ROS Master and each Turtlebot will connect to its dedicated ROS Master as represented in Figure 1.

2.1.1. Determine network settings

First, you need to check if there is no network issue. To do so, open a terminal (on the PC) and check the IP address of the PC by typing:

```
ifconfig
```

<u>Is the PC connected to the right network?</u>

#ANSWER

<u>Is the IP address the correct one?</u>

#ANSWER

2.1.2. Check network connectivity

On the PC, open a terminal and check if the PC can reach the Turtlebot by typing:

```
ping 192.168.1.num
```

Change **num** by the last number of the IP address of the PC minus 100.

For instance, if the IP address of the PC is 192.168.1.204, then $num = 204 - 100 = 104$

Is the connection viable between the PC and the Turtlebot?

#ANSWER

What does the function ping do?

#ANSWER

2.2. ROS settings

- Edit the file ~/.bashrc with a text editor (Nano, Gedit, SciTE, vim, emacs, vi...)
- Check the following lines:
  - ROS_MASTER_URI=http://<PC IP Address>:11311
  - ROS_HOSTNAME=http://<PC IP Address>
- Edit the line(s) if necessary
- Source your bashrc file to take into account the modifications using:

```
source ~/.bashrc
```

2.3. ROS Master

- Start a ROS Master on your PC by typing the following command in a terminal:

```
roscore
```

2.4. Turtelbot Bringup

- Open a new terminal and connect to your turtlebot (the password is : **turtlebot**) using the following command:

```
ssh ubuntu@<TURTLEBOT IP ADDRESS>
```

- Launch the “bringup” nodes ON THE TURTLEBOT NOT ON THE PC

```
roslaunch turtlebot3_bringup turtlebot3_robot.launch
```

What does it do?

#ANSWER

3. Turtlebot Nodes and Topics

3.1. Turtlebot Nodes

What are the nodes launched with the “bringup” command line?

#ANSWER

What command did you use to list them?

#ANSWER

3.2. Turtlebot Topics

Please list below the topics created by the turtlebot nodes.

#ANSWER

Tip: alternatively, you may launch the rqt tool to retrieve more info.

3.2.1. Published topics

What topic(s) correspond(s) to data coming from sensors embedded in the turtlebot?

#ANSWER

Please describe the message structure for the “LiDAR” topic.

#ANSWER

Please do the same for the inertial (IMU) data.

#ANSWER

3.2.2. Subscribed topics

Please list the available subscribed topics.

#ANSWER

Play a music tune on the Turtlebot. What command line did you use?

#ANSWER

Please explain how you found the correct list of arguments.

#ANSWER

Make the Turtlebot spin at 0.1 rad/s. What command line did you use?

#ANSWER

Please explain how you found the correct list of arguments.

#ANSWER

4. Cartography

4.1. Teleoperation

4.1.1. Teleoperation from keyboard

WARNING! Place the robot on a safe ground! Do not test on the table as the robot might fall!

- On the remote PC, open a terminal and type the following command to start the teleoperation module using the keyboard:

```
roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch
```

You can drive the robot using the following keys:

- **w** : to go in front
- **x** to go backward
- **a** : to turn left
- **d** : to turn right
- **s**: to stop

Keep this node up and running.

How are the nodes exchanging data? Please explain.

Tip: You may use Node graph in rqt to look at the connections

#ANSWER

4.1.2. Teleoperation from Android app (*optional*)

WARNING! Place the robot on a safe ground! Do not test on the table as the robot might fall!

- Download the ROS Control Android app
- Connect your smartphone to the Wifi network SSID: RoboticLab
- Connect to the ROS Master by providing its IP address (i.e. <PC IP Address>)
- Go to Topic -> Preferences and Change the following topic names:
    - Joystick Topic should be /cmd_vel
    - LaserScan topic should be /scan
    - Odometry topic should be /odom

Enjoy! You can control the Turtlebot directly from your smartphone.

4.2. SLAM

SLAM stands for **S**imultaneous **L**ocalization **A**nd **M**apping**.** It is a technique to draw a map by estimating the current location in an arbitrary space. It uses the robot's odometry and kinematics and also scan data from the LiDAR sensor.

To start the SLAM node, type the following command on the PC:

```
roslaunch turtlebot3_slam turtlebot3_slam.launch
```

For your information, the TurtleBot3 supports several SLAM algorithms: *gmapping*, *cartographer*, *hector*, *karto* and *frontier_exploration*. You can change the SLAM algorithm by changing the *slam_methods:=<algorithm_to_use>* option in the previous command line. You may need to install additional packages.

To build a rich map, the robot needs to explore its environment. Therefore, use the teleoperation node to discover the world but avoid vigorous movements. The more space the robot covers, the more accurate the map will be.

4.3. Map

Once you have created a complete map of an area of interest, you can save it to the local drive. You may use it later on for navigation purposes for instance.

```
rosrun map_server map_saver –f ~/EENG/maps/LI104
```

You may need to create the directories beforehand where the map will be stored. Try to choose a descriptive filename.

What files were created? Please explain their roles.

#ANSWER

5. Navigation

WARNING! Place the robot on a safe ground! Do not test on the table as the robot might fall!

Navigation corresponds to the robot moving from one location to a specified destination in a given environment. For this purpose, a map that contains geometry information of free area and obstacles of the given area is required. This map was created and saved in the previous section.

Launch the navigation node using the following command:

```
roslaunch turtlebot3_navigation turtlebot3_navigation.launch
map_file:=$HOME/EENG/maps/LI104.yaml
```

The ROS tool Rviz (http://wiki.ros.org/rviz) is launched.

What is the green point cloud on the Rviz interface?

#ANSWER

5.1. Set initial pose

The first phase corresponds to the initial pose estimation of the robot:

- Press the **2D Pose Estimate** button in the menu of RViz, a large green arrow is then displayed
- Move the arrow to the current pose of the robot and while holding down the left mouse button, drag the green arrow to the direction where the robot's front is facing
- Repeat these operations until the Lidar point cloud overlays the saved map
- Use the teleoperation node to refine the robot pose (optional)

5.2. Set target pose

Once the robot is perfectly located on the map, navigation goals can be submitted.

- Press the **2D Nav Goal** button in the menu of RViz, a large green arrow is then displayed
- Move the arrow to the target pose of the robot and while holding down the left mouse button, drag the green arrow to the direction where the robot's front should be facing

The robot plans a path towards its destination avoiding obstacles based on the map. Then, the robot moves along the path.

N.B.: if the robot does not move and a warning (Costmap2DROS transform timeout) appears in the terminal, the clocks of the robot and the PC may not be perfectly synchronized. Perform the following action on the robot side:

```
sudo ntpdate –s < PC IP Address>
```

Once the robot starts moving, add an obstacle along the path. Describe the robot's behaviour. Look at the terminal for logs.

#ANSWER

6. Assessment

The assessment will be based on the behaviour (lateness, involvement, questions) during the lab session and a report.

The deadline to submit the report is October 20th 2024 at 11.59pm. The report should be in **English** and submitted as a **pdf** file via Moodle. The title of the report should be **AdvancedRobotics_ROS_Lab1_Student1Name_Student2Name.pdf** (please change *Student1Name* and *Sudent2Name* accordingly).

A penalty of 1 point will be applied for late submission every 5 min i.e. -1point if submitted between 12am and 12.05am, -2 points if submitted between 12.06am and 12.10am...

A penalty of 0.5 point will be applied if a figure does not have a cross-reference in the text, if the axis labels are missing, or if the legend/title are absent. These penalties can be accumulated i.e. up to -1.5 point / figure.

The document should be composed of an Introduction (1 page maximum), a Methods section (2 pages maximum), a Results section (2 pages maximum) and a Conclusion (1 page maximum). A Reference section should be added if necessary. The report should be based on the provided Word template.

7. Useful links

http://wiki.ros.org/

http://wiki.ros.org/turtlebot3_bringup

https://emanual.robotis.com/docs/en/platform/turtlebot3/overview/