Lab Procedure for Python
Forward Kinematics

Setup

1. It is recommended that you review Lab 1 – Application Guide before starting this lab.

2. Turn on the QBot Platform by pressing the power button once. To ensure the robot is ready for the lab, check the following conditions.

    a. The LEDs on the robot base should be solid red.

    b. The LCD should display the battery level. It is recommended that the battery level is over 12.5V.

    c. The Logitech F710 joystick's wireless receiver is connected to the QBot Platform. Before use**, always make sure the switch on top is in the X position and that the LED next to the Mode button is off.**

    d. Make sure your computer is connected to the same network that the QBot Platform is on. If using the provided router, the network should be `Quanser_UVS-5G`.

    e. Test connectivity to the QBot, using the IP displayed in the robot's LCD display, enter the following command in your local computer terminal and hit enter: `ping 192.168.2.x`

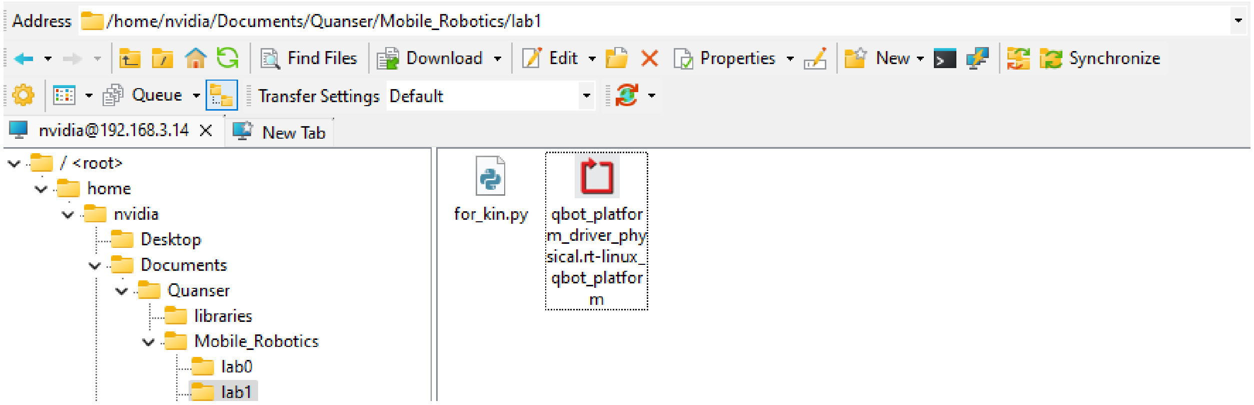
3. Open for_kin.py. On Section A, change the value of the variable "ipHost" to the IP address of your local Windows machine. To get this IP, open a Command Prompt and enter `ipconfig`. Your IP can be found under IPv4 address and should look like the one the QBot has, `192.168.2.X`.

Note: You will be coding on your local Windows machine and then transfer the code to the QBot Platform. You must go through step 4 – 7 every time a change is made in the code.

4. Open WinSCP, enter the IP address of the QBot Platform for Hostname, and enter "nvidia" for both Username and Password, then click the login button.

5. In the WinSCP window, transfer required files to run the application:

    a. navigate to "~*Documents**Quanser**Mobile_Robotics*" and create a new folder "*Lab1*", as shown in Figure 1. Then copy the updated for_kin.py to "*Lab1*". (You can just drag the file over). In addition, copy the qbot_platform_driver_physical.rt-linux_qbot_platform over. (You can just drag the files over).



Address /home/nvidia/Documents/Quanser/Mobile_Robotics/lab1

Transfer Settings Default

Figure. 1. Correct directory for python script

b. navigate to "~\Documents\Quanser\libraries\python" on the QBot Platform as shown in Figure 2, create these directories if they don't exist. Copy over the "hal" and "pal" folder from your local Windows machine, replacing existing files on QBot.

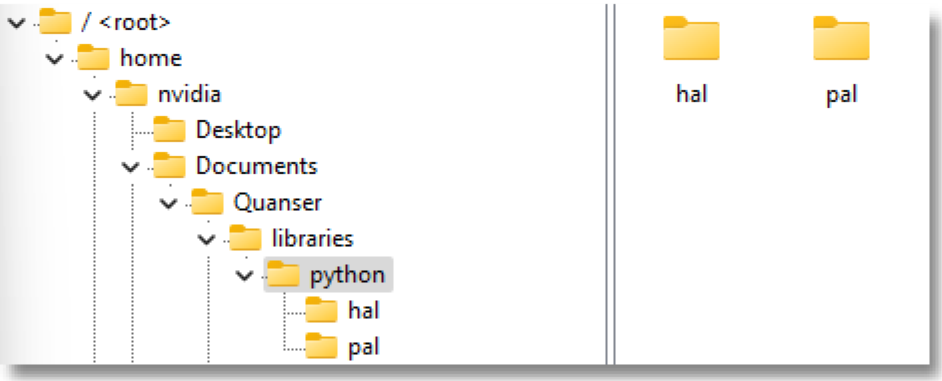


Figure. 2. 'hal' and 'pal' folder in correct directory

6. Run observer.py on your local Windows machine first to initiate receiving data feeds from the QBot Platform.

7. Run for_kin.py on QBot Platform:

   a. Open a PuTTY session by clicking in the top bars of the WinSCP window.

   b. In the PuTTY terminal, enter the password "nvidia".

   c. Navigate to the python script directory as shown in step 5.a. by using the `cd` command.

   d. Run the script using the following command:

```
sudo PYTHONPATH=$PYTHONPATH python3 for_kin.py
```

   e. When the script is run successfully, User LEDs will turn blue.

Drive with Individual Wheel Speeds Command

1. Press and hold the left button (labelled LB) to arm the robot. Notice that the LEDs turn green. Keep this button pressed as you teleoperate the QBot for future labs as well.

   Note: If your robot is ever in a position where it may collide with obstacles or people, disarm the robot by depressing the LB button (let go). The LEDs will turn blue again, indicating that the QBot Platform is disarmed.

2. While armed, use the following joystick sticks to move wheels while monitoring the **Motor Speed Plot** and **Body Speed Plot** window as shown in Figure 3. Take note of the wheel speeds, as well as what motion it produces for the QBot Platform body. **Body Speed Plot** displays turn speed from the gyroscope and body speeds from the forward kinematics model which you will compete later.

   a. move only the left stick up and down and determine the positive convention of the left wheel speed.

   b. move only the right joystick up and down and determine the positive convention of the right wheel speed.

   c. move both sticks to drive the QBot around. Take note of the stick combination that produce the following results,

      i. forward or backward motion

      ii. turn in place clockwise or counterclockwise.

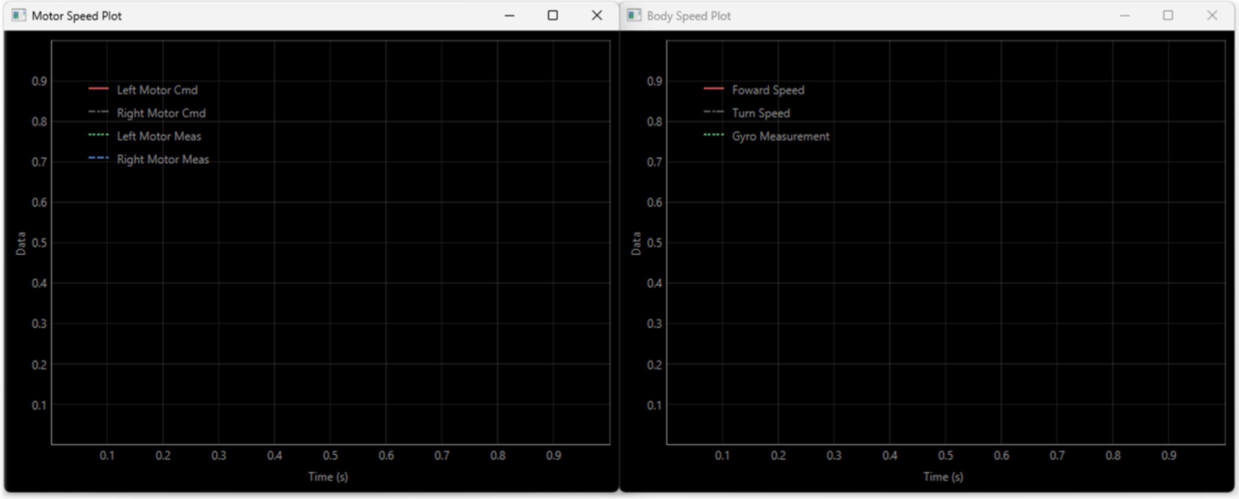


Figure. 3. QBot Speeds Scope

3. What relationship do you observe between the wheel speeds and turn speed of the robot?

4. Stand behind the robot and use both joysticks to control the QBot to follow the lines on the mat (straight or curved, or both). Take notes on the challenged experienced during this activity.

5. Using observer.py, open the downward facing camera feed as shown in Figure 4. Now, instead of walking behind the QBot, try making the robot follow the lines on the mat using only the downward facing camera feed. Document the strategies you utilized to follow the line and additional challenges faced when using only the camera feed.

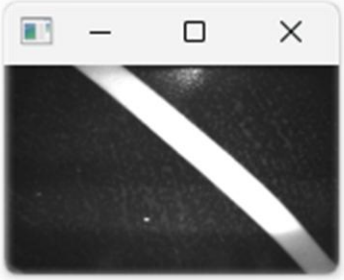


Figure. 4. Downward Facing Camera Feed

6. Stop the code by pressing the right button (RB) on the joystick.

Forward Kinematics Formulation

1. Go back to your local Windows machine and open for_kin.py. In Section B, right click on the **QBPMovement()** class and select "Go to Definition". This class handles the computation of forward kinematics as well as things needed in future labs, including inverse kinematics and line following, but these methods are incomplete. This class also contains physical QBot parameters.

2. Using the predefined global variables with the QBot's physical dimension, complete **diff_drive_forward_velocity_kinematics()** function under **QBPMovement()** to formulate the QBot forward kinematics equation via geometric derivation.

3. Save your changes to **QBPMovement()**. Using the function you completed on the previous step, go back to the for_kin.py file and complete the missing code in Section F. Note: The 'QBPMovement()' class was instantiated as 'movement' in Section B. This object can be used to reference the functions within the class.

4. Notice that the function you created is part of the Quanser Python Libraries, this change must be transferred to the QBot Platform as well. Following step 5 in the Setup section, copy over the "pal" and "hal" folders in your compute, as well as other changed files, replacing the existing ones on the QBot.

5. Run observer.py on your local Windows machine first, then run for_kin.py in the PuTTY session terminal, and drive the QBot.

6. Using the scope display window, verify that the turn speeds output from your forward kinematics function closely matches that from the gyroscope. Revise your function if needed. When you're satisfied with the output of your function, take a screenshot of the scope window before shutting down the script. Also monitor the forward speed of the robot.

7. What is the maximum forward or turn speed of the robot?

8. Stop your script using the RB button. Ensure that you save a copy of your completed files for review later.

9. Turn OFF the robot by single pressing the power button (do not keep it pressed until it turns off). Post shutdown, all the LEDs should be completely OFF.