Lab Procedure for Python
Inverse Kinematics

Setup

1. It is recommended that you review Lab 2 – 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 inv_kin.py. In 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**Warehouse_Robotics*" and create a new folder "*Lab2*", as shown in Figure 1. Then copy the updated inv_kin.py to "*Lab2*". In addition, copy the qbot_platform_driver_physical.rt-linux_qbot_platform over. (You can just drag the files over).

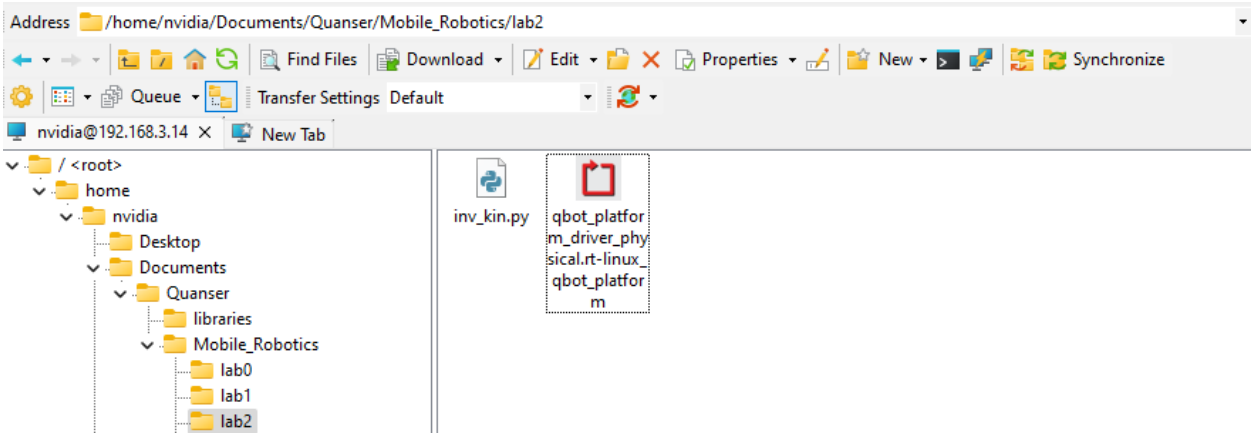

Figure 1. Correct directory for python script

b. navigate to "~\Documents\Quanser\libraries\python" on QBot 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.

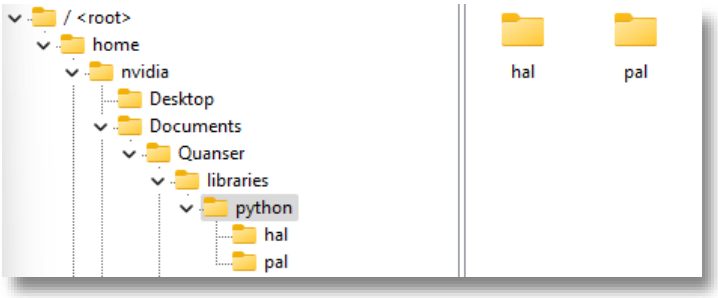

Figure 2. 'hal' and 'pal' folder in correct directory

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

7. Run inv_kin.py on QBot Platform:

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

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

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

   f. Run the script using the following command:

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

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

Inverse Kinematics Formulation

1. In Section B of inv_kin.py, right click on the **QBPMovement()** object and select "Go to Definition". Verify that **diff_drive_forward_velocity_kinematics()** function has been completed. If not, copy the code from last lab to compete the function.

2. When considering a pure forward velocity for a differential drive robot, what do you expect each wheel to do? How about when turning?

3. When commanding a positive turn velocity for the QBot (counterclockwise), which wheel should spin faster?

4. Navigate to **diff_drive_inverse_velocity_kinematics()** under the **QBPMovement()** class**.** This function is incomplete and must be compete before use. This function should take body speeds as input and output QBot wheel speeds in rad/s.

5. Complete the **diff_drive_inverse_velocity_kinematics()** function. Save your changes to **QBPMovement()**. Go back to the inv_kin.py and use the function you completed on the previous step to complete Section D. Note: The joystick commands are mapped to forward speed and turn speed, respectively.

6. Transfer the changed files to the QBot Platform as per step 5 in the Setup section.

7. Run observer.py on your local Windows machine first, then run inv_kin.py in the PuTTY session terminal, and drive the QBot. Focusing on the scope display window, how closely do the commanded wheel speeds output from your inverse kinematics function match the measured wheel velocities? Revise your function if needed.

8. How closely do the commanded body velocities match the measured velocities through forward kinematics?

9. When you're satisfied with the output of your function, take a screenshot of the scope windows.

10. What is the key difference between this application and the drive mode in the Play lab?

11. What is the maximum forward or turn speed of the robot you can command? What wheel velocity command does this correspond to?

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

13. 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.