Robot Lab 0: Safety and Commands to the Robot

This is a partner assignment.

At the end of this lab session a short verbal discussion is necessary with the lab TA

Completion Date: 09/20/2022

1 Introduction

In this class, we will be working with the Franka PANDA robot arm which is a 7DOF robot manipulator.

Robot lab 0 is an introduction for you to get oriented working with the robot. The goal of these sessions is not for you to become an expert user of the robot, but instead for you to be exposed to the safety guidelines, workflow, and capabilities of the robot in lab. Either you or your partner will sign up for **a 30 minute time slot**, during which time you are both required to go through a safety introduction to the robot.

The lab exercises are a mandatory component of the class. Robot lab 0 will have a completion grade. Nothing will be submitted to Gradescope. Instead, at the end of your 30 minute lab time you will have a short discussion with the lab TA checking your comprehension of the tasks covered. For future robot labs, you should discuss the results of the robot lab in your lab report.

In addition to a safety introduction or review at the start of each lab, small tasks using the robots are provided to help demonstrate the robot workflow. Just like the labs, these robot exercises will build on one another. It is important if you do not understand a step to ask for assistance early and often. After 25 minutes with the robot you will be asked to pack up and do a quick 5 minute review discussion with one of the lab TAs. This is to ensure that you have grasped the safety guidelines and the exercises done in lab.

Lab exercise 0 needs to be completed by **September 20, 2022**. Once **September 20, 2022** is over, you will no longer be allowed to do this lab. You can request an extension before the deadline if you need one due to a special situation such as illness.

2 Making a Lab Reservation

2.1 Making an account

All students in the class will need a ClusterMarket account. Create one at

`https://app.clustermarket.com/register/4uG2_gp_WQNOeN2EQ60-VQ`

If you have used Penn Equipment before (e.g. RPL space, etc.) and you already have a ClusterMarket account please send an email to Peter Bruno: `pbrun@seas.upenn.edu` with MEAM520 Fall 2022 in the email heading and let him know you need to be added to our course.

2.2 Making a reservation

Once you have an account, **you and your partner can reserve one 30 minute slot on the robot September 14 to September 20**. Do not sign up for a slot before September 14 as the robot lab will not be staffed. Only one partner should sign up for the slot, but both members of the group are required to attend the lab time. If both members of your group do not show up you will not be able to do the lab. Both group members should make a ClusterMarket account. This is how you will reserve time for your final project, so even if you do not make a reservation for the labs you will want to be able to reserve time for the final project.

Available time slots are listed on ClusterMarket. You will select a robot, then book time in the open windows. If for some reason there is no available time that you and your partner can make before the deadline please make a private post on Ed Discussion to the teaching staff. We will do our best to be accommodating, but please try to make the lab slots we have provided work.

2.3 Lab reservation rules

We have limited availability to work with the robots, so it is important that each partner group only signs up for **one 30 minute slot** that you share. Either partner may sign up on ClusterMarket, but only one partner should sign up for a slot per robot lab assignment. Attendance will be taken during the labs. Groups who attempt to take more than one 30 minute slot in the robot lab will receive a warning and will not be allowed to use the robot a second time. Repeated offenses will result in a zero for the assignment. Book early so that you are not stressed about getting the lab done before the deadline.

3 Getting to Lab

The robots are located in Towne B2 on the basement level. If you enter Towne from inside campus, you will walk down the stairs, and lab is the first door on your right. The lab room is locked at all times, and TAs will let students in. **Please show up at least 5 minutes early for your lab slot**. There will be a transition between students leaving lab and entry.

4 Lab Safety

Come to lab in **standard lab attire**: closed toed shoes, long pants, and hair tied back. Our lab space tends to be cold, so you may want to bring a jacket or coat.

When you arrive at lab there will be copies of the safety document accompanying each work station. The TA running your lab session will walk through this document with you and answer any questions.

We want you and your classmates to have a fun and safe experience in lab; to ensure everyone's safety we will be strictly upholding the rules in the lab documentation. Violations of the rules laid out in the safety guidelines will get 1 warning; infractions after this will result in asking you and your group mates to leave the lab space, forfeiting the rest of your lab time.

5 Using the Software Stop

The software stop will halt all motor commands to the robot and should be used when the robot is exhibiting unintended or undesired behavior. ***All teams should feel comfortable using the software stop.*** We will practice in non-critical moments so that in the future if the robot looks like it might hit itself or the table you are prepared to stop the robot.

Both teammates should test using the software stop (black button). One person should be in charge of running the code and the other is in charge of the software stop. Note that the next set of instructions are done in parallel and require you and your partner to communicate.

5.1 Your Task

The person running the code should do the following:

1. Make sure all of your teammates are clear of the robot workspace
2. Communicate to your partner that you are ready for the software stop to be released
3. Open two terminals, please know that if you want to kill code that is running use `Ctrl-C`.
4. Type the following steps in the first terminal

```
$ cd meam520_ws
$ ./franka.sh master
$ roslaunch franka_interface interface.launch
```

5. In a second terminal type the following:

```
$ cd meam520_ws/src/meam520_labs/labs/lab0
$ python software_stop.py
```

The person manning the software stop should do the following:

1. Communicate with your partner that you have released the software button (black button) which will turn the light on the robot blue before code is run
2. Once your partner has started the robot and it has begun to move hit the black button

When you are done:

1. Kill the processes in both terminals, `Ctrl-C`, and close the terminals.

Switch roles with your partner so that each person gets a chance to perform both jobs. Note the complete procedure will need to be redone because once the software stop is enabled it requires restarting the ROS node, which is why we ask you to close the terminals.

5.2 Questions to consider:

- What happened?
- Can you imagine scenarios where it would be easy to have an accident if clear roles for robot operation are not decided on ahead of experiments?

6 Getting information from the robot

The robot can report information about its current status and the positions of its joints. Look at the code in the core directory of your `meam520_labs` workspace.

We have provided some example functions that report information from the robot in `core/interfaces.py`. These are:

- `joint_limits(self)`
- `get_positions(self, include_gripper)`

6.1 Your Task

1. Modify the `demo.py` file, which can be opened with atom or a text editor such as vim, vi, emacs, or nano:

   ```
   $ cd ~/meam520_ws/src/meam520_labs/labs/lab0/
   $ atom robot_lab_demo.py
   ```

   Your job is to include at least one call to a function that returns information about the state of the robot. We know from our previous labs that we interact with the robot through a python Class. You are provided an instance of the robot arm, a line of code as follows:

   ```
   arm = ArmController()
   ```

   In the stub code there are commented sections on where you should write your own code. Go to the comment that says "Add Student Code HERE" and add a call to the above functions. The lab workstations have all of the same editors as you have on the VM (atom, emacs, nano, and vim).

2. To test the code, we will be using a two part work flow. First you will test everything in simulation and then you will test it on the robot.

3. **STEP 1: VERIFY IN SIMULATION**

   The ***most important*** rule of robotics is

   RULE 1: A system that does not work in simulation will not work in hardware.

   Corollary: Therefore, it is important any time you want to run code on the hardware platform, that you test it in simulation first.

   To run a simulation:

   (a) In a terminal, type the following:

   ```
   $ cd meam520_ws
   $ roslaunch meam520_labs lab0.launch
   ```

   (b) In a second terminal do the following:

   ```
   $ cd meam520_ws/src/meam520_labs/labs/lab0
   $ python robot_lab_demo.py
   ```

   Questions to always ask yourself:

   - What do you expect the output to be?
   - Did the output reflect what you expected? If no, then why not?

- Should you test on the robot if the output did not match what you anticipated?

4. **STEP 2: TEST ON HARDWARE**

   If the code works in simulation, then you can proceed testing it on hardware. The second most important rule of robotics is

   RULE 2: A system that worked in simulation may not work in hardware.

   Corollary: Therefore, DO NOT make any assumptions about whether or not your hardware test will succeed. Constant vigilance is the key!

   To run code on the robot:

   (a) Make sure all of your teammates are clear of the robot workspace

   (b) Release the software button (black button) which will turn the light on the robot blue (if this step is not done before launching ROS then the robot will not move)

   (c) NOTE: If you are using the same two terminals as from your software stop, you do not need to perform the `./franka.sh master` step. This sets environmental variables once in the terminal and it persists as long as the terminal is open. If you closed those terminals then open two terminals and complete the next steps completely.

   (d) In the first terminal do the following:

   ```
   $ cd meam520_ws
   $ ./franka.sh master
   $ roslaunch franka_interface interface.launch
   ```

   (e) In the second terminal do the following:

   ```
   $ cd meam520_ws/src/meam520_labs/labs/lab0
   $ python robot_lab_demo.py
   ```

   (f) When the script has finished executing, enable the software stop again (black button); the lights on the robot will turn white.

   Questions to always ask yourself:

   - What do you expect the output to be?
   - Did the output reflect what you expected? If no, then why not?
   - What could you do to improve your understanding of the output you are getting?

Notice above that the simulation workflow and the robot workflow are very similar. It is important to ensure that your simulation interface is similar to your robot hardware interface to minimize the number of surprises.

Congratulations!! You have finished your first robot exercise!