Lab 1: Getting Start with Franka Panda

Robot Autonomy

Prof. Oliver Kroemer

0 Prerequisites

In this lab, you will learn how to start the Franka Panda Robot, how the e-stop works, and how to control the robot with Python on your computer.

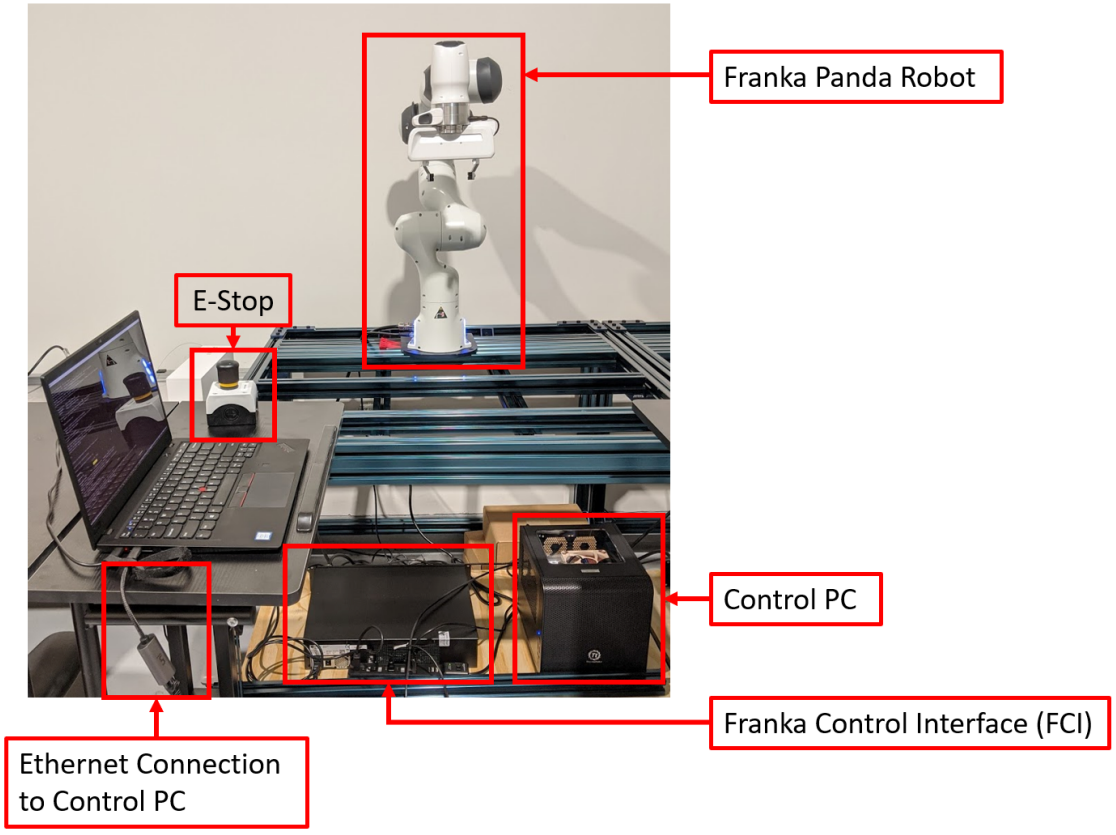

Figure 1: Lab Workstation

0.1 Computer Setup

Please install ROS Melodic on your laptop if you're on Ubuntu, or use the VM we provided below. The following is identical to the setup used in Homework 2, so if you have Homework 2 up and running, you're all set and can move on to the next section.

0.1.1 Ubuntu 18.04

1. Install ROS Melodic on your computer following instructions here: http://wiki.ros.org/melodic/Installation/Ubuntu. (Make sure to install the Desktop-Full Install version of ROS Melodic.)

2. Install libfranka and franka_ros using the following command:

```
sudo apt install ros-melodic-libfranka ros-melodic-franka-ros
```

0.1.2 Windows and Mac

For Windows and Mac users, we have created a virtualbox image which is available to be downloaded here: https://drive.google.com/a/andrew.cmu.edu/file/d/1jUnMLzQEw_bEQhxoGK_fWXZFrPPFXCOl/view?usp=sharing

1. You will need to download virtualbox from here: https://www.virtualbox.org/wiki/Downloads. (During installation, you must allow networking to pass through because we will be using this virtual machine in order to communicate with the robots during the labs and for your projects.)
2. You can load the virtualbox image by following the import OVA instructions here: https://www.maketecheasier.com/import-export-ova-files-in-virtualbox/
3. The username for the Ubuntu 18.04 account is: `student`. The password is: `16-662`.

Note: all command instructions given below assume you're on the VM running 18.04. If you're not on the VM and/or using 16.04, the exact commands may be slightly different.

0.1.3 Note on Safety

The robot has the following modes denoted by the indicator light on the side of the robot base:

1. Yellow - Lock Mode. The robot's joints are locked, and it cannot be moved. Locking and unlocking the joints are done in a webpage on the Control PC (you will do this in a later section). Once the joints are unlocked, the robot can be either in White or Blue.
2. White - Manual Mode. The robot may be moved by a human pressing down on the gray buttons near the robot hand. However, when the robot is in manual mode, it cannot be commanded via a program. White is achieved when the e-stop is pressed down.
3. Blue - Program Mode. The robot can be commanded with a program, but it cannot be moved by a human pressing down on the gray buttons. To achieve Blue, twist and release the e-stop.
4. Pink - Small error. Someone attempted to move the robot manually while it was in Blue. When in Pink, press down on the e-stop to return the robot to White.
5. Red - Serious error. This happens when the robot incurs a significant collision. A full restart is required to move the robot again.

When running a program that commands the robot, please:

1. Stay outside of the robot's workspace (defined by the edges of the table).
2. Always keep a hand on the e-stop. Under no circumstances should the e-stop be out of reach.

1 Installing Frankapy-Public Library and Dependencies

The library `frankapy-public` contains all the code needed to control the Panda arms. To install it on your machine, do:

1. Install pip if you haven't already:

   ```
   curl https://bootstrap.pypa.io/get-pip.py | sudo -H python3.6
   ```

2. Make a new Python 3.6 virtualenv:

   ```
   sudo apt install -y virtualenv
   mkdir ~/envs
   virtualenv -p python3.6 ~/envs/franka
   source ~/envs/franka/bin/activate
   ```

   You may want to add this last line to the end of your `~\.bashrc` file.

3. Make a new workspace folder:

   ```
   mkdir ~/ws
   cd ~/ws
   ```

4. Install frankapy-public inside the `~/ws` folder using the following instructions:

   ```
   git clone --recurse-submodules https://github.com/iamlab-cmu/frankapy-public.git
   cd frankapy-public
   pip install -e .
   /bash_scripts/make_catkin.sh
   source ~/ws/frankapy-public/catkin_ws/devel/setup.bash
   ```

   You may want to add this last line to the end of your `~\.bashrc` file.

2 Turning on and Connecting to the Robot

1. Flip on the power switch on the FCI. The robot should now boot up. It will take a couple minutes. The robot has finished booting up when the indicator light is solid yellow.

2. Connect the Ethernet cable from the Control PC to your laptop. Use the provided Ethernet-USB converters if needed.

3. Do this step if you're using the VM, otherwise skip to the next step.

   Before starting the VM, add a "Bridged Adapter" to the second networking interface, and set the attachment to the newly connected Ethernet interface.

   Do this by selecting the VM > Settings > Network > Adapter 2. Check "Enable Network Adapter" > Choose Bridged Adapter > Select the name of the Ethernet interface that's connected to the Control PC.

   The exact name of the Ethernet interface you need to choose may be different from the one shown in the screenshot.

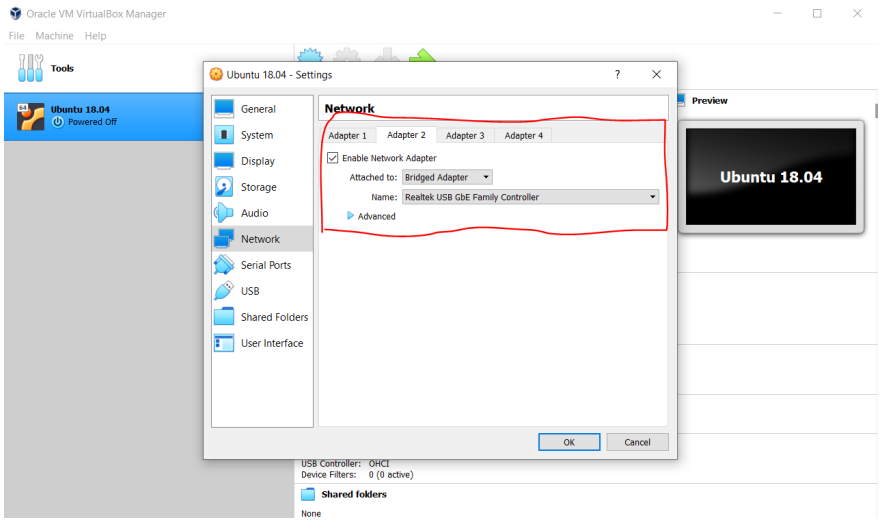

Figure 2: VirtualBox Network Setting

   Now launch the VM, and it'll have access to this Ethernet connection to the Control PC.

   If you're using the VM, the following steps are all done **inside** the VM.

4. Set the IP of your newly connected Ethernet Interface to have a static IPv4 address of `192.168.1.3`. See Figure 3 for details.

5. Add the name and IP of the Control PC to your network host file. The name depends on which robot your workstation has, and it should be in the format of `iam-<name>`. The Control PC IP is `192.168.1.2`.

   To do this on Ubuntu, add the following line to the end of `/etc/hosts` (editing this file requires sudo):

   `192.168.1.2 iam-<name>`

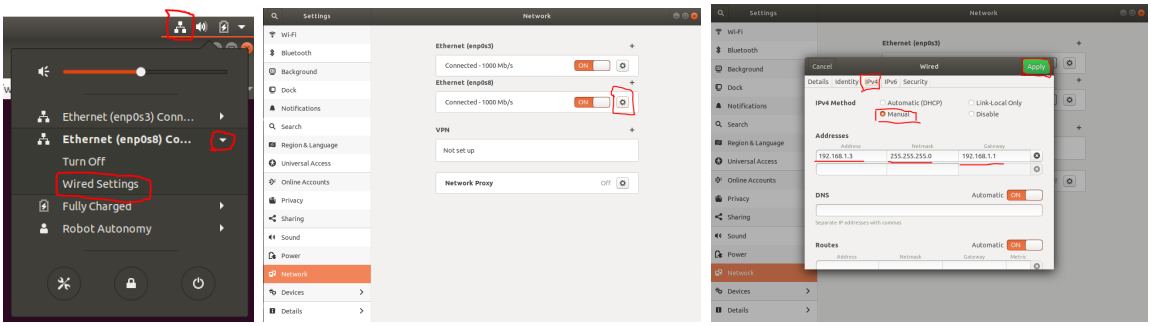

Figure 3: Steps (from left to right) of Setting the IP of the Local Machine in the Ethernet Connection to the Control PC

6. Now we will set up SSH keys on the Control PC so that you can SSH into the Control PC without a password. This also allows us to easily remote launch the robot-interface code on the Control PC.

   First we will generate the SSH keys on your machine. Run:

   `ssh-keygen -t rsa -b 4096 -C "your_email@example.com"`

   `[Press enter]`

   `[Press enter]`

   `[Press enter]`

   `eval "$(ssh-agent -s)"`

   `ssh-add ~/.ssh/id_rsa`

   You can view the newly generated SSH key:

   `cat ~/.ssh/id_rsa.pub`

   Now we will copy and paste this SSH key to the Control PC. Open a new terminal and ssh to the robot using the command:

   `ssh student@iam-<name>`

   Password is `16-662`

   Open this file on the Control PC with a text editor:

   `vim ~/.ssh/authorized_keys`

   Press the letter `i`.

   Press the `enter` button on your computer to create a new line.

   Press your `up` arrow key.

   Copy the entirety of your machine's `id_rsa.pub` file that was printed out earlier.

   Press the keys `Ctrl`, `Shift`, `v` in order to paste your key into the `~/.ssh/authorized_keys` file.

   Press the key `Esc`. Then type `:wq` and press your `enter` button to close the file.

   Now you should be able to SSH into the Control PC without typing the password.

7. If you're using the VM, skip this step.

   We also need to add your computer's hostname to the Control PC's `/etc/hosts` file. Use the previous instructions in order to modify the file using vim or your own text editor.

   SSH into the Control PC with the `student` account and add the following line to the Control PC's `/etc/hosts` file:

   `192.168.1.3 <your computer name>`

8. Unlock the joints on the robots. To do this you'd need to open up a webpage on the Control PC. Do this by:

   `ssh -X student@iam-<name>`

   Launch Firefox by running `firefox` in the terminal.

   (If running `firefox` or `-X` is giving errors, try restarting the terminal.)

   Then open the webpage `172.16.0.2` to unlock joints:

   (If it prompts for a login, the username is `admin`, and the password is `shavethepandas`)

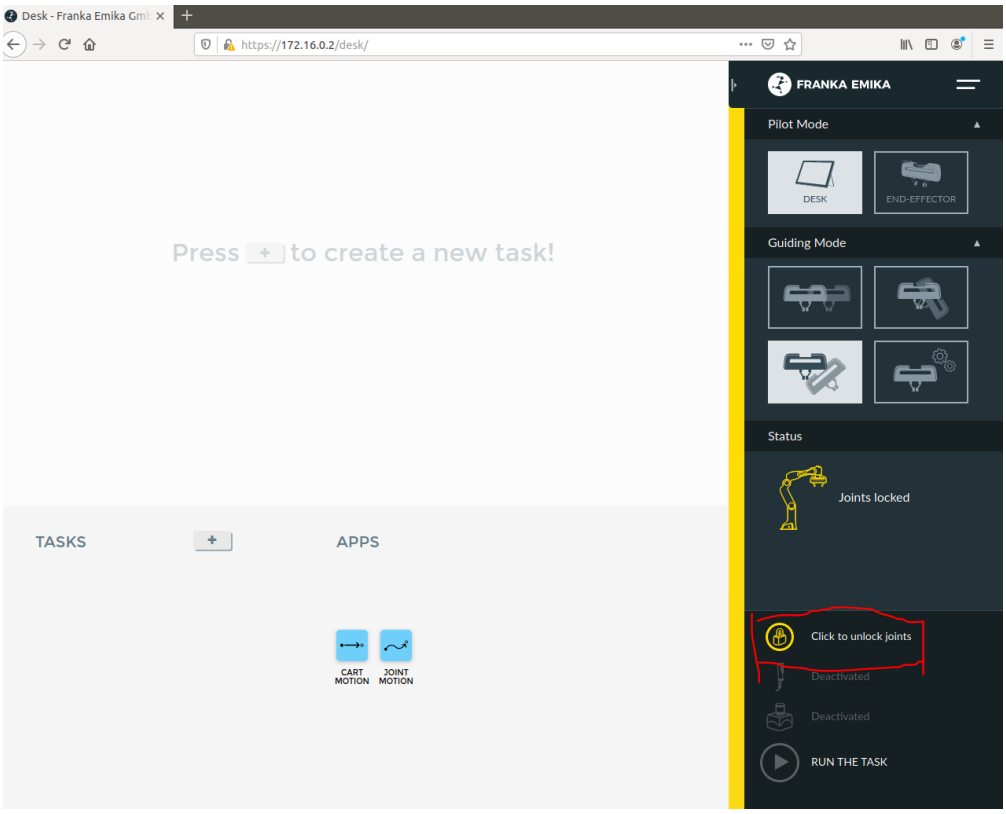

Figure 4: Unlock Joints

The joints are unlocked when the indicator lights are solid white.

3 Controlling the Robot

1. Clone the labs repo on your machine:

   ```
   cd ~/ws
   git clone https://github.com/iamlab-cmu/robot-autonomy-labs.git
   cd robot-autonomy-labs
   ```

2. Set the robot to command mode. Do this by twisting the e-stop and releasing it. The e-stop should prop up from its original position, and the indicator light should turn solid blue.

3. Remote launch the robot control software on the Control PC. Do this by opening a new terminal on your machine, `cd` to the `frankapy-public` folder you cloned earlier, and running:

   ```
   cd ~/ws/frankapy-public
   bash bash_scripts/start_control_pc.sh -i iam-<name>
   ```

   This should launch 3 new terminals indicating the server is running on the Control PC.

4. Next open another terminal that you will use to run the scripts.

   If you have not added the lines to your `~/.bashrc` file, you will have to do this every time.

   ```
   source ~/envs/franka/bin/activate
   source ~/ws/frankapy-public/catkin_ws/devel/setup.bash
   cd ~/ws/robot-autonomy-labs
   ```

5. Run the example script in the `lab1` folder:

   ```
   python lab1/move_robot.py
   ```

   This script will demonstrate a few common commands used to interact with the robot.

6. Now we'll try using the e-stop. Run the previous script again, but while the robot is moving, press down on the e-stop to stop the robot. The indicator light should turn white.

   You can move the robot during this mode by pressing on the two gray fingers as shown in the Figure 5.

   After the robot has come to a stop, you can resume controlling the robot by twisting and releasing the e-stop (the indicator light should turn blue), closing all 3 of the terminals created by the `start_control_pc.sh` script, and rerunning `start_control_pc.sh`.

7. Now do `python lab1/run_guide_mode.py` script in the `lab1` folder.

   This script will bring the robot into "guide mode," allowing you to freely move the robot around without pressing the gray buttons. This is useful for physically setting the robot's pose or joint angles and recording those values afterwards.

Figure 5: 2 Buttons that can move the robot when in White mode.

8. Now do `python lab1/run_guide_mode.py` script in the `lab1` folder.

   This script will bring the robot into "guide mode," allowing you to freely move the robot around without pressing the gray buttons. This is useful for physically setting the robot's pose or joint angles and recording those values afterwards.

9. To show this actually working, first run the `python lab1/reset_arm.py` script in the `lab1` folder.

   Then run the `python lab1/record_trajectory.py` script in the `lab1` folder.

   Move the robot arm around for 10 seconds, and then when the script ends, run the final script in the same folder: `python lab1/run_recorded_trajectory.py`

10. If you have time left in the lab, play around with the different example commands in `move_robot.py` and create your own script that moves the arm.

4 Turning Off the Robot

It's important to return the robot to its home position and turn the robot off when it's not being used.

1. Reset the robot to its home position by running the following script in the frankapy-public folder:

   ```
   cd ~/ws/frankapy-public
   ```

   ```
   python scripts/reset_arm.py
   ```

2. Close the terminals related to the Control PC to shut off the remote server.

3. Set the robot to manual mode by pushing down on the e-stop. This should return the indicator light to solid white.

4. Turn off the robot by returning to the webpage on the Control PC and clicking shutdown:

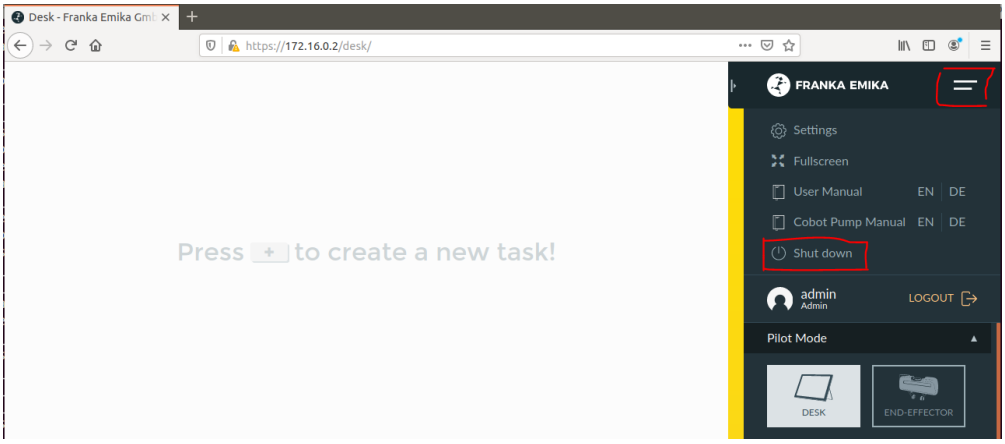

Figure 6: shutdown

Important! Wait for a minute for the robot to shutdown, then flip the physical switch on the FCI (do not flip the switch before the robot shutdown has finished).