ME 586: Biology-Inspired Robotics

University of Washington, Prof. Sawyer B. Fuller

Problem Set 3 (*updated 2023.02.07*)

Goals: Install the Crazyflie development software and learn how to program it to perform simple flight maneuvers and plot the result.

In this problem set, you will install the Crazyflie software and write a program to fly a palm-sized Crazyflie helicopter in a pre-programmed pattern, providing a plot of the estimated trajectory it took. You will work in small groups, each of which shares a single helicopter, but you will write your own control and analysis code.

1. Follow the getting started with Crazyflie instructions given in the link below, performing any assembly steps if you have an unopened kit. Install the mobile phone app if you want, but it is very hard to control without the extra sensors of the Flowdeck. Attach the Flow deck underneath the helicopter, matching the icons to determine orientation.
   https://www.bitcraze.io/documentation/tutorials/getting-started-with-crazyflie-2-x/

   (a) In that same page, follow the provided instructions to install the Crazyflie client and development software on your computer using either the virtual machine ("VM") or "Windows/Mac/Linux" local install option (direct link).
   - The VM is easier, but uses more disk space (∼10-20 Gb).
   - For the Windows/Mac/Linux option, do not "install from source." That option is provided only if you intend to modify the Crazyflie's software itself. If you get an error about a missing pysdl2, you may need to additionally perform the following:
     ```
     pip install pysdl2 pysdl2-dll
     ```

2. Download the controller skeleton code `takeoff_and_land.py` provided under the software_examples section of the public website.

3. Perform a short flight. Turn on the Crazyflie or plug in its battery while it is sitting still on a flat surface in an open space. It will beep and spin the four propellers briefly. Then, in a terminal (e.g. `cmd.exe` on Windows or `Terminal.app` on a Mac) run the Python code you downloaded in the previous step:
   ```
   python takeoff_and_land.py
   ```

4. Now, program your Crazyflie to perform a sequence of commands, flying in a square-shaped pattern. Use a text editor (I suggest VS Code or PyCharm) to create a new file based on the given above, changing code between the ### symbols to command motions in sequence. You may want to consult the step-by-step guide to using the Motion Commander interface here:
   https://www.bitcraze.io/documentation/repository/crazyflie-lib-python/master/user-guides/sbs_motion_commander/

5. Create a plot of the path of your helicopter using Matplotlib. The `takeoff_and_land.py` program continuously prints to the terminal the following quantities the helicopter as it flies:
   time (s), $\hat{x}$ (m), $\hat{y}$ (m), $\hat{z}$ (m), $\dot{\hat{x}}$ (m/s), OF_x (pixels).
   Capture these into a text file using the terminal command
   ```
   python takeoff_and_land.py > data1.csv
   ```
   Then create another Python script that loads this data into an array using
   `data = np.loadtxt('data1.csv', delimiter=',')`
   Use Matplotlib (or your favorite plotting system) to plot the helicopter's x-y trajectory. For example, `x = data[:,1]; y = data[:,2]; plt.plot(x,y)`. Make sure to label your plot axes. Save your plot using `plt.savefig('myfigname.pdf')` and submit your plot and code.