Flybratron Demo Pipeline: Intro

This demo provides a full pipeline for operating Flybratron, Kinefly, and Axoscope. You will learn to:

1. Launch and operate Kinefly
2. Launch Flybratron and run autotune
3. Run experiment scripts and record data in Axoscope
4. Load and analyze recorded Axoscope files

Prerequisites

- Linux machine with ROS and Flybratron software installed
- A tethered fly (for experiments, not required for autotune)
- Axoscope installed and configured

Clone repository

You can get everything you need from the GitHub repository:

- **Clone with Git**

```
git clone https://github.com/silentkind05/Flybratron_demo.git
```

1. Launch Kinefly

1. Open your terminal

- Shortcut: `Ctrl + Alt + T`
- Or find **Terminal** in your applications menu

2. Run the launch command (adjust the path if needed):

```
roslaunch Kinefly flyrig.launch
```

3. Wait until you see Kinefly has started successfully.
4. To close the Kinefly: `Ctrl + C`

- Axoscope installed and configured

2. Launch Flybratron & Run Autotune

1. Open a second terminal

- Shortcut: `Ctrl + Shift + T`
- Or choose **File** → **New Window** in your terminal menu

2. Change into the Flybratron examples folder (update the path as needed):

```
cd
'/home/fponce/work/wbd/flybratron/software/python2/flybratron/examples'
```

3. Run autotune:

Before running autotune, check parameters by running:

```
python get_param.py
```

If under **compensation_table**, the *phase* values are all 0 and the *amplitude* values are all 1, it means you have not yet run autotune.

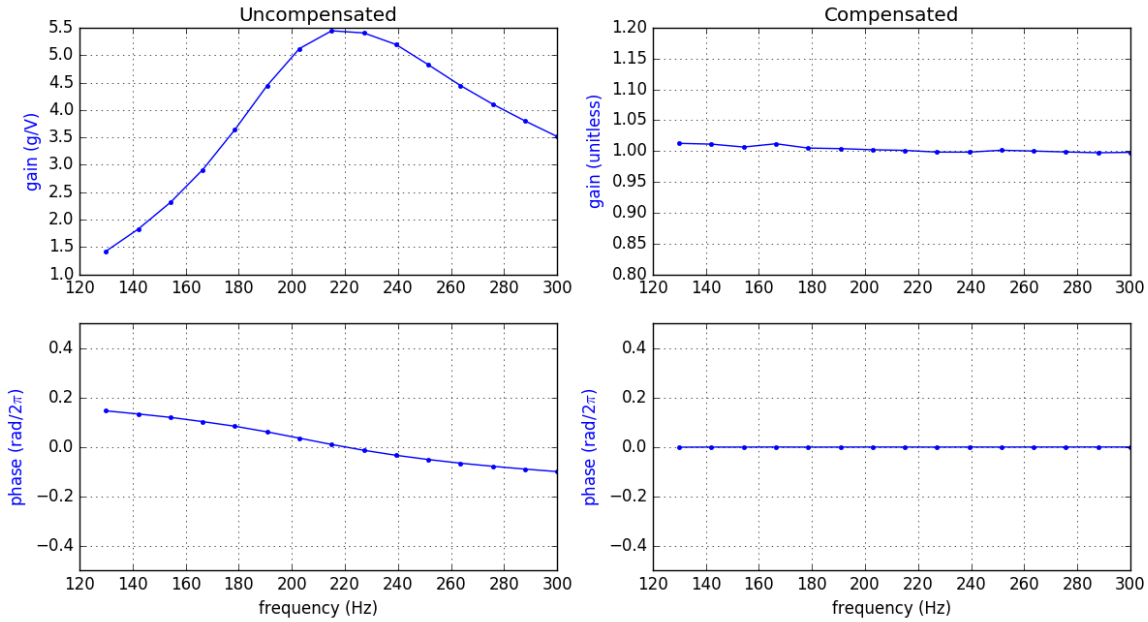
```
python autotune.py
```

4. Verify the new parameters:

```
python get_param.py
```

You should see a compensation table in which the phase and amplitude values are no longer the default settings.

A reference compensation table for yaw:



> A flat line in the compensated table (gain = 1, phase = 0) indicates a successful autotune.

5. (Optional) To manually control Flybratron from IPython:

```
python set_operating_mode.py free

python set_amplitude.py =.5 # change to your desired amplitude

python set_amplitude.py = 0 # change the amplitude back to zero

python set_operating_mode.py sync # Turn off the manual mode
```

3. Run Your Experiment Script

1. Open a third terminal (`Ctrl + Shift + T`)
2. Change to your experiment folder:

```
cd ~/work/ivo/
```

3. Start IPython:

```
ipython
```

> ⚠️ **Don't forget:** Press **Record** in Axoscope **before** running your experiment script.

4. Within IPython, find optimal phase:

```
run -i Yw_PH.py
```

This will iterate through a window of different phase leads or lags of the fly wing beat to find the optimal phase that elicits a **symmetric** response.

5. Apply optimal phase:

Once you obtain an optimal phase (e.g., `-0.05`), open your experiment script and set:

```
phase_use = -0.05 # replace with your optimal phase value
```

Save your script before moving on to the next step.

6. Run your script (replace with your filename):

Before running your script, make sure to set the parameters to your desired values.

```
run -i your_experiment_script.py
```

> ⚠️ To **stop** the experiment midway, press `Ctrl + C` to terminate the Python process.

Lists of experimental scripts:

Yaw

1. **Yw_PH.py**: phase analysis
2. **Yw_AM.py**: amplitude set test
3. **Yw_DU.py**: duration set test

4. Launch Axoscope

1. On the Axoscope computer, open the Axoscope application.
2. Click **Record** to begin capturing data **before** running your experiment script.
3. After your experiment completes, click **Stop**.
4. Go to **File** → **Last Recording**.
5. Select **Save** and choose your destination folder for later data processing.

5. Data Analysis Demo

1. **Set up your Jupyter notebook environment**

   Build and activate the Conda environment:

   ```
   conda env create --name flybratron --file environment.yml

   conda activate flybratron
   ```

2. **Access code samples and data sets**

   All demo materials live under the demos/ folder in your cloned repo:

- **Demo notebooks:** Open `demo_notebooks` folder
- **Data sets:** Open `data_sets` folder

3. **Run a demo notebook**

```
cd ~/Flybratron_demo/demo_notebooks

jupyter notebook
```