Ana de Sousa web: *linkedin.com/ana.sousa* anacsousa@lara.unb.br

FES knee trajectory control [2019-07]

About the package

In previous work, we created a simulation environment for FES gait control of one leg (de Sousa et al., 2019). The system tracks knee and hip trajectories of a hybrid neuroprosthesis (HNP), composed of a hip orthosis and FES-controlled knee motion. In this work, we control knee movement with four different FES knee controllers (bang-bang, PID, PID-ILC and PID-extremum seeking) on quadriceps and hamstrings.

The code was run with Ubuntu 16.04.6 LTS 64-bit Operating System, Python 2.7 and the Robot Operation System (ROS[1]).

Materials: We used the RehaStim (Hasomed, Germany), an 8-channel stimulator that provides biphasic current pulses and enables online update of stimulation amplitude and pulse width (Figure 1). Through surface electrodes on the subject, the stimulator applies the corresponded signal to the quadriceps and/or hamstrings muscles. To calculate the crankset cadence speed, we first measured the angle with an inertial sensor composed of 3-axis accelerometer, gyroscope, and magnetometer. The sensor (3-space, YEI Technology, USA) features onboard quaternion-based Kalman filtering algorithms to estimate angles. We also used a chair that allows regulations to fixate the torso and leg, and also allows free movement of the knee during the test.

Libraries: In the folder *yourpath\FES_Knee_2019\.idea*, you find scripts related to the github environment. We do not recommend changing any of those files.

In the folder *yourpath\FES_Knee_2019\cfg*, you find the script *Tutorials.cfg*, which uses the graphical user interface (GUI) tool from ROS called rqt[2]. Be careful when changing this script.

In the folder *yourpath\FES_Knee_2019\data*, you find data examples from experiments, and Matlab scripts to plot data.

In the folder *yourpath\FES_Knee_2019\launch*, you find the script *controllers.launch*, which uses the launch tool from ROS to launch multiple ROS nodes locally and remotely via SSH[3]. We do not recommend changing any of those files, unless you create new nodes.

In the folder *yourpath\FES_Knee_2019\msg*, you find the script describing a specific message from the inertial measurement sensor. We do not recommend changing this file.

In the folder *yourpath\FES_Knee_2019\scripts*, you find scripts that describes the nodes and topics from ROS (Figure 2)[4]. We do not recommend changing any of those files.

Other files in the folder are related to github (e.g., *README.md*, need update) or to the rqt perspectives (e.g., *gait_perspective.perspective*).

Run it: After plugging both stimulator and the inertial sensor to the computer, connect the hardware (stimulator and IMU dongle) using the corresponding USB cables. To control the knee trajectory, apply electrodes on the quadriceps (channel 1) and hamstrings (channel 2).

For running the code, open three new terminal windows (or three tabs in the same window) and follow the instructions below:

1. Configure it for running the main script. Initially, check which ports correspond to the dongle and the stimulator; usually, this step is only necessary once. Change the permissions using
   - sudo chmod 777 /dev/tty/ACM0
   - sudo chmod 777 /dev/tty/USB0
2. Start the control system in terminal 1
   - roslaunch ema_fes_controllers controllers.launch
3. Start user interface in terminal 2
   - rqt

[1]Follow the instructions for *Installing and Configuring Your ROS Environment*: http://wiki.ros.org/ROS/Tutorials/InstallingandConfiguringROSEnvironment

[2]Follow instructions for understanding *rqt* here: `http://wiki.ros.org/rqt`

[3]Follow instructions for understanding *launch* here: `http://wiki.ros.org/roslaunch`

[4]Follow instructions for understanding *nodes* here: `http://wiki.ros.org/pt_BR/ROS/Tutorials/UnderstandingNodes`

4. Start ros recording system in terminal 3 (optional)
    - roscore record -a

After opening the user interface (Figure 3), it is necessary to set the maximum stimulation parameters (pulse width and current), which muscles are going to stimulate, which controller to use - and its initial parameters -, and the path of the reference file for trajectory tracking (.mat). Then you should press the button to start control.

To finish the experiment, exit the rqt or press *control* + *c* at each terminal.

Results from simulations: After each simulation, the code saves the results in *yourpath*\ *FES_Knee_2019*\ *Data*. I added one example for better understanding:

- test_noCA_L_3_2019-07-02 10_31_15.228324.mat
  The files are named based on the initial parameters, i.e., no coactivation of antagnoist muscles (noCA), left leg (L), controller 3 (PID-ES), date and time of the experiment (2019-07-02 10_31_15.228324).

In this folder, there are functions to visualize the results. In MATLAB, run the file *PLOT_RESULTS.m*, chose the dataset you want to visualize, and MATLAB will extract some essential data and plot them. The first plot shows how the control parameters (K_p, K_i and K_d changes over time), the second plot shows the control signal sent to the stimulator, the third plot shows the knee angle response, the fourth plot shows the error based on the reference, and the last plot shows the pulse width applied to the muscles for both quadriceps (pw_q) and hamstrings (pw_h).

If you chose to run the rosbag function, you also get a rosbag file that can be replayed in ROS. In future versions, we will provide files for extracting information from the rosbag and plotting in MATLAB.

Final notes: Before testing the controller, use the stimulator *physiotherapist mode* to test the currents to find the maximum tolerable currents for quadriceps and hamstrings. Remember to change it back to the *science mode* for the control.

Put the inertial sensor over the ankle with a strip with the green light point towards the movement (pitch angle).

With ROS, it is possible to create new nodes for other hardware, so the code is adaptable for different systems.

References

de Sousa, A. C. C., Freire, J. P. C. D., and Bo, A. P. L. (2019). Integrating hip exosuit and FES for lower limb rehabilitation in a simulation environment. *IFAC-PapersOnLine*, 51(34):302–307.

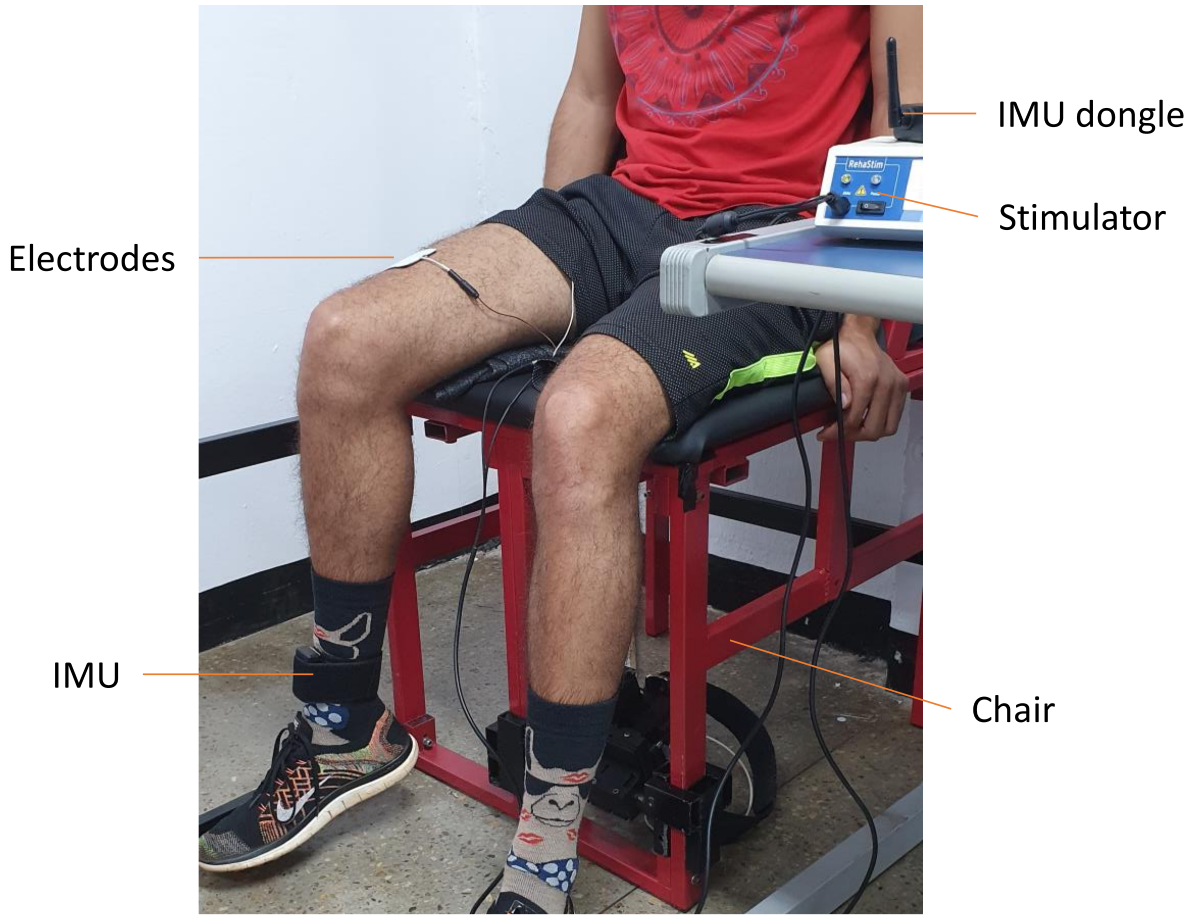

Figure 1: Volunteer and equipment during experiment.

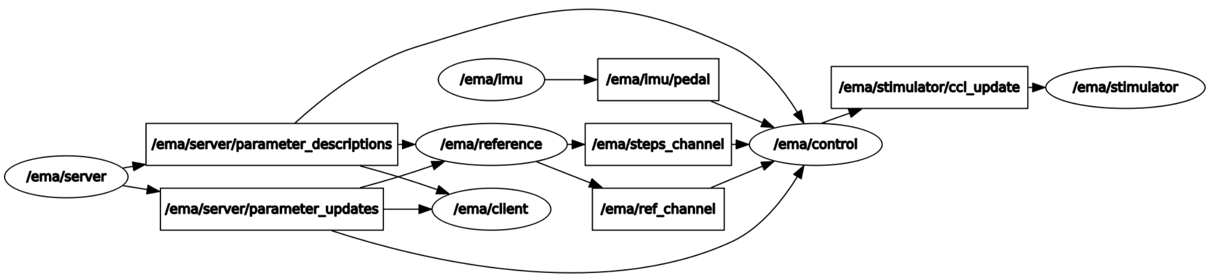

Figure 2: Diagram of nodes and topics of the experiment. ROS nodes are illustrated by elliptical shapes, and ROS topics are illustrated by arrows (inward and outward arrows represent subscribing and publishing, respectively).

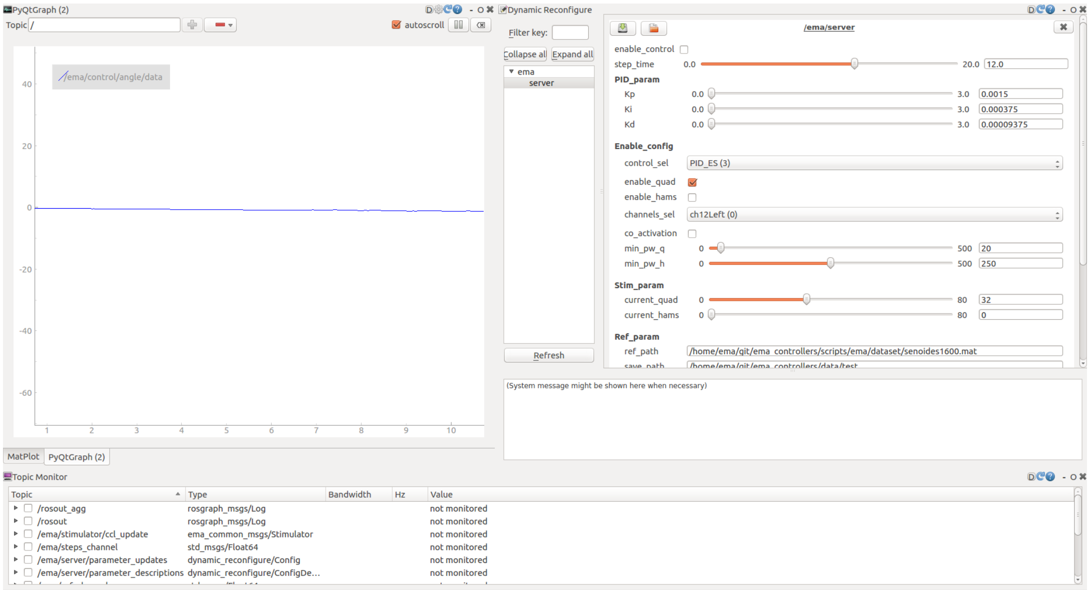

Figure 3: ROS rqt GUI for the FES knee control. Print screen of the application on the Ubuntu 16.04 operating system.