FES cycling with passive orthoses [2019-09]

About the package

In previous work, we created a simulation environment for FES cycling control (de Sousa et al., 2016). In this release, we investigated the use of passive knee orthoses in cycling (de Sousa et al., 2019). The newest version compares the cycling cadence and quadriceps excitation using an FES cycling simulation platform for different spring torques and ranges (c.f., (de Sousa et al., 2019)). Based on Hunt (2005); Bo et al. (2017), we defined ranges to excite each muscle to achieve cycling (Figures 1a and 1b, c.f., (de Sousa et al., 2016)). The control structure (Figure 2) incorporates the musculoskeletal dynamics, the predefined range angles, and the chosen controller, c.f., (de Sousa et al., 2016).

The code was run with MATLAB R2018a, OpenSim 3.3, Windows 10 Home 64-bit Operating System, and uses the OpenSim API in Matlab[1].

OpenSim model: In the folder *yourpath\FES_Cycling_2019\Model*, you find the FES cycling model and objects. You can visualize the cycling model loading *FESCyclingModel.osim* in OpenSim 3.3 (this step is not necessary to run the code). This file was created with the script *createModel_passiveOrthosis.m* (lines 14 to 22).

Libraries: In the folder *yourpath\FES_Cycling_2019\OpensimLibrary*, you find scripts from the OpenSim API. We do not recommend changing any of those files.

In the folder *yourpath\FES_Cycling_2019\CyclingLibrary*, you find specific scripts files for the cycling simulations. Be careful when changing these scripts.

Run it: For running the code, it is necessary to describe the initial configuration. There are two examples of those configurations in files *yourpath\ FES_Cycling_2019\RUNME_BB.m* and *RUNME_PID.m*. You may change the parameters in these files.

- RUNME_BB.m : runs the FES cycling using the open-loop (bang-bang) controller.
- RUNME_PID.m : runs the FES cycling using the PID controller.

During the run, we may see some information, e.g., *40/250: gear: -58.740003 , vel: 201.302303, u: 1.000000*. That means that it is the 40th simulation of 250 simulations, the gear angle is at -58.7º, the crankset speed is at 201.30ºs, and the intensity of muscle excitation is 1.

The simulations may take a while. Depending on the computer, a 20 seconds simulation may take around 30 minutes.

When finished, Matlab saves the workspace data and opens a figure showing the plots.

Results from simulations: After each simulation, the code saves the results in *yourpath\ FES_Cycling_2019\Results*. I added one example for better understanding:

- cycling__Tf5_P0_Kmax0_s92_r20_L0_M0_STIM0.5_QHG110_C_Open-loop_R250_FatRL0_0.sto
  After loading the model in OpenSim, you can load this .sto file and visualize the cycle (this step is not necessary to run the code). If the path is too long, OpenSim may present an error, try transferring this file to another location, e.g., Desktop.
- cycling__Tf5_P0_Kmax0_s92_r20_L0_M0_STIM0.5_QHG110_C_Open-loop_R250_FatRL0_0.mat
  When we open this file in MATLAB, a figure and the workspace is loaded. The figure presents four plots over time. (1) the plot only shows quadriceps excitation, even if other muscles are excited (will be improved for newest versions). (2) knee orthoses torque, in this case, is zero because there is no orthoses. (3) gear and knee angles. (4) angular speed at the crankset.

The files are named based on the initial parameters, i.e., final time (Tf5, 5 seconds), initial position of the right leg (P0, i.e., position 0º), passive orthoses parameters (Kmax0_s92_r20, i.e., maximum torque 0Nm - no orthoses -, starting angle 92º and range angle 20º, c.f., (de Sousa et al., 2019)), load and motor on the crankset (L0_M0, i.e., 0Nm torque on the load and on the crankset, i.e., only the muscles interact with the movement), excitation for muscles (STIM0.5, i.e., muscles are excited at 0.5 intensity, goes from 0 to 1), muscles to be excited (QHG110, i.e., quadriceps and

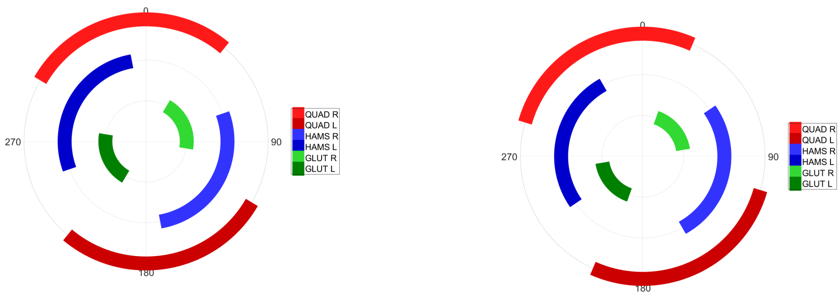
[1] Follow the instructions for *Setting up your Matlab Scripting Environment*: https://simtk-confluence.stanford.edu:8443/display/OpenSim33/Scripting+with+Matlab

hamstrings turned on, gluteus turned off), controller (C_Open-loop, i.e., controller is an open-loop where we do not control intensity of excitation), reference speed (R250, in the cause of controllers different from the open-loop, that is the reference [º/s] for the controller), fatigue (FatRL0_0, i.e., no simulation of fatigue).

The script overwrites old files if the newest file contains the same basic configuration (usually because those files contain the same results). If you need to keep old data, it is necessary to transfer them to another folder.

Final notes:

- The RUNME files call for cycling_script.m (*yourpath\FES_Cycling_2019*), which initializes the system and calls for the OpenSim API along with the cycling_control.m (*yourpath\FES_Cycling_2019\CyclingLibrary*). The cycling_control.m reads the angle and, based on the initial setup, calls for the script corresponding to the controller (e.g., cycling_pid.m).
- With the results of the chosen controller, cycling_control.m applies excitation to the muscles.
- In folder *yourpath\FES_Cycling_2019\CyclingLibrary*, you find the scripts that generate the profile of excitation of muscles, i.e., when the code should excite quadriceps, hamstrings or gluteus (Profile from Figure 1a).



(a) Muscles range without phase adjustment.

(b) Muscles range with phase adjustment.

Figure 1: Predefined muscles range angles for excitation during one pedal stroke for the right and left legs. Right and left quadriceps (QUAD R and L) in red, hamstrings (HAMS R and L) in blue, and gluteal (GLUT R and L). The average crankset cadence $\bar{\dot{\theta}}_c$ is 260°/s.

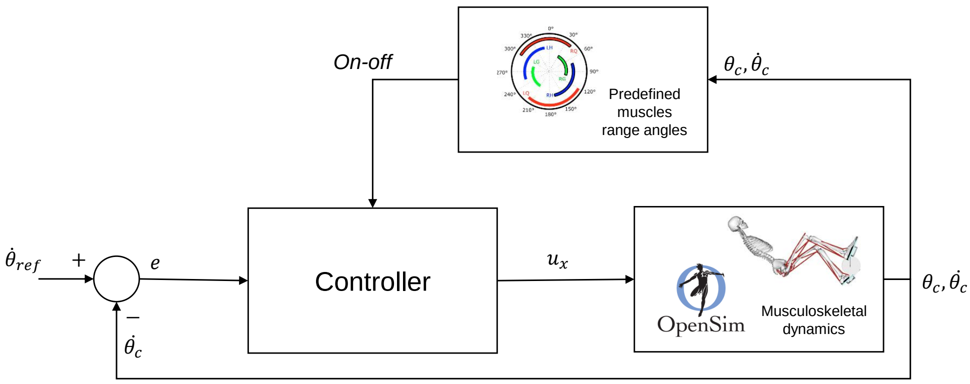


Figure 2: In this cycling control architecture, the controller provides a signal (u_x) based on the error (e) between the reference cadence speed ($\dot{\theta}_{ref}$) and the crankset speed ($\dot{\theta}_c$). The controller also considers the muscle range angles for excitation based on the crankset position (θ_c). With the control signals (muscles excitation), OpenSim calculates the musculoskeletal dynamics (θ_c and $\dot{\theta}_c$).

References

Bo, A. P., Da Fonseca, L. O., Guimaraes, J. A., Fachin-Martins, E., Paredes, M. E., Brindeiro, G. A., De Sousa, A. C. C., Dorado, M. C., and Ramos, F. M. (2017). Cycling with Spinal Cord Injury: A Novel System for Cycling Using Electrical Stimulation for Individuals with Paraplegia, and Preparation for Cybathlon 2016. *IEEE Robotics and Automation Magazine*, 24(4):58–65.

de Sousa, A. C. C., Ramos, F. M., Narvaez Dorado, M. C., da Fonseca, L. O., and Lanari Bó, A. P. (2016). A Comparative Study on Control Strategies for FES Cycling Using a Detailed Musculoskeletal Model. *IFAC-PapersOnLine*, 49(32):204–209.

de Sousa, A. C. C., Sousa, F. S. C., and Bo, A. P. L. (2019). Simulation of the assistance of passive knee orthoses in FES cycling *. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3811–3814, Berlin, Germany. IEEE.

Hunt, K. J. (2005). *Control Systems for Function Restoration, Exercise, Fitness and Health in Spinal Cord Injury*. Thesis, University of Glasgow.