Helathy and epileptic hippocampus simulator

by Amélie Aussel, Radu Ranta and Laure Buhry
contact : aaussel@bu.edu

This PDF contains useful information to start using the simulator.

1) References :

The hippocampus model used for this simulator has been described extensively in the PhD thesis manuscript by Amélie Aussel (available on the « these.fr » website), as well as in the article [Aussel et al. 2018] (for a previous version of the model). Another journal article has been submitted to the European Journal of Neuroscience.
The simulator is based on the Brian2 libraries for Python (see [Stimberg et al. 2019]).

2) Requirements :

This simulator was developed using Python **3.6** with the following packages :
- Brian2 version **2.0.2.1**
- Numpy version **1.12.1**

The simulator was developed in a Linux environment and later tested on a Windows 10 operating system.
The complete specifications used for the Windows environment can be found in the file : **`environment_hipp_sim.yaml`**
A similar virtual environment can therefore be created by using the command : **`conda env create -f environment_hipp_sim.yaml`** (and later be activated using the command **`conda activate hipp_sim`** before starting simulations).

3) User interface :

Two files are provided to set up and start simulations easily :
`user_interface_simplified.py` and **`user_interface_extended.py`.**
Executing any of these files opens a Tkinter window with diferent tabs enabling to choose the parameters to be used in the simulation, as well as the output variables to save or plot.

The **`user_interface_simplified.py`** file regroups some of the parameter values so that the user only has to choose between « slow-wave sleep » and « wakefulness » conditions, and it also includes different epileptogenic abnormalities.

The **`user_interface_extended.py`** file enables to choose all neuron numbers and connectivity settings separately so as to have a finer control over the model.

In both interfaces, the input applied to the hippocampal network can be either square-wave currents or Poisson synaptic inputs derived from user provided .txt fles. Such .txt file should be presented with one value on each line.

When a simulation is run, a folder is created with a name of the form « **`results_date_time`** », which contains simulations results along with a « **parameter.txt** » file with the parameter choices. The LFP signal generated by the network is also always stored as a .txt file in this folder.

4) Parallel processing :

The file **`parallel_processing.py`** can be used to setup multiple simulations and run them in parallel (using the **joblib** library).

A user interface is not provided at the moment. Instead, the choice of parameters, inputs and outputs of the model should be entered manually in the .py file. The number of core to be used for simulation can also be specified (by default, all cpu cores will be used).

For each parameter, all the values to be used should be entered as a list (details are available as comments next to each parameter name in the file). The complete set of simulations will then be the cartesian product of all the list of parameter values provided.

The output to be recorded and saved should be the same for all the simulations.

5) Other fles in the simulator :

The following files are used to run the model :

- **`global_vars_and_eqs.py`** : regroups the diferential equations defining neuron dynamics as well as other global constants
- **`topology.py`** : setups the spatial position of neurons
- **`preparation.py`** : creates neuron groups and synapses, using the two previous files
- **`apply_input.py`** : defines the input function to be used to stimulate the network
- **`annex_functions.py`** : regroups other useful functions, such as those used to save simulation outputs as txt fles
- **`single_process.py`** : runs one simulation from a given set of parameters, using all the previous files

The parallel_processing.py file also uses the set_vars_and_process.py file to setup and launch a single simulation from the set of simulations tu run.

References :

- Aussel, A., Buhry, L., Tyvaert, L. et al. “A detailed anatomical and mathematical model of the hippocampal formation for the generation of sharp-wave ripples and theta-nested gamma oscillations”, J Comput Neurosci (2018) 45: 207. https://doi.org/10.1007/s10827-018-0704-x
- Aussel Amélie, « Computational modeling of healthy and epileptic hippocampal oscillations », PhD thesis manuscript, 2019.
- Stimberg, M, Brette, R, Goodman, DFM. “Brian 2, an Intuitive and Efcient Neural Simulator.” eLife 8 (2019): e47314. doi: 10.7554/eLife.47314.