Challenging the point neuron dogma: FS interneurons as 2 stage integrators

INFORMATION ABOUT THE SIMULATIONS AND CODES.

Dear Colleagues,

In this .pdf file you can find detailed descriptions on the codes used in this work and information about how to run the simulations.

Hope you will find them useful!

Best wishes,

Alexandra Tzilivaki

In case you have any questions do not hesitate to contact me at : alexandra.tzilivaki@charite.de (Twitter: @ATzilivaki)

For questions regarding the network model please also contact Dr. George Kastellakis at: gkastel@gmail.com

1) Folder Description

a. ANN

This folder contains the codes and data used to model the Artificial Neural Network models.

Related to Figure 6, Supplementary Figure 10 and Table 1 of the paper. For further details, please read the Methods section of the paper.

Requirements for running the codes:

Python 3.6.8

Pytorch 1.0.1.post2 (pytorch.org)

Contents:

inn_cell.py	This is the main code that simulates the ANNs and generates the output data.
plot_cells.py	This code generates the regression analysis results and the figures
CSVDATA	This folder contains the data (input and output) used to train the ANNs
data	The outputs generated by running the inn_cell.py will be saved in this folder.
run_cells.sh	This bash script runs the inn_cell.py code and enables the user to choose which of the 4 ANN formalisms (bimodal nonlinear, linear, only supralinear, only sublinear) wish to run and choose the number of epochs and seed

How to run (open a terminal and type):

> conda activate

>sh run_cells.py

>python plot_cells.py

b. Network Model:

This folder contains the network simulator that simulates memory engram formation

in a population consisting of excitatory and inhibitory neurons with independent

dendritic subunits.

Related to Figure 7 and Supplementary Figure 11 of the paper.

For further details, please read the Methods section of the paper as well as the Kastellakis, G., Silva, A. J., & Poirazi, P. (2016). Linking memories across time via neuronal and dendritic overlaps in model neurons with active dendrites. Cell reports, 17(6), 1491-1504.

Requirements for running the codes:

gcc 4.4.7

python 2.7

GNU Make 3.81

Contents/Directory layout:

data/	Contains simulation output data. Used to generate figures
src/	Contains the implementation of the simulator. Specifically:
src/lamodel.cpp	Main simulator entry point with command line option parsing
src/constructs.h	Data structure definitions
src/constructs.cpp	Implementation file of simulation dynamics / connectivity and plasticity
src/tests.cpp	Unit tests
figs/	Figure-generating python scripts (requires the simulator output data)
figs/engrams.py	Generates main text figure and the .txt files for the figure data
figs/supl.py	Generates supplemental figure
run_simulations.sh	Script to run all simulations serially
submit_lamodel.sh	Submission script used to run the simulations in a PBS compatible cluster (not used by default)

How to run (open a terminal and type):

(To compile the simulator and generate data:)

> make -C src clean all

> sh run_simulations.sh

(To generate figures)

> cd figs

> python engrams.py

> python supl.py

(To run unit tests:)

> ./tests

c. Multi-compartmental Biophysical Models

This folder contains the codes and data used to model Multi-compartmental biophysical models of the Fast Spiking Basket cells.

Related to Figures 2, 3 and 4 and Supplementary Figures 1-9 and 12 of the paper. For further details, please read the Methods section of the paper.

Requirements for running the codes:

NEURON 7.4 version

MATLAB

For more help on downloading and running the NEURON codes (.hoc files), please read here: https://senselab.med.yale.edu/ModelDB/NEURON_DwnldGuide.cshtml

Sub-folders Description:

- **bash_templates**

basic_graphics.hoc	This file creates graphs while running NEURON simulations.
current_balance_fs.hoc	This file sets the resting membrane potential to -68 mV.

- **mechanism**

This file contains all the .mod files used to build the multicompartmetal models. For more information about the .mod files please read the methods of the paper.

*** IMPORTANT***

After downloading the "Multicompartmental_Biophysical _models file, a successful compilation of the .mod files is needed in order to run the codes in the experiment folder.

How to compile the .mod files (please open a terminal and type:)

>cd mechanism

>nrnivmodl

- experiment

Contents:

disperse.hoc	This files simulates synapses in a randomly disperse pattern in the dendrites (see also in the Methods of the paper: disperse synaptic allocation.)
disperse	Executable file for running the disperse.hoc code
in_branch.hoc	This files simulates the grouped synapses in a few randomly selected dendritic branches. (see also in the Methods of the paper: Grouped synaptic allocation.)
in_branch	Executable file for running the in_branch.hoc code
Model_validation.hoc	This files is used to apply multiple current-clamp configurations to the models for validation tests.
model_validation	Executable file for running the Model_validation.hoc code
IO.hoc	This file is used to activate synapses on each of the dendrites respectively.
io	Executable file for running the IO.hoc code
gap.hoc	This file is used to implement gap junctions in the model cells.
vecstimgap.hoc	This file is used to implement a firing frequency ~ 30 Hz to the presynaptic cell
experiment_with_gap.hoc	This file simulates gap junctions and records dendritic responses.
rungap	Executable file for running the experiment_with_gap.hoc code
Vecstim.hoc	This file simulates synaptic input that lead to ~ 3 Hz (background activity)
Dend_InputResistance.hoc	This code simulate current clamp on the dendrites to calculate input resistance.
dendinputresistance	Executable file for running the Dend_InputResistance.hoc code
SynCurrents_Validation.hoc	This file simulates multiple voltage clamp configurations for validating the synaptic currents of the model cells.
Syncurrents	Executable file for running the SynCurrents_Validation.hoc code
PFCtemplate.hoc	Template with the membrane properties used for the 3 cortical FS BC reconstructions.

tempSomogyi1.hoc	Template with the membrane properties used for the Hippocampal Somogyi_1.hoc reconstruction.
tempSomogyi23.hoc	Template with the membrane properties used for the Hippocampal Somogyi_2.hoc and Somogyi_3.hoc reconstructions.
tempSomogyi45.hoc	Template with the membrane properties used for the Hippocampal Somogyi_4.hoc and Somogyi_5.hoc reconstructions.
Model_reconstructions	This folder contains all the morphological FS BCs reconstructions, used in this study.

If you wish to run a particular protocol (e.g the disperse.hoc) please open a terminal and type:

>cd experiment

>./disperse

- Figure 2

This folder contains the datasets (in the .mat files) and the script used to generate Figure 2.

- Figure 3

This folder contains the datasets (in the .mat files) and the script used to generate Figure 3.

- Figure 4

This folder contains the datasets (in the .mat files) and the script used to generate Figure 3.