

Overview

This is a NEURON compartmental model designed to validate experimental data showing that mGluRIII activation reduces proximal and distal inhibition onto CA1 pyramidal cells (PCs) from parvalbumin and somatostatin interneurons (PV- and SST-INs, respectively). The model is implemented using the NEURON software (v8.2.0), developed in Visual Studio Code with Python (v3.10.14), and runs using the 3D morphology of a biocytin filled CA1-PCs previously developed by the Sciemmi lab (PMID: 33053337). The spatial distribution of PV- and SST-inputs onto CA1-PCs is set using NRN-EZ (v1.1.7; PMID: 36627356) PV-inputs are located on the soma and $<50\text{ }\mu\text{m}$ away from it on apical dendrites. SST-inputs are located at a distance of $>200\text{ }\mu\text{m}$ away from the soma, on apical dendrites. *First*, the model aims to reproduce voltage escape errors that occur when performing somatic voltage clamp recordings from CA1-PCs (folder **Voltage escape**). This is done by introducing a passive conductance along the dendrites, which becomes larger at increasing distance from the soma. To set g_{pas} , we randomly distribute one inhibitory synaptic input along the soma and apical dendrites of the CA1-PC. We measure the attenuation ratio for each event (i.e., local/somatic amplitude), and adjust g_{pas} so that the space dependency of the attenuation ratio matches with the one obtained by using dendritic patch-clamp recordings (PMID: 18552844) and collected in prior computational work (PMID: 30835719). *Second*, the model is used to set the synaptic weight of inhibitory inputs onto CA1-PCs based on somatic voltage clamp recordings of mIPSCs from CA1-PCs in our own experiments (folder **Set I-weight from mIPSC**). *Third*, we reproduce the effect of mGluRIII activation on IPSCs evoked by optogenetic stimulation of PV- and SST-INs (oIPSCs; folders **Effect of DHK on PV inhibition** and **Effect of DHK on SST inhibition**).

Generate a location file for inhibitory synaptic inputs using NRN-EZ

Files containing information about the spatial distribution of inhibitory synapses are generated using the software NRN-EZ (<https://github.com/sciemmi/NRN-EZ>). In the NRN-EZ user interface, set the input parameters as follows:

Left Panel

1. Click on 'Browse' to load the morphology file of the CA1-PC (.swc file).
2. Set the output path for the destination folder using the "Set Path" field.
3. Set the value of "Run" to 1. This can be changed to a different number when generating multiple simulations with inputs distributed over the same range of distances from the soma.

Middle Panel

1. From the drop-down option in the "Module" section select the module entry "Synaptic Input".
2. Assign a name to your module (e.g., IPSC) and set the "Weight" to "Single". This allows the synaptic weight to be the same for all inputs.
3. Set values for the GABA IPSC reversal potential ("E"), rise time ("Tau1"), and exponential decay time ("Tau2"). In our case, we used the following values for each simulation:

Simulation Name and Number of Inputs	E _{GABA}	Tau1	Tau2
Voltage escape 100 inputs in soma and apical dendrite	-70 mV	Case I: 0.4 ms Case II: 6 ms Case III: 3 ms	Case I: 4 ms Case II: 18 ms Case III: 30 ms
Set I-weight from mIPSCs 100 inputs on soma and apical dendrite (<200 μ m from the soma)	-70 mV	1.5 ms	20 ms
Effect of DHK on PV inhibition Ctrl: 40 inputs <50 μ m from soma DHK: 30 inputs <50 μ m from soma	-70 mV	Ctrl: 2 ms DHK: 2 ms	Ctrl: 22 ms DHK: 25 ms
Effect of DHK on SST inhibition Ctrl: 332 inputs >200 μ m from soma DHK: 260 inputs >200 μ m from soma	-70 mV	Ctrl: 4 ms DHK: 4 ms	Ctrl: 44 ms DHK: 46 ms

4. Set the input location to “Multiple Uniform” to randomly distribute inputs on the dendrites of the CA1-PC. We used the following “Mean” and “S.D.” values:

a. **Voltage escape:**

“Segment Number” = 0 of “Soma”

“Mean” = 250 μ m

“S.D.” = 250 μ m

These settings allow to distribute inputs across the soma and the entire Apical Dendritic tree, which extends up to ~500 μ m from the soma. This requires setting the Location Limits in the Right Panel (see below).

b. **Set I-weight from mIPSCs:**

“Segment Number” = 0 of “Soma”

“Mean” = 100 μ m

“S.D.” = 100 μ m

These settings allow to distribute inputs on the soma and within the most proximal 200 μ m of the apical dendrite. This requires setting the Location Limits in the Right Panel (see below).

c. **Effect of DHK on PV inhibition:**

“Segment Number” = 0 of “Soma”

“Mean” = 30 μ m

“S.D.” = 20 μ m

These settings allow distributing PV-inputs on the soma and <50 μ m away from the soma, on the apical dendrite. This requires setting the Location Limits in the Right Panel (see below).

d. **Effect of DHK on SST inhibition:**

“Segment Number” = 319 of “Apical”

“Mean” = 0 μ m

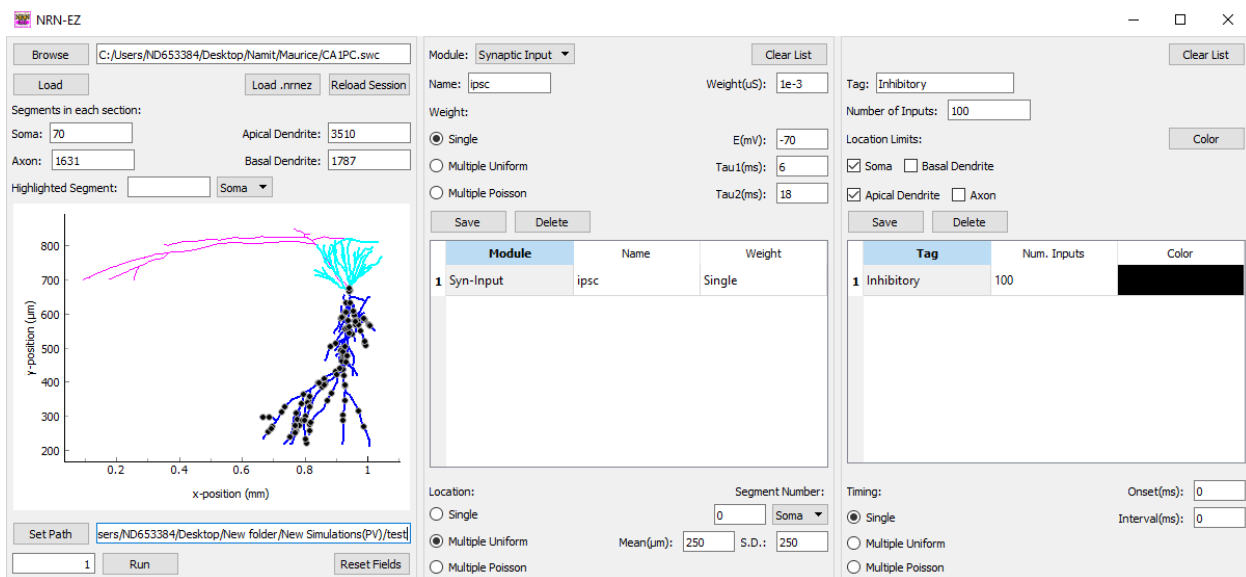
“S.D.” = 165 μm

We calculated the distance of each segment of the apical section from the soma. The “Apical” “Segment Number” 319, located ~360 μm away from the soma, was set as our reference point to distribute SST-inputs >200 μm from the soma, in the apical dendrite. This simulation requires setting the Location Limits in the Right Panel (see below).

Right Panel

1. Set the “Location limits” to identify the compartments with inhibitory inputs. In our simulations, we selected the following settings:
 - a. **Voltage escape:**
Soma, Apical Dendrite
 - b. **Set I-weight from mIPSCs:**
Soma, Apical Dendrite
 - c. **Effect of DHK on PV inhibition:**
Soma, Apical Dendrite
 - d. **Effect of DHK on PV inhibition:**
Apical Dendrite
2. Assign a “Tag” to the inputs (e.g., “Inhibitory”), and set the “Onset time” (100 ms), and “Interval” (0 ms). The “Number of inputs” are elected as follows:
 - a. **Voltage escape:**
100 inputs
 - b. **Set I-weight from mIPSCs:**
100 inputs
 - c. **Effect of DHK on PV inhibition:**
Ctrl = 40 inputs; DHK = 30 inputs
 - d. **Effect of DHK on SST inhibition:**
Ctrl = 332 inputs; DHK = 260 inputs
3. Set “Timing” to “Single”.

Finally, click “Run” in the bottom left panel. This creates a folder called “run_1” within the main simulation folder (e.g., nrnez_2024_08_01_10_02_01). The “run_1” folder contains all files generated by NRN-EZ, including the synapse location, onset time, weight, a morphology .nrn file, and a .mod file that can be used to adjust the biophysical properties of the neuron. An example of the compiled NRN-EZ interface window is shown below:



Run the NEURON Model Using VS Code

To run each simulation, follow these steps:

1. Open the .ipynb file in VS Code
2. In the header of the .ipynb file, set the following variables:
 - “output_file_path”: This is the path to the csv output file containing the time course of the mIPSC/oIPSC.
 - “h.load_file”: This is used to set the path to the “CA1PC.nrn” file, in the “run_1” folder created by NRN-EZ
 - “syn_loc_file”: This is the path to the syn_loc.dat file containing the location of inhibitory inputs, created by NRN-EZ in “run_1” (e.g., Inhibitory)

```
# Save the average soma current to a CSV file
output_file_path = 'Enter a destination path for your .csv output file here'

# Function to calculate distance from soma
def calculate_distance(section):
    h.distance(sec=h.soma[0]) # Set soma as reference
    return h.distance(0.5, sec=section)

# Function to run a single simulation and return the somatic current
def run_simulation():
    # Load morphology and hoc files
    h.load_file('Enter a path to your "CA1PC.nrn" morphology file here')

    # Define file path for the NRN-EZ location file
    syn_loc_file = 'Enter a path to your "syn_loc.dat" file here'
```

3. Click “Run All” from the main toolbar of the VSCode user interface.

4. The code generates an output .csv file (also displayed as a graph) and prints a summary of the amplitude and kinetics of the mIPSCs/oIPSCs. When applicable, experimentally measured amplitude and kinetics values for mIPSCs/oIPSCs are reported as a reference at the end of each code. For the **Effect of DHK on PV inhibition** and **Effect of DHK on SST inhibition** the variable called “selected_synapses” represents the number of randomly selected synapses that allow reproducing the experimental oIPSCs in Ctrl or DHK.

Contributors

The model was created by Namit Dwivedi (namitdwivedi08@gmail.com), conceptualized and supervised by Dr. Annalisa Scimemi (scimemia@gmail.com or ascimemi@albany.edu). This work was funded by the NIH grant R56NS129556.