**MODELING A TWO COMPARTMENT FAST SPIKING INTERNEURON**

**Problem Statement**: Design an interneuron with specified passive and firing properties.

You are to model an interneuron reported in [Kim et al. (2013)](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3761046/). For this, you will use the package NEURON. As a first step, peruse the paper by Kim et al. (2013) that has a complete description of the model and make sure you understand all the biological properties modeled.

WHERE DO YOU BEGIN: First run the files passive.hoc and then spiking neuron model in ‘Getting started with NEURON’.

THEN, you can start with the file named ‘main.hoc’ (or any name you want to provide) which should make a cell from a cell-template file(interneuron\_template.hoc), provide it with a current injection input and run it. Note that the interneuron\_template.hoc uses several .mod files of appropriate channels from that folder. Note that you will have only three ‘.hoc’ files, interneuron\_template.hoc, graphic\_library.hoc and main.hoc’, which call other ‘.mod’ files provided to you in the “Interneuron\_Startup Files” folder.

**What properties need to be matched?**

1. Passive properties.
2. Firing properties:

(i) responses to various current injection values by varying only kD, and

(ii) the frequency-current response (FIR) curve for the neuron.

**Using Real Cell Traces to Match Properties**

1. Fill in the pages before the appendix to document your passive and firing properties
2. Graphs for observed interneurons have been provided for you to match. Matlab code and data export is also provided.
   1. Current was injected at 30 Pico amp intervals starting at -60 excluding 0. (-60, -30, 30, 60, 90…)
   2. Document the conductances and V\_rest for the modeled neuron
   3. Calculate the time constant of your modeled neuron based on calculations from the observed neuron. Try to match the properties as close as possible.
   4. Calculate the input resistance of your modeled neuron by injecting it with -60pA of current like the observed neuron

Steps you can follow:

1. Insert current channels nainter and kdrinter into interneuron\_template.hoc (given to you in the “Interneuron\_Startup Files” folder (see Appendix A).
2. Then make a main.hoc file that loads interneuron\_template.hoc into it (see Appendix B).
3. Assign conductances as gna\_nas = x.xxx, gkdr\_kdr = x.xxx. The default unit is Siemens/cm2. Use plotting code from the graphics code as appropriate (see Appendix C to see how they are used)
4. Try different gna\_nas and gkdr\_kdr values (within the ranges provided), and find the values that can make a cell spike.
5. Observe the frequency and amplitude for the different parameter sets.
6. Now insert kD into interneuron\_template.hoc if necessary to match all the properties required.

Suggestions for model development

* NEURON models the compartment membrane as a cylindrical surface without top and bottom caps In the template file provided to you, we use L=15 µm, and diam=15 µm for the soma and L=150 µm, and diam=10 µm for the dendrite.
* The default capacitance of the model is x µm/cm2
* Use the following reversal potential values:

L

* ELeak = - 70 mV
* Ena = 45 mV
* Ek = - 80 mV

diam

* Initialize the membrane potential to xxx mV.
* Membrane capacitance, channel conductances and output currents are provided per unit area in NEURON, i.e., in the units of S/cm2, and mA/cm2, respectively. It takes care of converting values internally.
* The default units for current injection and synaptic currents are nA.
* Useful links:

NEURON website: <http://www.neuron.yale.edu/neuron/node/47>

NEURON official forum: <http://www.neuron.yale.edu/phpBB2/viewforum.php?f=15>

NEURON Course Hand-outs <http://www.neuron.yale.edu/neuron/static/courses/2008/course/handson.html>

Programmer's Reference <http://www.neuron.yale.edu/neuron/static/docs/help/quick_reference.html>

**Mod files (.mod) for the following currents are provided to you:**

* nainter.mod Fast spike generating Na+ current channel
* kdrinter.mod Delayed rectifier K+ current channel
* leakinter.mod Leak current

**Name**: PV Interneuron

**Original source:** (Kim, Pare, and Nair 2013) For RMP

**Compartments**: Soma: 1; Dendrites: 1

**Original** **Soma/Dendrite Diameters:** 15μm / 10μm – adjusted, see below

**Original Soma/Dendrite Lengths:** 15μm / 150μm – adjusted, see below

**Original** **Conductances – unchanged in model**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Soma** | |  | **Dendrite** | |
| **Channel** | **Gmax(S/cm2)** | **Channel** | **Gmax(S/cm2)** |
| Leak |  | Leak |  |
| Na |  | Na |  |
| K |  | K |  |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Comparison of passive properties of experimental (from Madhu) and model PV neuron:**

|  |  |
| --- | --- |
| **1. V\_rest = \_70\_ mV**  **2. Calculation of time constant:**  Start inject: 100ms / -65mV Final Value: ~ -74.81mV Difference: -4.81 ms | 63.2% = 3.04 ms| -70 - 3.04 =  - 73.04 Time at - 73.04: 116.3ms τ = 116.3-100  **τ = 16.3 ms**  **3. Input Resistance**  ΔV/ΔI = ( -70 – (-74.8) )/( 0 – (-60) )  = 4.81mV / 60pA   = .00481 V/.00000000006 A = 80166666.667 Ω **R\_in = 80.67 MΩ** | **1. V\_rest = \_\_\_\_ mV**  **2. Calculation of time constant:**  Start inject: \_\_\_ms / \_\_\_mV Final Value: ~ \_\_\_\_mV Difference: \_\_\_\_ ms | 63.2% = \_\_\_\_ ms| \_\_\_\_ - \_\_\_\_ = \_\_\_\_\_ Time at \_\_\_\_\_ : \_\_\_\_\_ ms τ = \_\_\_\_\_-\_\_\_\_\_  **τ = \_\_\_\_\_ ms**  **3. Input Resistance**  ΔV/ΔI = ( \_\_\_\_ – (\_\_\_\_) )/( 0 – (-60) )  = \_\_\_\_mV / 60pA   = \_\_\_\_\_ **V**/.00000000006 A = \_\_\_\_\_\_\_\_\_\_\_\_\_ Ω **R\_in = \_\_\_\_\_ MΩ** |
| **Experimental Cell Provided by Madhu** | **Model after Adjustments** |

**Cell Model Soma/Dendrite Diameter after Adjustments**: Scaled by a factor of 2.158  
Soma Length: 32.37 μm  
Dendrite Length: 320.37 μm  
Soma Diameter: 32.37 μm   
Dendrite Diameters: 21.58 μm

**Appendix A:** Edit the file “interneuron\_template.hoc” given to you in the startup files, and insert channels (suggested ranges for parameters are provided) at the right locations in that file. Don’t change the other statements in that file.

// In the interneuron\_template.hoc file, add the following as appropriate

insert leakinter

insert nainter

insert kdrinter

Indicate the reversal potentials after inserting all channels.

el\_leakinter = -70 // (mV)

**Appendix B: Make a file** Main.hoc**; define a cell using the template you made, and then provide it with current injection (Note: some commands may be missing!)**

**{load\_file("nrngui.hoc")}**

**{load\_file("graphic\_library.hoc")}**

**{load\_file("interneuron\_template.hoc")}**

**// main.hoc**

**tstop = 500**

**objref CellA // declare neuron object**

**CellA = new InterneuronCell() // build neuron from template**

**// for stimulating it using current injection**

**objref ccl**

**CellA.soma ccl = new IClamp(.5)**

**ccl.del = 50**

**ccl.dur = 100**

**ccl.amp = 1**

**Appendix C: Using show\_output function in** Main.py membrane voltage can be plotted.

**allInOneBoxCurrents("CellA", "soma")**

**allInOneBoxSpiking("CellA", CellA, ccl)**

**Appendix D: Ranges for current conductances**

soma nainter = .025 - .05

soma kdrinter = .005 - .015

dend nainter = .010 - .020

dend kdrinter = .001 - .005