2-IKV-188: Assignment 2 (worth 20 marks)

Project with NEURON

The goal of this assignment is to investigate the behaviour of the model of a small neural network. You will creatively extend the programme code called **sthC3.hoc**.

The program **sthC3.hoc** simulates four neurons from the rat subthalamic nucleus. Each model neuron is represented by the soma plus two dendritic trees A and B as described in the NEURON tutorial part C. We connected **SThcells[1]** to the dendritic branch 7 of **treeA** of the neuron **SThcells[0]** with one exponential synapse using the command:

```
SThcells[0].treeA[7] syn[0] = new ExpSyn(0)
```

As only the neuron with index *i* = 1 is stimulated with an injected electric current from 100 to 200 ms, and there is only one synapse from neuron 1 to neuron 0, then only the neuron SThcells[1] will be generating spikes and there will be a series of EPSPs at the soma of neuron 0.

/***

Note on excitatory and inhibitory synapses

In the ExpSyn synapse, which we use in **sthC3.hoc**, the weight parameter is the peak amplitude of the synaptic conductance. This has two implications. First, the weight parameter for a conductance change synapse should be always >= 0. Second, whether the synapse is inhibitory or excitatory depends on whether the synaptic reversal potential lies above or below spike threshold. When we create a new instance of ExpSyn, we introduce these variables (with certain default values):

```
syn.tau // decay time constant in ms
syn.e  // reversal potential in mV
syn.i // synaptic current in nA
```

The default value for **syn.e** is for an excitatory synapse. So, if we want an inhibitory synapse we have to set a new value for its reversal potential, e.g., **syn.e ≅ -60** should work.

***/

In your written report, describe the parameters (number of synapses, their location, etc.) and the results of simulations of the following tasks and document them with the corresponding graphs.

Task 1: more synapses

In the code **`sthC3.hoc`**, move the injected current from the soma of neuron with index *i* = 1 to the soma of neuron with index *i* = 0 so that only neuron 0 is stimulated and no other neuron receives an external stimulation. Change the direction of synaptic connection so that it goes from neuron 0 to neuron 1, i.e.

```
SThcells[1].treeA[7] syn[0] = new ExpSyn(0)
```

Run the simulation to see whether the neuron 0 fires spikes and neuron 1 generates series of EPSPs. Then, connect the neuron 0 with neuron 1 with nine more synapses impinging at different branches of tree A and B of neuron 1. If you use a loop, then the index [k] in the loop containing the following command will run from 0 to 9.

```
SThcells[0].soma   SThcells[1].nclist.append(new   NetCon(&v(0),
syn[k], -20, 1, 0.5))
```

Change the values of parameters of these synapses (taus, weights, delays) to achieve that neuron 1 also produces one or more spikes after neuron 0 is injected with an electric current. Choose delay $0 < \Delta t <= 10$ ms. You can choose the target branches in trees A and B at random.

Task 2: network with *feedforward* connections

Now you should have 10 synapses from neuron 0 → neuron 1. Add 10 synapses from neuron 1 → neuron 2 and 10 synapses from neuron 2 → neuron 3. To make the model more realistic use different parameters, i.e., different weights, delays, and locations for each synapse you create.
Recall, so far we have only created 10 objectvars to refer to synaptic objects in the original code **`sthC3.hoc`**; so you need to increase the maxsyn parameter.

Try playing with the connection parameters (taus, delays, weights, thresholds) so that **each** neuron in the network generates spikes in response to stimulation of neuron with index $i = 0$. You may want to change also the parameters of the IClamp current injection into the neuron 0, i.e., its amplitude and duration. Also, it may be necessary to make the total run time longer than 300 ms, for instance tstop = 1000 ms.

Task 3: network with excitatory *feedback* connections

Our small network has only feedforward connections, i.e., 0 → 1 → 2 → 3. In addition to these feedforward connections from the previous task, introduce 10 feedback synapses from 3 → 2, ten feedback synapses 2 → 1, and ten feedback synapses 1 → 0, and see what happens. You may want to play with the synaptic parameter values (including the number of synapses themselves) to obtain various regimes of network spiking activity. What can you conclude about the network spiking activity if we have only excitatory synapses in it?

Task 4: network with inhibitory feedback connections

Change all the *feedback* synapses in the previous network to inhibitory synapses, thus simulating the intermediate interneurons. Will the spiking activity in the network cease? Under which condition will it cease? How, if at all, can you control the level and pattern of spiking in the network so that it does not cease entirely? Again, you will have to play with synaptic parameters and see how the network activity is affected. Optional: If you want, you can also increase the number of neurons in your network, but then you would have to make more synapses. What can you conclude about the role of inhibition in the network?

Programmer's reference: http://www.neuron.yale.edu/neuron/static/docs/help/contents.html

IMPORTANT: E-mail the codes you used and your reports to lubica@ii.fmph.uniba.sk. If you need an extension, for whatever reason, please email or speak to me about it. I will subtract 10% of total marks for each working day late **without prior arrangement** with me.