Spike-triggered synaptic transmission homework

Create a 1 section model cell called 'soma' with

- surface area 100 um2
- nseg 1
- pas channels with e -65 mV and g 5e-5 S/cm2 (membrane time constant 20 ms)

Attach an ExpSyn with tau 3 ms, e 0 mV to soma(0.5).

Drive the ExpSyn with events from a NetStim with

- interval 10 ms
- number 1
- start 5 ms
- noise 0

Set the NetCon's delay to 1 ms.

Homework *continued*

Run a simulation for 100 ms. How big must the NetCon's weight[0] be to elicit a 1 mV EPSP at soma(0.5)? (2 significant figures)

Now uninsert pas and insert hh. What is the minimum positive weight[0] that triggers a spike?

My implementation strategy: exploit the fact that this is the same task applied to different model cells.

Writing code that works with different model cells

`pascell.py`

- defines passive Cell class

`hhcell.py`

- defines Cell class that has hh

`netex1.py`

- imports specified Cell class and creates an instance
- sets up
  - cell-class-specific parameters
  - common instrumentation
    (signal sources, Vector recording)
  - simulation flow control and GUI (RunControl panel and graphs via `rig.ses`)

Homework *continued*

Extra credit:

Using the model with hh, adjust weight[0] to a value that elicits a 1 mV EPSP.

Next change the NetStim's interval to 1 ms, number 1e9, and noise to 1. Run 100 simulations that include 1000 ms of synaptic input and record the number of spikes per run. Generate a histogram of number of spikes per run (binwidth = 1).

Strategy:

- reuse netex1.py, hhcell.py, and rig.ses
- for new task create netex2.py based on netex1.py