**Lab 4: Modeling neural recordings and stimulation**

# Introduction

In Labs 2 and 3, we used specialized software to model neural activity and current spread. To study neural interfaces, such as neural recordings and stimulation, these two types of models must be combined. By exporting currents from NEURON and K matrices from COMSOL, we can both incorporate neural currents and electrostatic models into a single model. At the end of this lab, you should be familiar with both analytical and computational approaches to solving this problem.

# Software

This lab must be completed using MATLAB and NEURON. (You will also need a COMSOL output from Lab 3.)

# Part 1) Analytical model of neural recording

An important question to ask about neural recordings is, “In what conditions should we be able to record an action potential?” How does distance from the neuron affect the recording? What about neuron size? Here, we’ll create a simple model of neural recording outside of a single neuron.

Recall Cindy’s favorite equation:

By applying this equation to currents exported from NEURON, we can predict the voltage trace recorded by an electrode near the simulated neuron.

For this lab, we will be using the files currents\_big and currents\_small. These files are in the same format as the NEURON outputs from Lab 2.

1. Load currents\_big.mat into MATLAB. This file contains currents (in amps) from a NEURON simulation of an action potential from a layer V pyramidal neuron (which is often around 20 um in diameter). Each current is associated with a specific XYZ coordinate in space (in micrometers) indicating the part of the neuron that produced the current.
2. The axon hillock is located at the origin, with the soma and dendrite extending along the x-axis. Pick a point 50 um from the axon hillock, extending perpendicularly from the neuron’s axis. (That is, don’t pick a point inside the cell.)
3. Use Cindy’s favorite equation to calculate the voltage reading that would result from the current produced at the origin and picked up by an ideal electrode at your chosen point. Assume . CHECK YOUR UNITS.

Note: It will be helpful to write this step as a function of the form:

voltageTrace = calcVext(currentTrace, currentXYZ, electrodeXYZ)

You will need to reuse this calculation for several parts of this lab.

1. Now do this for every point along the neuron and use superposition of voltage to calculate the net voltage trace your chosen point would record from the whole neuron.
2. Repeat this process for points 100, 200, and 300 um from the axon hillock. Again, make sure your points extend radially away from the axis of the neuron.
3. At what distance would an action potential be no longer detectable above a noise floor?
4. Repeat 2-6 for currents in currents\_small.mat. The currents in this file are closer to what we would expect from a small layer II neuron in cortex.

# Part 2) Simulate electrode recording of multiple neurons

Real-world neural recordings usually don’t occur in noise-free environments with just one neuron in the vicinity. Here, we will extend the model from Part 1 to a multi-neuron environment with tissue injury along with thermal and biological noise.

1. Using the coordinates and currents from currents\_big.mat, randomly distribute 10 neurons within a 100 um cube centered on a recording site. Just the axon hillocks need to be within the cube; don’t worry if part of the dendrite or axon extends beyond. Serendipitously, pyramidal neurons in the cortex naturally tend to point in the same direction, so there’s no need to randomize orientation.
2. Using the waveform from currents\_big.mat, generate a 2-50 Hz bursting pattern for each neuron that is one second long.. Assume in the current files.

Note: This must be programmed in a way such that 1) action potentials from a single neuron do not overlap; and 2) action potentials from multiple neurons do not happen at *exactly* the same time. A handy way to do this is to randomize the time between spikes for each neuron.

1. In the same way as you did in Part 1, calculate the net potential recorded at the electrode, this time accounting for all ten neurons with their variable firing patterns.
2. Add pink noise to the recorded potential, using the following lines of code and scaling it appropriately:

cn = dsp.ColoredNoise('Color','pink','SamplesPerFrame',tTot/dt);

recordedNoise = cn();

Note: Real thermal noise is not uniformly distributed over all frequencies, like white noise, which is why we use pink noise here. For more detail on thermal noise in neural recordings, read Lempka et al, 2011.

1. Oftentimes, electrode insertion results in a “dead zone” immediately around the electrode. Repeat this process, but exclude neurons that are within 50 um of the electrode.
2. Plot the resulting voltage traces with and without the “dead zone” in 2 subplots of the same figure.

# Part 3) Model neural recording with a variable conductivity environment

In Parts 1 and 2, we used an analytical approach to calculate voltage recordings of neural firing. Complex conductivity environments, however, can make the calculation prohibitively complex. Here, we will approach the same problem with a computational approach and compare differences in output.

In Lab 3, Part 2, you generated a matrix of voltages in space that result from a 1 A current at the origin of a variable conductivity environment. We will use it as a K matrix to transform currents at a distance to voltages at the electrode. Recall the relationship described by the K matrix:

1. Extract your K matrix from the COMSOL output into MATLAB. CHECK YOUR UNITS.
2. Using currents and coordinates from currents\_big.mat, place a cell with the hillock 50 um from the origin, in the Y or Z direction.
3. Since the coordinates of the currents and the K matrix do not line up exactly, you will have to interpolate the coordinates of one to match the others. MATLAB’s griddata() function can be helpful for this.
4. Use your K matrix to model the extracellular waveform recorded 50 um from the axon hillock.
5. Compare to your results from Part 1.

# Part 4) Model neural stimulation

In Part 1-3, we focused on simulations of neural recording. We now turn our attention to simulations of neural *stimulation*. To carry out this simulation, we will use the simple 20-um axon you created in Lab 2, Part 2.

1. Using your NEURON model, determine the threshold current necessary to trigger an action potential along the axon.
2. Assume that an electrode tip is 1 mm away from the middle of the axon. Using MATLAB, calculate the external voltage seen by every compartment along the axon, if the electrode were to inject 1 mA of current. Make your axon 5 mm in total length. Plot the results.
3. Using the equation described by Warman, Grill, and Durand (1992), calculate the equivalent intracellular current that comes from the injected extracellular current. Assume .
4. Determine the minimum current necessary to activate an axon 1 mm away.

# Guidelines for Lab Report (on Labs 3 and 4 together)

*Introduction:* The introduction should be one paragraph long summarizing the motivation for electrostatic models, what data they draw upon from past experiments, and a brief summary of everything you will show in this lab report.

*Methods:* From Lab 4, there should be methods paragraphs (and diagrams where necessary) on:

1. Assumptions of the models used
2. How the models were designed
3. The different levels of model complexity

Include the code as an Appendix to your report. Cite sources for any values used in your models.

*Results:* You should include the following in your Results:

1. Plots from big\_currents and small\_currents from Part 1, at distances of 50, 100, 200, 300 um.
2. Plot neural traces from Part 2 (with/without “dead zone”).
3. Plot your modelled voltage waveform from Part 3.
4. Include the minimum current you calculated in Part 4
5. Describe the effects of increasing model complexity and any implications for electrophysiological recordings and deep brain stimulation.

Include all figures produced by MATLAB that could help explain and illustrate your findings.

*Discussion:* Should be 2-3 paragraphs long describing what you could use these models for in the future.

This report will be combined with Lab 3, to create one cohesive report. The report (not including Appendix) should be no longer than 4 pages. Use 12 pt. font and 1.15-1.5 line spacing. If your text is over the 4-page limit with figures, you can move your figures to an appendix section that goes beyond the 4-page limit. However, any text that goes beyond this limit will not be graded, except for figures, figure titles (no captions), and your code.

Please upload your report to Canvas and leave a hard-copy with your GSI in lab. The hard-copy will be graded, so be sure different lines on your plots are distinguishable (using color or different line styles).