**Lab #2 BME218/418 Simulating Single-Neuron Electrophysiology**

For background on action potential characterization, see Hodgkin and Huxley’s Nobel Prize lectures:

<http://www.nobelprize.org/nobel_prizes/medicine/laureates/1963/>

The goal of this lab exercise is to gain an understanding of action potential dynamics by studying the underlying ion-channel dynamics. This goal will be accomplished by creating a computational model of membrane and ion-channel kinetics. We’ll use both basic 1st-order approximations to the derivatives in the differential equation and ultimately more sophisticated solutions that can be implemented with Matlab’s ODE solver.

In class, we discussed the differential equations that describe the kinetics of [Na+] and [K+] channels and the membrane voltage. In this lab, you will use these equations, plus descriptions of other ion channels, to construct a computational model in Matlab that is capable of simulating the response of a few different types of neurons to input currents.



**I. Simulation of a linear RC Circuit**

We’ll start by constructing a model of a basic linear system   
with one state: the RC circuit shown in figure 1. In order to   
simulate this system, you’ll need to set up a differential equation  
that describes how the capacitor voltage (Vm) changes with time.   
Use the forward Euler’s approximation for this differential equation  
 to simulate the step response of this circuit. Vary the size of the time step to demonstrate the weakness of the forward Euler’s approximation in this situation. Next, use the backward Euler’s approximation and convince yourself that the simulation is more robust across a range of time steps. Your simulation is most accurate for a very small time step. Find the largest time step that you can use that doesn’t affect the simulation (keep this code for comparison to later versions). Finally, use Matlab’s ODE45 function to simulate the differential equation using the 4th-order Runge-Kutta algorithm for approximating the derivative. Review the documentation on Matlab's ordinary differential equation (ODE) solvers:   
<http://www.mathworks.com/help/matlab/math/ordinary-differential-equations.html>  
Also – see the tutorial posted on Blackboard (from Bucknell University.)

Figure 1. Schematic of a parallel RC circuit with an input current.

The benefit of ODE45 is the variable time step. Compare the number of time steps that it uses to the number required for the backward Euler’s approximation. The number of steps will vary when you set a “tolerance” for the error using “options.” Also, try using “tic/toc” in Matlab to estimate the time required by the difference algorithms.

**II. Simulation of a Cell Membrane with Constant Channel Conductances**

Set up a simulation with 3 parallel conductances, representing the three ion channels in the Hodgkin-Huxley model. Ultimately, you’ll want to use the ODE45 function, but if you prefer to first set up your code with Euler’s method, that’s fine. As shown in figure 2, an ion-permeable cell membrane can be approximated by a capacitor (the cell membrane) in parallel with several conductors (the ion channels) and voltage sources (the ionic gradients across the cell membrane associated with each ion).

Construct a Matlab script that estimates the voltage across the cell membrane as a function of time in response to an arbitrary set of input current steps. Plot both the input waveform and the output voltage as functions of time.  For now, set the conductance of each channel to a fixed value, e.g. the steady-state value associated with each conductance (we'll worry about voltage sensitive changes in conductance later).



Figure 2. Electrical circuit model of a neuron cell membrane.

(From <http://www.nobelprize.org/nobel_prizes/medicine/laureates/1963/huxley-lecture.pdf>)

**III. Simulation of the Hodgkin-Huxley Neuron Model**

Implement the Hodgkin-Huxley neuron model by incorporating the additional states that describe the voltage-sensitive ion-channel conductances of the neuron. Your function should allow you to input an arbitrary sequence of current steps and should plot input current, membrane voltage, and activation/inactivation probabilities for the m3, n4, and h gates as a function of time. An example of the desired functionality is shown in figure 3.



Figure 3. Example of Hodgkin-Huxley Model Results. The first plot shows the input current (flow into the cell), the second plot shows the cell membrane voltage in response to this input current, and the third shows the activation probabilities for the three types of ion channel present in the model (blue represents the m3 gates, green represents n4 gates, and red represents h gates) Note that in this example the membrane potential is referenced with respect to the cell exterior (resting voltage = -70mV), although your model could be referenced with respect to resting internal voltage (resting voltage = 0mV).

Use your model to answer the following questions.

1. How are stimulus magnitude and duration related with respect to action potential? Can a weak, sustained stimulus produce the same response as a strong, brief stimulus?
2. What happens when a stimulus sufficient to produce an action potential is sustained over a long period of time? Do the inter-spike interval and spike amplitude remain constant? Why?
3. How does interspike interval vary as a function of sustained input current amplitude? Use this relationship to estimate the absolute refractory period for the HH neuron.
4. It is common in neurophysiology experiments to use a neurotoxin produced by the puffer fish, Tetrodotoxin (TTX), to selectively block [Na+] channel conductance. Simulate the effect of TTX application to your model neuron. What happens to the model neural response to an input current? Based on this result, why is exposure to TTX hazardous to most animals?

**IV. How do Novel Ion Channels Alter Neural Function?**

An important application of models for individual neurons is using them to predict how neural behavior is altered by the presence of different types of ion channels. Below are references to two point-neuron models. Recreate one of these models and characterize the unique properties of the simulated neuron. Predict potential implications of this altered neuronal behavior (i.e. how might this neuronal behavior be useful for the system these neurons belong to?). Make sure to include in your lab report figures that demonstrate the neuronal properties you observed.

Note: Due to the highly nonlinear (“stiff”) properties of these models, ode45 may have a difficult time correctly simulating them. Consider switching to a different solver if your solver fails with the message “Warning: Unable to meet integration tolerances without reducing the step size below the smallest value allowed”. Check the documentation on ODE solvers for additional information:

<http://www.mathworks.com/help/matlab/math/ordinary-differential-equations.html>

**For Undergraduate Students:**

*Low Threshold Potassium Channel KV1.1 in the auditory midbrain*– Svirskis, Kotak, Sanes, Rinzel 2004: <http://jn.physiology.org/content/91/6/2465.long>

**For Graduate Students:**

*Cricket Cercal System Neurons* – Eaton, Crook, Cummins, Jacobs 2004:

<http://www.sciencedirect.com/science/article/pii/S0925231204000797>

Note: As printed, the equations for this model contain an error! Being able to identify peculiar behavior in a model is a valuable skill for any engineer, so it is up to you to find and correct the error in this model. You can verify correct model functionality by comparing your simulation results to the ones presented in the paper.

**Lab Report:**  For the Methods section of the report, include a basic description of your approach and a thoroughly commented copy of your code. Imagine posting this code on the Internet for other colleagues and students to use – be sure your comments “step the reader through” each section of the code. Also, the questions at the bottom of page 3 should all be addressed, and illustrated, in your lab report. Finally, include a description and illustrations of your results for Part IV.