PROBLEM SET 1, PART 1

Gain control, modulation, and homeostasis in single neuron models

Contents

1 Gain modulation in single neurons

1.1 Background

Neural systems often need to modulate the activity of specific neurons in order to produce a particular response. Gain modulation is a multiplicative scaling of a neuron's response to a particular input. That is, for a particular stimulus, gain modulation results in a change in the slope of the firing rate versus current (f-I) curve[1] used to describe the neuron.

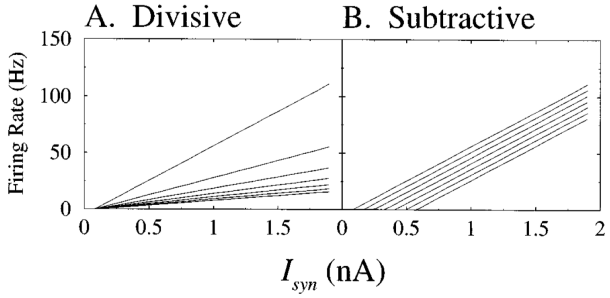

Figure 1: Modulation of a single neuron's response function. (A) Gain modulation is defined as having a divisive (or multiplicative) effect on the firing rate versus current (f-I) curve. (B) A subtractive (or additive) effect does not change the slope of the curve, and therefore does not affect the gain. Figure copied from [3].

Here, you will investigate plausible mechanisms for gain modulation in a simplified model of a neuron, the integrate-and-fire model. The particular mechanisms we will investigate involve (1) shunting inhibition[2] and (2) increased background synaptic input. Shunting inhibition was originally proposed as a plausible mechanism for gain modulation, however, Holt and Koch [3] showed (using a variety of single neuron models) that this type of inhibition only had an additive (or subtractive) effect on the f-I curve of the neuron, not a multiplicative effect. Building off of this work, Chance, Abbott, and Reyes [2] showed that the addition of background synaptic input, when combined with shunting inhibition, could together serve to modulate the gain of a neuron.

To explore these ideas, we will first simulate an integrate-and-fire neuron using Matlab (§1.2). Then, we will modify this simulation in §1.3 to investigate the effect of shunting inhibition on the f-I curve of the neuron. Finally, in §1.4 you will add background synaptic input to the model, to understand how this (together with shunting inhibition) can give rise to gain modulation.

1.2 Integrate-and-fire neuron

To better understand mechanisms of gain control in individual neurons, we will use the leaky integrate-and-fire (LIF) neuron model[3]. This is a model of the soma of a neuron consisting of two electrical elements in parallel:

[1] The f-I curve is a commonly used tool to describe the response of a neuron as a function of the net current input to the neuron.

[2] Shunting inhibition is a particular form of inhibition which involves inhibitory synapses near the soma that, when open, *shunt* excitatory current out of the cell.

[3] This is one of the oldest computational models in neuroscience, first studied by Lapicque in 1907. The popularity of the LIF model persists due to its simplicity, analytical tractability, and computational efficiency.

a capacitor and a resistor (Figure 2). These elements are meant to model the electrical properties of the cell membrane: the capacitor models the membrane's ability to build up charge and the conductance describes leakage of ions across the cell membrane.

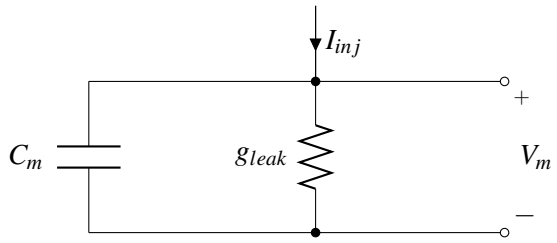

Figure 2: RC circuit model of a neuronal membrane. The membrane is modeled with a capacitor C_m and leak conductance g_{leak} in parallel. The voltage drop across each of these elements is the membrane voltage, V_m. Input current to the cell is modeled as the injeted current I_{inj}.

The equation governing the circuit dynamics of the LIF model is:

$$C_m \frac{dV_m}{dt} = I_{inj} - g_{leak}(V_m - V_{\text{rest}}) \tag{1}$$

To simulate the dynamics of the LIF model, we discretize[4] the above equation in time and express the voltage at some small time step $(t + \Delta t)$ in the future in terms of the current and voltage at time t:

$$C_m \frac{V_m(t+\Delta t) - V_m(t)}{\Delta t} \approx I_{inj}(t) - g_{leak}(V_m(t) - V_{\text{rest}}) \tag{2}$$

$$V_m(t+\Delta t) \approx V_m(t) + \frac{\Delta t}{C_m}\left(I_{inj}(t) - g_{leak}(V_m(t) - V_{\text{rest}})\right) \tag{3}$$

Part 1 First, you will simulate an integrate and fire neuron in Matlab. Use the provided starter code as a guide. Take a look at the function `pulse.m`, which you can use to generate a square current pulse. You will then need to:

1. Fill out the function `integrate.m`, which simulates the leaky integrate-and-fire neuron given an injected current, time step, and cell parameters.
2. Write a script that simulates the LIF neuron with the current pulse, and plot the injected current and the membrane voltage (you can use the provided template, `leaky_integrate_and_fire.m`). You should generate a figure similar to the bottom panel in Figure 3.

1.3 Shunting inhibition

Now that we have our simulation, we will explore the role of shunting inhibition on the f-I curve of the LIF neuron. The f-I curve is a function that plots the firing rate of a neuron against the amplitude of the injected

[4] We are doing what is known as *forward Euler* numerical integration, which is a simple numerical integration algorithm.

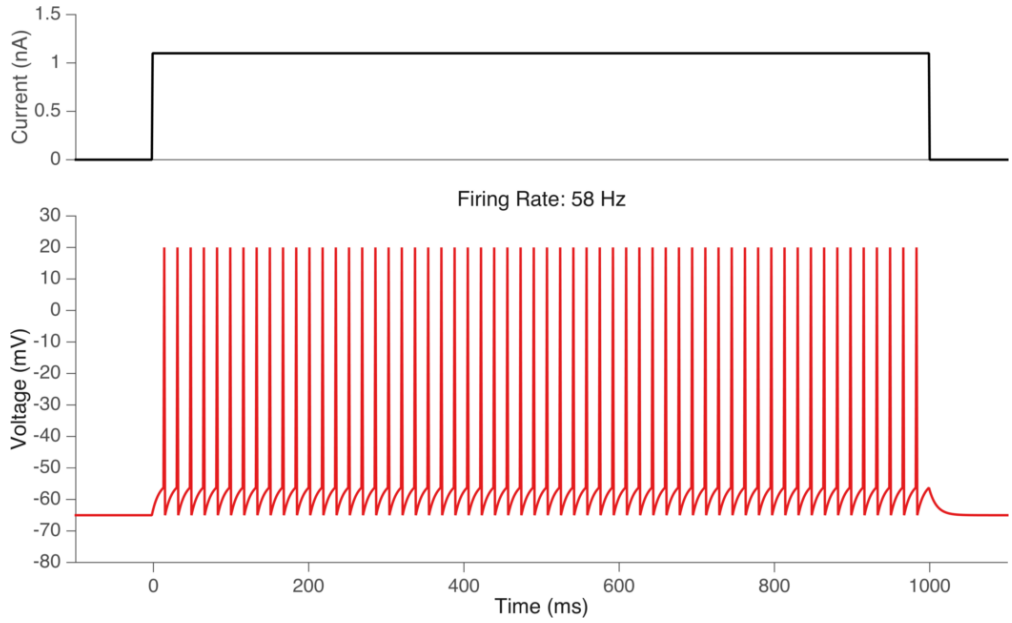

Figure 3: Simulation of a leaky integrate-and-fire neuron in response to a step current pulse.

current. Remember, we are interested in mechanisms for scaling the f-I curve by a multiplicative factor (thus changing the slope of the curve).

Shunting inhibition is a mechanism for inhibiting a neuron by adding ion channels in the cell membrane with a reversal potential close to the resting membrane potential. When the neuron is stimulated, the membrane potential is pulled away from the resting potential, making the driving force on the ion current large and thus shunting current through the ion channels (as opposed to charging the membrane).

We can capture this effect in our integrate-and-fire model simply by modifying the leak conductance of the LIF neuron. The leak conductance has a reversal potential exactly equal to the resting potential, therefore increasing the leak conductance can be thought of as increasing the amount of shunting inhibition.

Part 2 You are given a function, `ficurve.m`, which computes an f-I curve for the LIF neuron by repeatedly stimulating it with different current amplitudes and measured the resulting number of spikes. To investigate how shunting inhibition affects the f-I curve of the neuron, do the following:

1. Write a script that plots the f-I curve for a few different values of the leak conductance, on the same plot. Your figure should look similar to Figure 4.

2. Do the f-I curves show gain scaling (a multiplicative shift) as the amount of shunting inhibition is modified? (Bonus: see if you can derive an analytic expression for the f-I curve. Does this expression agree with your simulation? Shunting inhibition affects this expression in *two* ways, what are they? See Chance, Abbott and Reyes [2] p. 776 for a hint.)

3. Explore how the f-I curve changes as a function of the membrane capacitance. Is changing the membrane capacitance a biophysically plausible mechanism for gain modulation?

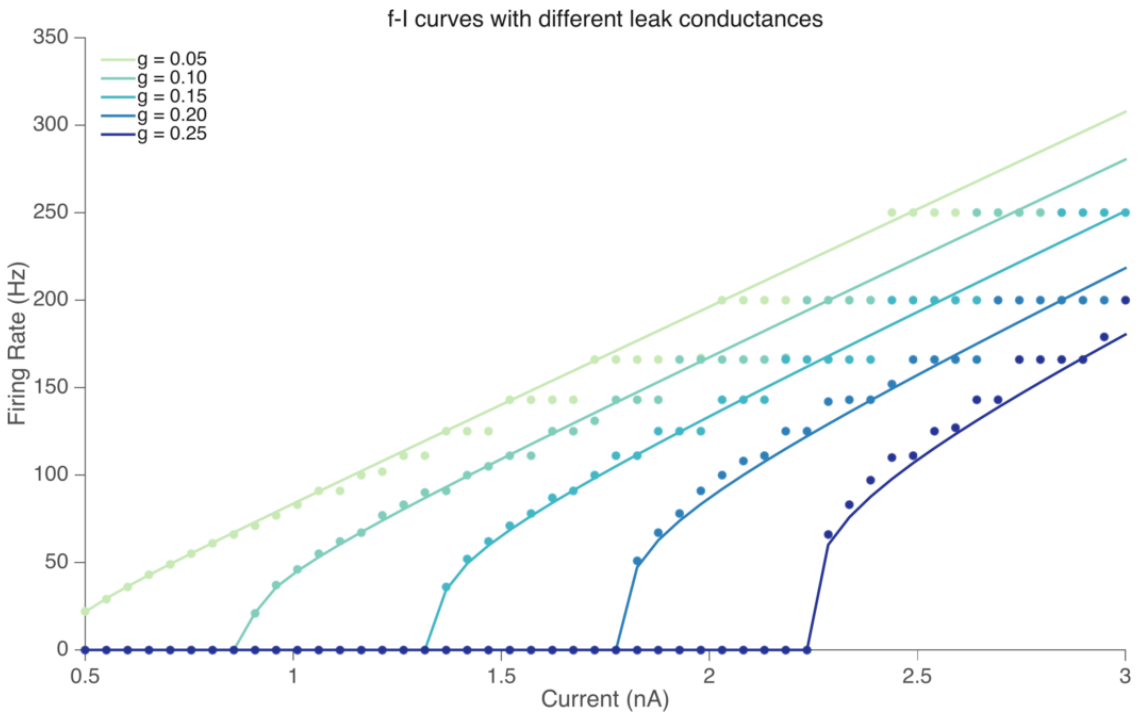

Figure 4: Shunting inhibition shifts (but doesn't scale) the f-I curve of the LIF neuron.

1.4 Background synaptic input

Neurons are constantly receiving background synaptic input from the numerous synapses on the dendritic tree. We will add background synaptic input to our simulated neurons, to see how that might affect the f-I curve. Our model for background input will be extremely simple: we will assume that the net excitatory and inhibitory input each contribute to the soma potential independently at every time step. We will model this input with a gaussian distribution. Furthermore, we will assume that the excitatory and inhibitory inputs are balanced[5]. This means that the mean excitatory input equals the mean inhibitory input, so after we add them, the mean is zero. This lets us model the background input with one parameter: the standard deviation of the background input.

Part 3 We will need to add a noise term to our simulations, and then investigate how this noise affects the f-I curve.

1. Modify your `integrate.m` function to add Gaussian noise to the simulated membrane potential of your neuron, where the noise is drawn independently at every time step of the simulation. The standard deviation of the noise, σ_v, should be a parameter that you can easily manipulate. The `randn` function in Matlab generates a random draw from a zero mean, unit standard deviation Gaussian. You will need to scale this noise to have the appropriate standard deviation given by the value of the parameter σ_v.

[5]For more information on E/I balance, check out `http://www.scholarpedia.org/article/Balance_of_excitation_and_inhibition`

2. Generate an f-I curve with and without added noise, holding all other parameters fixed. What happens to the f-I curve when you add noise? Why might this be the case?

3. The claim of Chance et. al. [2] is that shunting inhibition in addition to background synaptic input is sufficient to cause gain scaling. Can you tune these two parameters (g_{leak} and σ_v) to generate f-I curves with changing slope? Try to see if you can generate a figure showing gain scaling of the f-I curve (Similar to Figure 1A).

4. Discuss the advantages and disadvantages of using the simplified LIF model to explore these ideas, compared to more biophysically detailed simulations.

2 Homeostatic regulaion

2.1 Background

Neural excitability often relies on a carefully tuned balance of membrane and ion channel dynamics that give rise to particular patterns of neural activity [6]. Cells must maintain this balance in the presence of all of the various cellular and molecular processes going on inside the cell, which is a form of homeostasis. For example, the maximum conductance of a particular ion channel depends on the number of ion channels embedded in the membrane. As gene expression fluctuates, this maximum conductance can vary, altering the firing patterns of the neuron.

Here, we will investigate a potential homeostatic mechanism for regulating the conductance of various ion channels using calcium as a feedback signal [4, 5]. To do this, we will simulate a single compartment Hodgkin-Huxley neuron (§2.2). Then, we will explore a modified neuron which incorporates a couple of additional ion currents, including a calcium current (§2.3). Finally, you will build a negative feedback regulator that tries to keep the ion conductances at a desired target level (§2.4), thus making the cell robust with respect to fluctuating ion conductances.

2.2 Hodgkin-Huxley neuron

First, we need to simulate a Hodgkin-Huxley (HH) neuron. Recall that the circuit model for the HH neuron involves a capacitor in parallel with a number of ion channels, each of which is modeled as a battery in series with a resistor. The net current injected into the cell must equal the sum of the current through each of these elements. Mathematically, we can write this as:

$$C_m \frac{dV_m}{dt} = I_{injected} - g_{leak}(V_m - E_{leak}) - g_K(V_m - E_K) - g_{Na}(V_m - E_{Na}) \tag{4}$$

The sodium and potassium conductances in the HH model are voltage and time-dependent, represented as a maximum conductance multiplied by gating variables that control the fraction of open ion channels:

$$g_{Na} = \bar{g}_{Na}^{max} m^3 h \tag{5}$$

$$g_K = \bar{g}_K^{max} n^4, \tag{6}$$

and the gating variables m, n, and h have linear voltage-dependent dynamics. The gating variables evolve according to the following differential equation:

$$\frac{dx}{dt} = \alpha_x(V_m)(1-x) - \beta_x(V_m)x,$$

where x is a variable representing on of the gating variables and α and β are functions of the membrane voltage that define how the dynamics vary with membrane voltage. To simulate the HH equations, we will use the same numerical integration technique that we used to simulate the integrate-and-fire neuron.

The starter code provides a script, `hhsim.m`, which simulates the standard Hodgkin-Huxley neuron. You should familiarize yourself with this script and understand exactly what it is doing–it relies on a couple of helper functions, `make_channel.m` and `gates.m` for generating Matlab structures representing ion channels and for computing the values of the α and β functions for each ion channel species, respectively. Run the `hhsim.m` script and verify that it generates a figure similar to Figure 5.

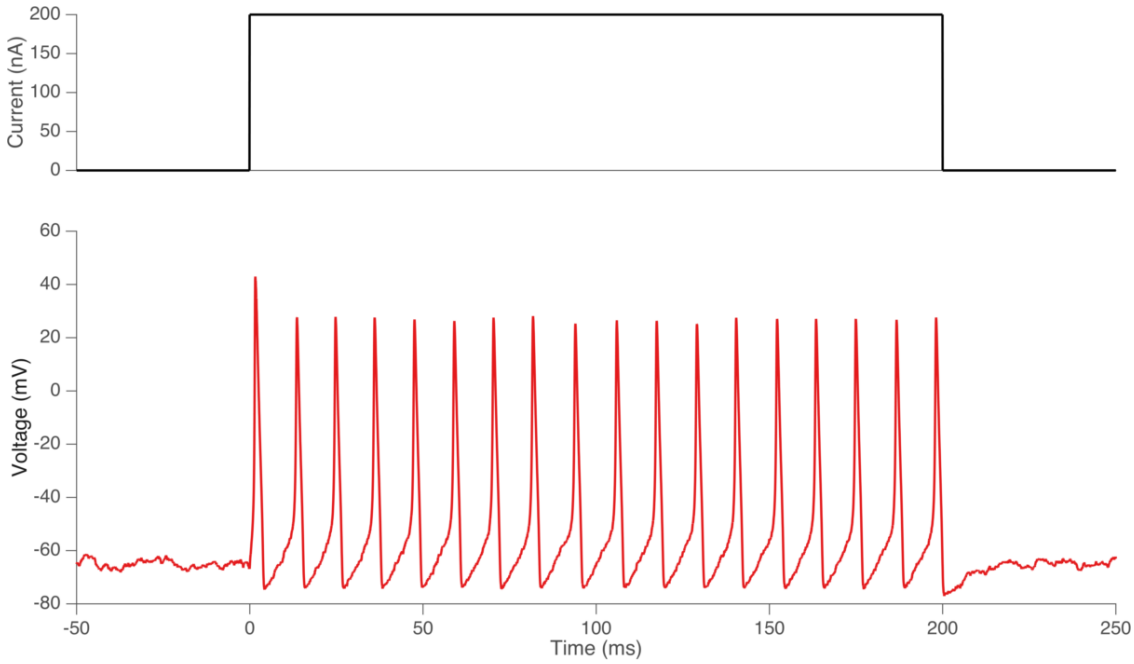

Figure 5: Simulation of a Hodgkin-Huxley neuron in response to a step current pulse.

2.3 Incorporating additional currents

Now we will modify the basic hodgkin-huxley model to explore the role of calcium feedback in homeostasis. We will add two additional currenst to the model: a persistent sodium current and a high-threshold calcium current. In the HH formulation, the maximum conductances g^{max} are fixed and constant. In real neurons, however, these values will fluctuate. We will first explore how changing these conductances can have drastic effects on the output of the neuron.

In the starter code, the script `hhsim_modified.m` adds a persistent sodium current and calcium current

to the simulation. The sodium current is an additional ion channel that has the same reversal potential as the existing sodium current (50 mV) but a much smaller maximum conductance (1.0 $\mu S/mm^2$, you will vary this soon). The addition of this current shouldn't vary the step response of the neuron too much, but if you increase the conductance (to say 10 $\mu S/mm^2$) you should start to see that the neuron fires even when no additional current is injected. The high-threshold calcium current required the addition of two new gating variables, q and r. Note the modified `gates.m` function which computes the α and β functions for two new gating variables as follows:

$$\alpha_q(V_m) = \frac{0.055(V_m+27)}{1-\exp(-0.263(V_m+27))} \quad (7)$$

$$\alpha_r(V_m) = 0.000457\exp(-0.02*(V_m+13)) \quad (8)$$

$$\beta_q(V_m) = 0.94\exp(-0.059(V_m+13)) \quad (9)$$

$$\beta_r(V_m) = \frac{0.0065}{1+\exp(-0.0357(V_m+15))} \quad (10)$$

The calcium current is given by: $I_{Ca} = \bar{g}_{Ca}^{max} q^2 r(V_m - E_{Ca})$. The calcium reversal potential is 134 mV and the max conductance to 0.1 $\mu S/mm^2$.

Part 4 Here you will explore the effect of varying the maximum conductance for different ion species.

1. Explore what happens when you vary the potassium max conductance g_K and the persistent sodium conductance g_{NaP} in the `hhsim_modified.m` script. For inspiration, see some of the behavior obtained by varying these parameters in Figure 6.

2.4 Homeostatic regulation of conductance

A neuron might want to regulate it's firing output in the presence of changing conductances. Here, we will incoporate a regulatory mechanism into the model that achieves this. Here's how it works: the activity of the neuron activates the calcium current, causing calcium to rush into the cell. Thus, calcium is a readout of the firing activity of the cell. We will then explicitly model the calcium concentration using another differential equation. The calcium current directly increases the intracellular calcium concentration, while a decay term models buffering of calcium within the cell. Finally, we will add differential equations that modulate the maximum conductance of a particular ion using the intracellular calcium level as a signal for the cell's activity. The idea is that the presence or lack of intracellular calcium could trigger signaling pathways to drive or inhibit gene expression that would result in an increase or decrease in the number of ion channels embedded in the membrane, modifying the ion's conductance.

Part 5 Here we will augment the model to describe the state of calcium inside the cell.

1. First, add an equation to `hhsim_modified.m` describing the calcium concentration (c.f. equation 2.6 in [1]):

$$\tau_k \frac{d[Ca]}{dt} = -[Ca] - 0.5 I_{Ca},$$

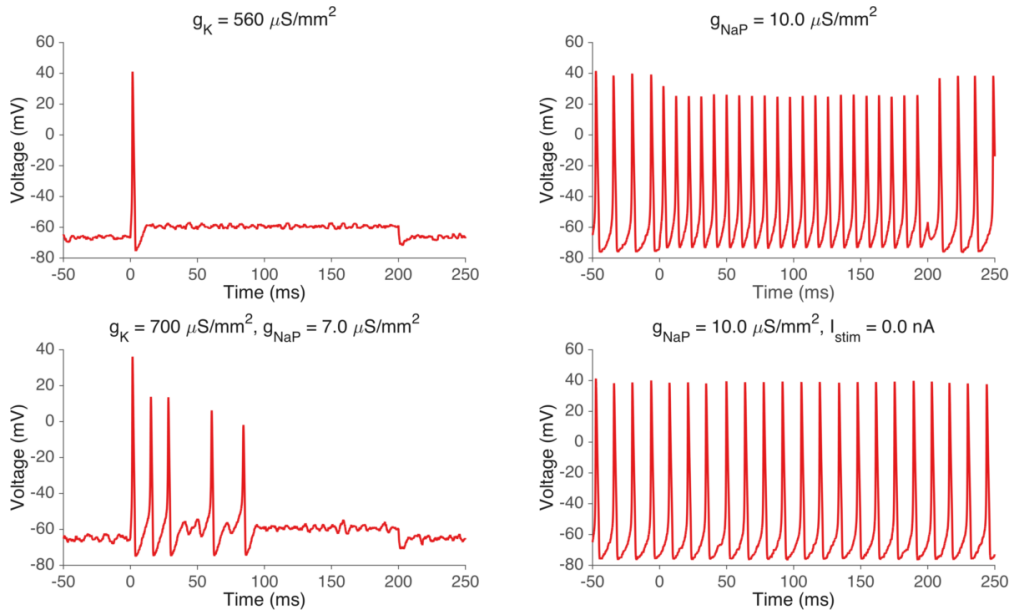

Figure 6: Effect of varying g^{max} for different ion species. Top left: increasing the value of g_K results in a neuron that fires a single action potential in response. Top right: increasing the persistent sodium conductance induces baseline firing in the absence of stimulation, and an increased firing rate during stimulation. Bottom left: modifying both the potassium and persistent sodium conductances yields irregular firing. Bottom right: similar to the top right, but the external current stimulation turned off (to see the baseline firing more clearly).

this now lets you model the internal calcium concentration in the cell. The parameter τ_k defines the kinetics of calcium buffering, and is relative slow (e.g. $\tau_k \approx 100ms$).

2. Plot the calcium concentration above the membrane voltage during a step current stimulation. Describe, in simple terms, what is happening to calcium inside the cell during stimulation.

3. Next, you will add equations that keep the maximum ion conductance at some target value, of the form (c.f. equation 2.10 in [1]):

$$\tau_g \frac{dg_{ion}^{\max}}{dt} = G_{ion}\sigma\left(\pm\frac{C_T - [Ca]}{2\Delta}\right) - g_{ion}^{\max},$$

where you can set G_{ion} to 2x the initial value (G_{ion} determines the target conductance, since when $C_T = [Ca]$ the sigmoid has output $\frac{1}{2}$ and the ion conductance is at steady state when $g^{\max} = G_{ion}/2$). Do this for two channels: the potassium conductance and the persistent sodium conductance. We will assume that the value of $g_{\max}$ for the fast sodium current and for calcium are fixed.

4. The value for τ_g governs the timescale of the feedback mechanism. In reality, this would be quite slow (on the scale of minutes to hours), but rather than wait that long, let's set it to something small (e.g. $\tau_g = 50ms$) to build intuition for what it is doing. The term $\sigma\left(\frac{\pm C_T - [Ca]}{2\Delta}\right)$ is a sigmoid function $\left(\sigma(z) = \frac{1}{1+\exp(-z)}\right)$ that depends on the descrepancy between the actual calcium concentration $[Ca]$

and the desired value C_T (you can use $C_T = 0.05$ for this). Note that you'll have to choose the proper sign in order to provide negative feedback (as opposed to the positive feedback). (Hint: the sign will depend on the direction of that ion's current, either into our out of the cell). Δ is a scaling parameter that controls the sensitivity of the calcium-ion conductance relationship, you can just set it to $\Delta = 0.05$.

5. Plot the current, $g^K_{\max}$ and $g^{NaP}_{\max}$, $[Ca]$, and voltage as a function of time on different subplots. This should give you intuition about how these different variables interact (and their timescales).

6. Create some simulations where you manually perturb the max conductance $g_{\max}$ of one of the channels at some point during the experiment and then observe how the calcium dynamics restore the conductance back to a reasonable level. For example, if you manually crank up the value of $g^K_{\max}$ during current injection, you will inhibit spiking (because you just increased the potassium conductance). However, your feedback mechanism should correct for this, and after a brief pause spiking should resume. (Note: this part is open ended on purpose. Submit plots that have the subplots mentioned in the previous point for whatever simulations you come up with).

7. Discuss advantages and disadvantages of using calcium as a feedback signal for regulating neuronal dynamics.

References

[1] LF Abbott and Gwendal LeMasson. Analysis of neuron models with dynamically regulated conductances. *Neural Computation*, 5(6):823–842, 1993.

[2] Frances S Chance, LF Abbott, and Alex D Reyes. Gain modulation from background synaptic input. *Neuron*, 35(4):773–782, 2002.

[3] Gary R Holt and Christof Koch. Shunting inhibition does not have a divisive effect on firing rates*. *Neural computation*, 9(5):1001–1013, 1997.

[4] Gwendal LeMasson, Eve Marder, and LF Abbott. Activity-dependent regulation of conductances in model neurons. *Science*, 259:1915–1915, 1993.

[5] Gina Turrigiano, LF Abbott, Eve Marder, et al. Activity-dependent changes in the intrinsic properties of cultured neurons. *Science*, 264(5161):974–976, 1994.

[6] A. H Williams, T. O'Leary, and E. Marder. Homeostatic Regulation of Neuronal Excitability. *Scholarpedia*, 8(1):1656, 2013. revision #140978.