Chem 163 2022
HW 6, Due Thurs Nov. 3, 2022
Modeling dynamical systems

1) Integrate-and-fire neuron
In this problem we'll make a simple Matlab simulation of an integrate-and-fire neuron. The membrane voltage evolves following:

$$C\frac{dV}{dt} + g_{Na}m_{\infty}(V)(V - E_{Na}) + g_K(V - E_K) = I, \qquad [1]$$

where C is the membrane capacitance, $m_{\infty}(V)$ is the sodium activation gate, V is the membrane voltage, g_{Na} and g_K are the sodium and potassium conductances respectively, E_{Na} and E_K are the corresponding reversal potentials, and I is the injected current.

A) Suppose the sodium activation gate, $m_{\infty}(V)$ is modeled as an activating conductance (open at more positive voltage) with a $V_{1/2}$ = -40 mV and a gating charge of q_{Na} = +6. Write an expression for $m_{\infty}(V)$ assuming that this conductance responds fast enough to always be at equilibrium.

B) Let's assume that the potassium conductance is a simple Ohmic leak with reversal potential E_K.

Let C = 100 pF
$g_{Na}^{max} = 5\ \mathrm{nS}$ (nanosiemens)
E_{Na} = 40 mV
$V_{1/2}$ = -40 mV

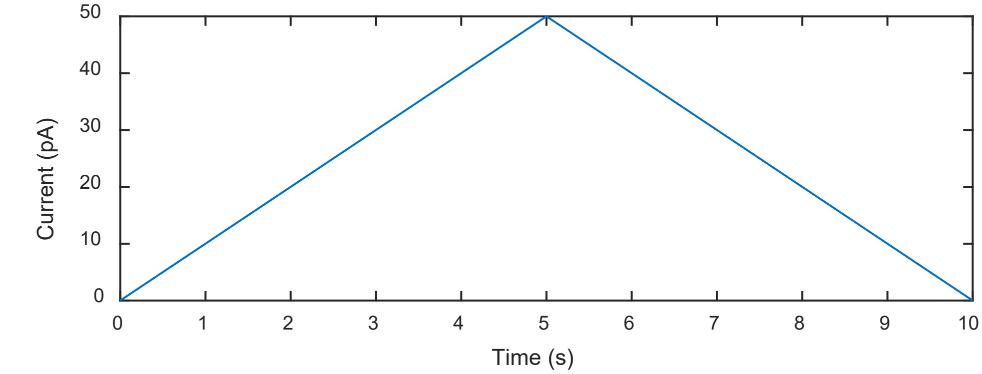
Make plots of:
m_{∞} vs. V
$g_{Na}m_{\infty}(V)(V - E_{Na})$ vs. V

C) Assume that E_K = -60 mV. Make a plot which overlays graphs of $g_{Na}m_{\infty}(V)(V - E_{Na}) + g_K(V - E_K)$ for g_K ranging from 0 to 10 nS, in steps of 1 nS.

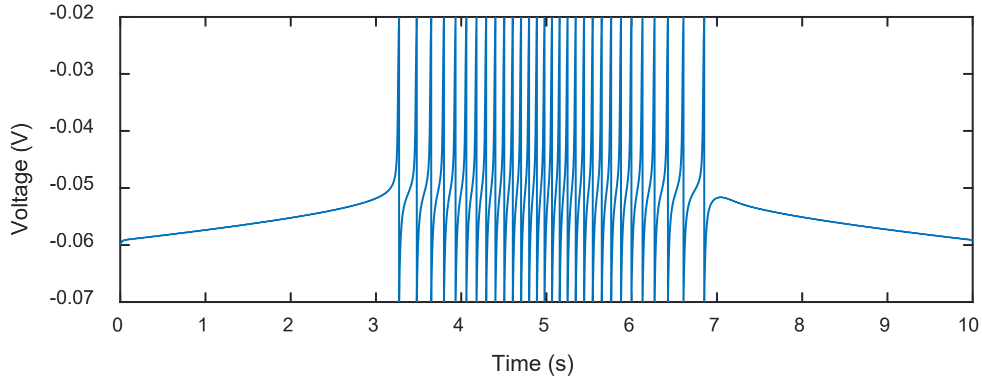
D) Assume g_K = 5 nS. For injected currents, I, ranging from -200 pA to 200 pA in steps of 1 pA, calculate the equilibrium voltage(s). Make a plot showing the voltage(s) as a function of I. Indicate stable equilibria with points (Matlab '.' linespec) and unstable equilibria with open circles (Matlab 'o' linespec).

E) Now let's look at the dynamics. Set up an integrate-and-fire neuron where the maximum voltage (the 'threshold voltage') is V_{th} = -20 mV, the 'reset voltage' is V_r = -70 mV, and the dynamics between V_r and V_{th} are governed by Eq. 1 above. If the voltage ever crosses V_{th}, it is immediately reset to V_r.

Set up a time axis of duration 10 s in steps of 0.1 ms. Set up a current ramp I that goes from 0 to 50 pA during the first 5 s, and then goes from 50 to 0 pA during the second 5 s, i.e. the current vs. time should look like this:



Simulate the voltage vs. time. Your result should look something like this:



Now repeat this calculation with V_r = -50 mV and make a plot of V vs. t. What is different about the plot of V vs. t?

Now repeat this calculation with V_r = -45 mV and make a plot of V vs. t. Can you explain what is going on?

2) Izhikevich model neuron

This problem is based on a famous article by the author of our textbook on Dynamical Systems in Neuroscience:

Izhikevich, Eugene M. "Simple model of spiking neurons." *IEEE Transactions on neural networks* 14.6 (2003): 1569-1572.

Read this article (it is short!).

A) The Izhikevich model is a generalization of the simple quadratic integrate and fire model. The Izhikevich model includes the fact that the parameters of an integrate-and-fire neuron can change depending upon the past history of spiking. Many neurons have additional feedback channels, such as Ca^{2+}-gated K^+ channels, which suppress excitability if a neuron is spiking at a high rate, i.e. the spiking threshold depends on the history. These feedbacks can lead to rich dynamics. A striking feature of the Izhikevich model is that it can capture many of the firing patterns observed in the brain with just a few simple parameters.

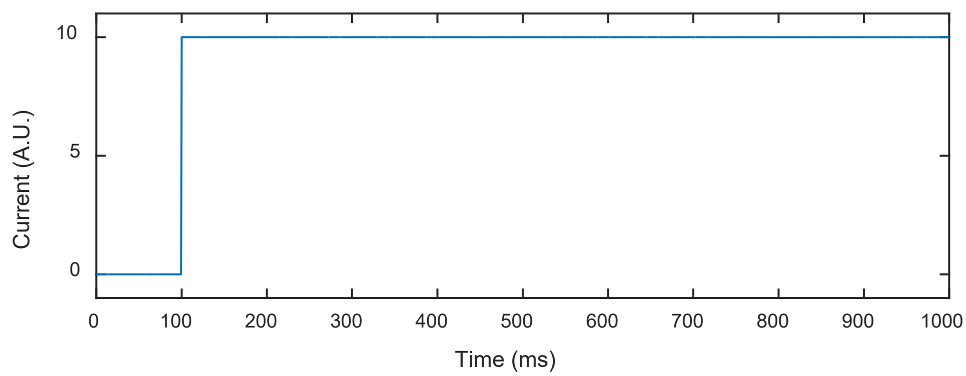
The parameter *u* in the model captures the slow feedbacks: it acts as another current which is added to the injected current.

Write a Matlab program to simulate the Izhikevich model (Eqs. 1 – 3 from the Izhikevich paper). Note that in the Izhikevich formulas, V is measured in mV and t in ms.

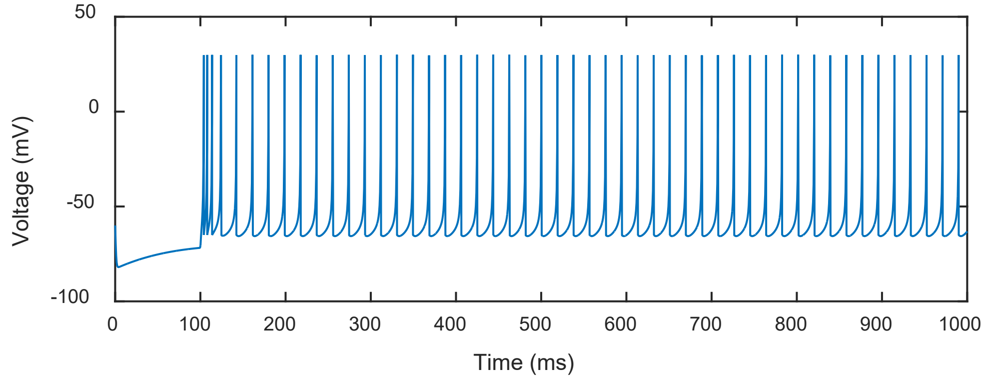
Simulate a 1 s voltage waveform using the Izhikevich model with the following parameters:

```
a=0.02;
b=0.2;
c=-65;
d=2;
Vth = 30; % mV
dt = 0.001; % ms
t = (0:dt:1e3); % in ms
```

Have the voltage start at -60 mV and the parameter *u* start at 0. Set the current to jump from *I* = 0 to *I* = 10 at 0.1 s. The input current should look like this:



Your result should look like this:



Now repeat this exercise with the following sets of parameters:

```
a=0.02;
b=0.2;
```

```
c=-55;
d=4;
```

and:

```
a=0.02;
b=0.2;
c=-50;
d=2;
```

These parameters mimic the behavior of different types of cortical neurons.

3) Simple conductance-based neuron model

In the prior examples we had to reset the voltage 'by hand' when it crossed a threshold. In this problem we will implement a simple model of a neuron where a voltage-gated K^+ channel drives the downstroke of the action potential.

The model is:

$$C\frac{dV}{dt} + g_{Na}m_{\infty}(V)(V - E_{Na}) + g_K n(V,t)(V - E_K) = I$$
$$\frac{dn}{dt} = \frac{n_{\infty}(V) - n}{\tau(V)}$$

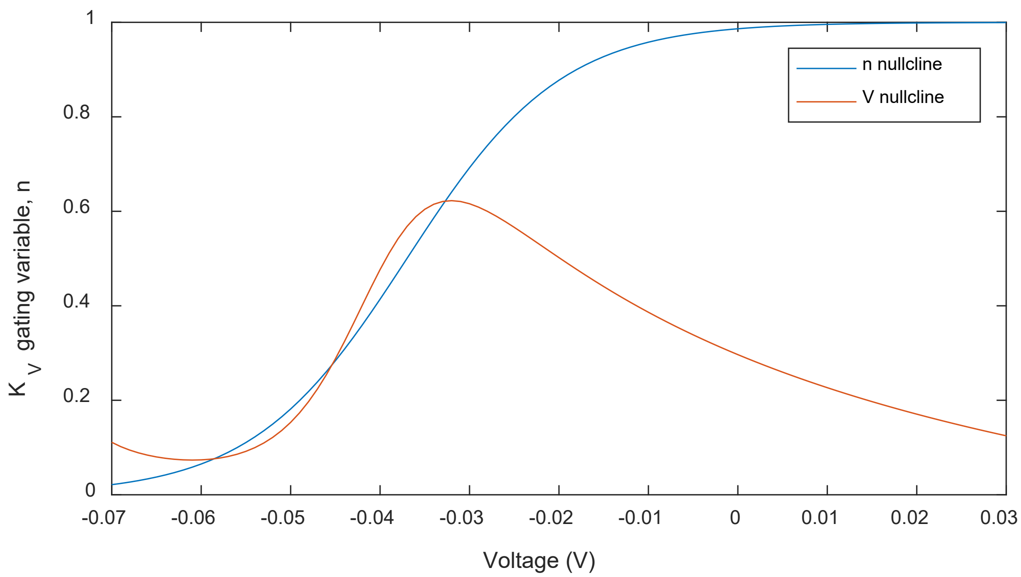
The new effect is that the potassium channel (K_V) now has an activation gate, *n*, which is not instantaneous. As before, the Na_V channel drives the upstroke of the action potential. But then the gradual activation of the K_V channel causes the upper fixed point to become unstable, so the voltage then returns to baseline.

A) Suppose the K_V channel activates at positive voltages, with a half-activation voltage $V_{1/2}^K$ and a gating charge q_K. Write an expression for $n_{\infty}(V)$ (this is similar to problem 1A).

B) Calculate the nullclines of this system, i.e. $n_V(V)$ for which dV/dt = 0, and $n_n(V)$ for which dn/dt = 0. Make a plot of these curves for V between -70 and +30 mV with the parameters:

```
gNa = 5e-9; % Siemens
ENa = 0.068;  % Sodium reversal potential, Volts
VNaHalf = -0.04;  % NaV channel half-activation voltage, Volts
qNa = 6;  % Dimensionless NaV gating charge
gK = 15e-9; % Siemens
EK = -0.08; % Potassium reversal potential, Volts
VKHalf = -0.037; % KV channel half-activation voltage, Volts
qK = 3; % Dimensionless KV gating charge
Iinj = 16e-12;  % Amps
```

Your result should look something like this:



Identify on your graph the fixed points of the system.

Find a set of parameters for which there is only one fixed point and plot the corresponding nullclines.

C) Suppose that the transition state for opening the K_V channel lies half way between the open and closed states, i.e. the voltage-induced shift in transition state energy is half of the difference between open and closed state energies. Write an expression for $\tau(V)$. You can use a parameter k_0 for the \`attempt frequency' (assumed to be the same in both the open and closed state) and U_0 for the barrier height when $V = V_{1/2}^K$.

Write an expression for $\frac{\tau(V)}{\tau\left(V=V_{1/2}^K\right)}$ and make a plot of this quantity as a function of *V*, assuming $V_{1/2}^K = -30\ \mathrm{mV}$.

D) Write a function with the following header:

```
Function [dvdt, dndt] = HHmodel(V, n, params);
% V is the membrane voltage
% n is the gating variable on a Kv channel
% params is a vector containing:
% [membrane capacitance (F), gNa (S), ENa (V), VNaHalf (V),
% qNa (dimensionless), gK (S), EK (V), VKHalf (V),
% qK (dimensionless), tau(V = VKHalf), I (Amps)]
```

Use the function ode45 to solve for V(t) and n(t) on t in [0, 3 seconds] using the following parameters:

```
C = 1e-10;  % Farads
gNa = 5e-9; % Siemens
ENa = 0.068;  % Volts
VNaHalf = -0.04;  % Volts
qNa  = 6;  % Dimensionless
gK = 15e-9; % Siemens
EK = -0.08; % Volts
```

```
VKHalf = -0.04; % Volts
qK = 4; % Dimensionless
tau0 = 50e-3; % seconds
Iinj = 16e-12;  % Amps
```

To call the HHmodel function using ode45, it is helpful to have the wrapper function:

```
function out = HHwrapper(t, y, params);
% wrapper function for HHmodel that concatenates the outputs into a single
% vector.  Useful for evaluating with ode solvers.

V = y(1);
n = y(2);
[dvdt, dndt] = HHmodel(V, n, params);
out = [dvdt; dndt];
```

To solve the ODEs on the time interval 0 – 3 s with the initial conditions V = -0.02, n = 0.5, you can run the code:

```
tSpan = [0, 3];
y0 = [-0.02; 0.5];
[t, y] = ode45(@(t,y) HHwrapper(t,y, params), tSpan, y0);
```

Then y(:,1) is the voltage waveform and y(:,2) is the n waveform.

Make some plots showing the dynamics on the phase plots (i.e. *n* vs. *V*) and vs. time. Explore the roles of the different parameters in the 'params' vector, and the initial conditions. See if you can make a system that oscillates stably. Here are two examples (you'll have to figure out the parameters!):