An Introductory Course in Computational Neuroscience—Paul Miller (Notes)

Malcolm

Started 14 Dec 2024

Contents

0.1 xLIF

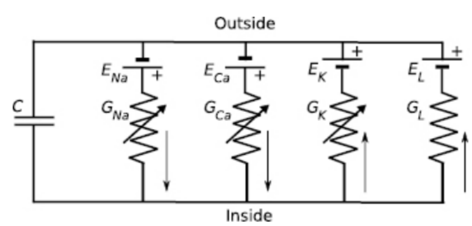
0.1.1 Modelling the Leaky membrane potential

Nernst Potential
The *Nernst potential* E_A of an ion A of charge z_A with intracellular concentration $[A_{\text{in}}]$ and extracellular concentration $[A_{\text{out}}]$ is given by

$$E_A = \frac{k_B T}{z_A q_e} \ln\left(\frac{[A_{\text{out}}]}{[A_{\text{in}}]}\right)$$

where T is the temperature in Kelvin, k_B the Boltzmann constant ($1.39 \times 10^{-23} JK^{-1}$) (which converts units of temperature to units of thermal energy). q_e is the fundamental electronic charge ($1.60 \times 10^{-19} C$).

Model
Considering this representation of a neuron's membrane:



If all channels with variable conductance are closed, then the current will only flow through the leak channels (subscript L) until the cell membrane is at the leak potential E_L. The current through a channel is given by

$$I_t = G_t(V_m - E_t)$$

Where G_t represents conductance and E_t the nernst potential; t represents the type of channel.

(next page)

Equilibrium

When the cell is at equilibrium the different currents balance each other out and sum to zero:

$$I_m = \sum_t I_t = \sum_t G_t(V_m - E_t) = 0$$

In the context of this current model this can be rewritten as

$$G_{Na}(V_m - E_{Na}) + G_{Ca}(V_m - E_{Ca}) + G_K(V_m - E_K) + G_L(V_m - E_L)$$

Solving for V_m we can see that the *resting membrane potential*—where no net current flows, is the weighted average of the individual Nernst potentials:

$$V_m = \frac{G_{Na}E_{Na} + G_{Ca}E_{Ca} + G_K E_K + G_L E_L}{G_{Na} + G_{Ca} + G_K + G_L}$$

The derivation of the resting membrane potential is typically more complicated.

Leaky membrane potential

Here we consider the passive properties of the cell, where the variable conductance of all channels are fixed. With this the we treat the circuit as having a single 'leak' conductance and potential.

The membrane potential is generated by the charge stored on the membrane; it depends on both the stored charge and the membrane's capacitance C_m via the equation

$$Q = C_m V_m$$

The current is defined as positive when it flows *out* of the cell; with that we have

$$\frac{dQ}{dt} = -I_m = -G_L(V_m - E_L)$$

Fixing the capacitance we obtain the dynamics of the resting membrane potential as

$$C_m \frac{dV_m}{dt} = G_L(E_L - V_m)$$

This is a linear first order ODE.

0.1.2 Solution for Leaky ODE

We had the dynamics of the resting membrane potential as

$$C_m \frac{dV_m}{dt} = G_L(E_L - V_m)$$

Expressing in standard form and solving by integrating factor:

$$\begin{aligned}
\frac{dV_m}{dt} + \frac{G_L}{C_m} V_m = \frac{G_L}{C_m} E_L \\
V_m = \frac{1}{\exp\left(\frac{G_L}{C_m} t\right)} \left(\int \exp\left(\frac{G_L}{C_m} t\right) \cdot \frac{G_L}{C_m} E_L \, dt + c \right)
\end{aligned}$$

To simplify we define the *time constant* $\tau_m = C_m / G_L$:

$$\begin{aligned}
V_m &= \frac{1}{\exp\left(\frac{t}{\tau_m}\right)} \left(\int \exp\left(\frac{1}{\tau_m} t\right) \cdot \frac{1}{\tau_m} E_L \, dt + c \right) \\
&= \exp\left(-\frac{t}{\tau_m}\right) \left(\exp\left(\frac{t}{\tau_m}\right) E_L + A \right) \\
&= E_L + \exp\left(-\frac{t}{\tau_m}\right) \cdot A
\end{aligned}$$

At initial condition $V_m(0)$:

$$V_m(0) = E_L + A \implies A = V_m(0) - E_L$$

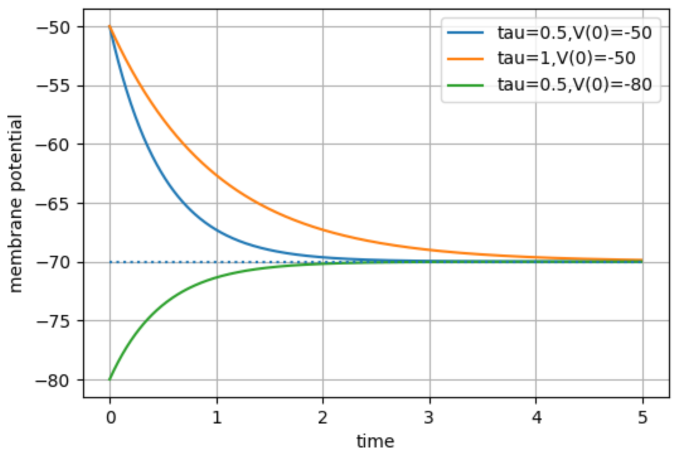
With that we have the solution

$$V_m = E_L + (V_m(0) - E_L) \exp\left(-\frac{t}{\tau_m}\right)$$

(next page)

Illustrated
See that the equation tends to E_L, and that τ_m dictates how fast this decay occurs (thus the name). Illustrated here using code from `leakymembrane.py`:

$$V_m = E_L + (V_m(0) - E_L)\exp\left(-\frac{t}{\tau_m}\right)$$



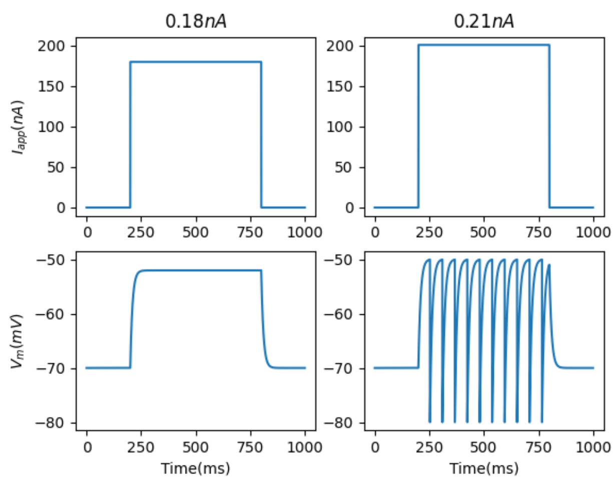
0.1.3 Leaky-Integrate-and-Fire

The LIF introduces a framework on which we can build more realistic models of neurons. It is essentially the intial model for the leaky membrane with an additional term I_{app} modelling an applied current. The spike is modelled by the membrane potential being reset to a low (hyperpolarised) value V_{reset} after the potential reaches some threshold V_{th} (threshold potential):

$$C_m \frac{dV_m}{dt} = G_L(E_L - V_m) + I_{\text{app}}; \text{if } V_m > V_{th} \text{ then } V_m \mapsto V_{\text{reset}}$$

Notice that an actual 'spike' shape hasn't been modelled, and would have to be put in by hand before V_m is reset; spike times are recorded at the time when the membrane potential crosses the threshold.

The following was simulated using code from `LIFfunc.py`:



Threshold current
See that there is an insufficient level for I_{app} where no 'firing' occurs (the membrane potential does not reach threshold and the model does not spike). Setting $dV_m/dt = 0$ allows us to obtain the *steady state* membrane potential as

$$V_m^{ss} = E_L + I_{\text{app}}/G_L$$

If the steady state is below threshold then the model does not fire. By setting $V_m^{ss} = V_{th}$ we can obtain the *threshold current* I_{th}—the minimum applied current required to elicit firing:

$$I_{th} = G_L(V_{th} - E_L)$$

0.1.4