FYS388

Relevant formulas and exercises.

Equations

Nernst-Planck

The Nernst-Planck equation

$$J_X = J_{X,\text{diffusion}} + J_{X,\text{drift}}$$

$$= -D_X \left(\frac{d[X]}{dx} + \frac{z_X F[X]}{RT} \frac{dV}{dx} \right)$$

Where D_X is the diffusion coefficient for ion X, $[X]$ the concentration of ion X ($d[X]/dx$ the concentration gradient), z_X the valence of the ion, F Faraday's constant, R the gas constant, T the temperature and V the potential (and therefore dV/dx the electric field).

Derivation of Nernst potential from the Nernst-Planck equation

Because we want net zero movement we set $J_X = 0$, and solve for the potential.

$$J_X = -D_X \left(\frac{d[X]}{dx} + \frac{z_X F[X]}{RT} \frac{dV}{dx} \right)$$

$$0 = -D_X \left(\frac{d[X]}{dx} + \frac{z_X F[X]}{RT} \frac{dV}{dx} \right)$$

$$= \frac{d[X]}{dx} + \frac{z_X F[X]}{RT} \frac{dV}{dx}$$

$$-\frac{z_X F[X]}{RT} \frac{dV}{dx} = \frac{d[X]}{dx}$$

$$-\frac{z_X F}{RT} \frac{dV}{dx} = \frac{1}{[X]} \frac{d[X]}{dx}$$

$$-\frac{z_X F}{RT} dV = \frac{1}{[X]} d[X]$$

$$-\frac{z_X F}{RT} \int_0^{E_m} dV = \int_{[X]_\text{in}}^{[X]_\text{out}} \frac{1}{[X]} d[X]$$

$$-\frac{z_X F}{RT} E_m = \ln \frac{[X]\text{out}}{[X]\text{in}}$$

$$E_m = -\frac{RT}{z_X F} \ln \frac{[X]\text{out}}{[X]\text{in}}$$

□

The Goldman-Hodgkin-Katz (GHK) model

Current equation

$$I_X = P_X z_X^2 \frac{F^2 V}{RT} \left(\frac{[X]_{\text{in}} - [X]_{\text{out}} e^{-z_X FV/RT}}{1 - e^{-z_X FV/RT}} \right)$$

Voltage equation

Given the GHK equation for current, and the equilibrium criterion ($\sum I = 0$) we get

$$E_m = \frac{RT}{F} \ln \left(\frac{\sum_X^{\text{ions}} P_X [X]_{\text{out (if + else in)}}}{\sum_X^{\text{ions}} P_X [X]_{\text{in (if + else out)}}} \right)$$

where P_X is the permeability (of ion X), z_X the valence, F Faraday's constant and V the membrane potential.

Capacitive current

The capacitive current is a current (charge flow) across the membrane that does not move through ion channels. This current charges/discharges the membranes capacitance – making the membrane have the same properties as a capacitor. The membrane can in other terms hold and separate charge on either side of the lipid bi-layer.

$$I_c = C_m \frac{dV}{dt}$$

where I_c is the capacitive current, C_m the capacitance and $\frac{dV}{dt}$ the rate of voltage change across the membrane.

Derivation of the reversal potential (Em) for a neuron with several quasi-ohmic ion channels

$$\begin{aligned} I_m &= I_x + I_y + I_z \\ &= g_x(V - E_x) + g_y(V - E_y) g_z(V - E_z) \\ &= (g_x + g_y + g_z) \left(V - \frac{g_x E_x + g_y E_y + g_z E_z}{g_x + g_y + g_z} \right) \\ &= g_m(V - E_m) \end{aligned}$$

where

$$g_m = \frac{1}{R_m} = g_x + g_y + g_z \qquad \text{and} \qquad E_m = \frac{g_x E_x + g_y E_y + g_z E_z}{g_x + g_y + g_z}$$

□

RC-circuit neuron

Derivation of a differential equation for a simple "RC-circuit" neuron

For a RC-neuron we have the capacitive and ionic currents

$$I_{\text{capacitive}} = C_m \frac{dV}{dt}$$

$$I_{\text{ionic}} = \frac{E_m - V}{R_m}$$

When looking at the whole cell, with a surface area a, through an electrode (using Kirchhoff's current law) we get

$$I_{\text{electrode}} = a \times I_{\text{capacitive}} + a \times I_{\text{ionic}}$$

$$a \times I_{\text{capacitive}} = -a \times I_{\text{ionic}} + I_{\text{electrode}}$$

$$C_m \frac{dV}{dt} = \frac{E_m - V}{R_m} + \frac{I_{\text{electrode}}}{a}$$

□

Solving for the membrane potential receiving constant current starting at an initial time and determining the constants A, B, and C

$$\frac{dV}{dt} = \frac{E_m - V}{C_m R_m} + \frac{I_{\text{electrode}}}{C_m a}$$

$$V(t) = A + B(1 - \exp\left(-\frac{t}{C_m R_m}\right))$$

When solving the differential equation we can see that $C = \tau = C_m R_m$, which is the time constant. A and B can further be found by solving the equation at $t = 0$ and $t = \infty$.

$$\begin{aligned} V(t=0) &= A + B(1 - \exp\left(-\frac{0}{C_m R_m}\right)) \\ &= A + B(1-1) \\ &= A \end{aligned}$$

□

$$\begin{aligned} V(t=\infty) &= A + B(1 - \exp\left(-\frac{\infty}{C_m R_m}\right)) \\ &= A + B(1-0) \\ &= V(0) + B \end{aligned}$$

Which means that $V(0) = A = E_m$ is the resting potential of the cell. □

What is the limiting value of V when t goes to infinity?

During current injection

When the cell is receiving constant current, the voltage goes towards a new "resting" potential.

$$\begin{aligned} V(0) &= E_m \\ V(t=\infty) &= V(0) + B \\ V(\infty) &= E_m + B \\ V(\infty) &= E_m + I_{\text{electrode}} R_m / a \end{aligned}$$

This new potential is the previous resting potential plus an additional potential difference.

After current injection

After the current has been injected, the voltage goes back towards the resting potential.

$$V(\infty) = E_m$$

The membrane time constant τ_m

The time constant is

$$\tau_m = C_m R_m$$

Hodgkin-Huxley model

Action potential generation

$$I_{\text{capacitive}} = C_m \frac{dV}{dt}$$

$$I_L = \bar{g}_L (V - E_L)$$

$$I_{\text{Na}} = g_{\text{Na}}(V, t)(V - E_{\text{Na}})$$

$$I_K = g_K(V, t)(V - E_K)$$

We can rewrite $g_x = \bar{g}_x p_x(V, t)$, where $p_x(V, t)$ is the (probability of the channel being open for single cells and) fraction of open channels (for the cell).

Potassium K+

Hodgkin and Huxley based the following equations on experimental data. They found that four gates for the K+ channel was a good fit (n^4).

$$I_K = \bar{g}_K n^4 (V - E_K)$$
$$\frac{dn}{dt} = \alpha_n (1 - n) - \beta_n n$$

$$\alpha_n(V) = 0.01 \frac{V + 55}{1 - e^{-(V+55)/10}}$$
$$\beta_n(V) = 0.125 e^{-(V+65)/80}$$

where V is in mV.

Sodium Na+

The sodium channels (unlike potassium ones) peaks and then decays to zero over time. The model therefore both has an activation variable m (as n for K+), and an inactivation variable h.

$$\begin{aligned} I_{\mathrm{Na}} &= \bar{g}_{\mathrm{Na}} m^3 h (V - E_{\mathrm{Na}}) \\ \frac{dm}{dt} &= \alpha_m (1 - m) - \beta_m m \\ \frac{dh}{dt} &= \alpha_h (1 - h) - \beta_h h \end{aligned}$$

$$\begin{aligned} \alpha_m(V) &= 0.1 \frac{V + 40}{1 - e^{-(V+40)/10}} \\ \beta_m(V) &= 4e^{-(V+65)/18} \\ \alpha_h(V) &= 0.07e^{-(V+65)/20} \\ \beta_h(V) &= \frac{1}{1 + e^{-(V+35}/10} \end{aligned}$$

Full Hodgkin-Huxley model

$$C_m \frac{dV}{dt} = -\bar{g}_L (V - E_L) - \bar{g}_K n^4 (V - E_K) - \bar{g}_{\mathrm{Na}} m^3 h (V - E_{\mathrm{Na}}) + I_{\mathrm{stim}}$$

Where

$$\begin{aligned} C_m &= 1.0 \frac{\mu F}{\mathrm{cm}^2} \\ E_{\mathrm{Na}} &= 50\,\mathrm{mV} & E_K &= -77\,\mathrm{mV} & E_L &= -54.4\,\mathrm{mV} \\ g_{\mathrm{Na}} &= 120 \frac{\mathrm{mS}}{\mathrm{cm}^2} & g_K &= 36 \frac{\mathrm{mS}}{\mathrm{cm}^2} & g_L &= 0.3 \frac{\mathrm{mS}}{\mathrm{cm}^2} \end{aligned}$$

Cable equation

Derivation of the cable equation from the fundamental equation for multi-compartmental modeling

$$I_j^M = I_j^{\mathrm{cap}} + I_j^{\mathrm{ion}} + I_j^{\mathrm{stim}} = \pi \times d \times l \times C_m \frac{dV_j}{dt} + \pi dl \frac{V_j - E_m}{R_m} + I_j^{\mathrm{stim}}$$

$$\pi \times d \times l \times C_m \frac{dV_j}{dt} = \pi dl \frac{E_m - V_j}{R_m} + \pi d^2 \left(\frac{V_{j+1} - V_j}{4R_a l} - \frac{V_j - V_{j-1}}{4R_a l} \right) - I_j^{\mathrm{stim}}$$

$$C_m \frac{dV_j}{dt} = \frac{E_m - V_j}{R_m} + \frac{d}{4R_a} \left(\frac{V_{j+1} - V_j}{l^2} - \frac{V_j - V_{j-1}}{l^2} \right) - \frac{I_j^{\mathrm{stim}}}{\pi dl}$$

□

Derivation of the steady-state solution for a semi-infinite cable with a constant current injected in one end

$$C_m \frac{\partial V(x,t)}{\partial t} = \frac{E_m - V(x,t)}{R_m} + \frac{d}{4R_a}\left[\frac{1}{\partial x}\left(\frac{V(x+\partial x,t)-V(x,t)}{\partial x} - \frac{V(x,t)-V(x-\partial x,t)}{\partial x}\right)\right] + \frac{I_e(x,t)}{\pi d}$$

Where the term inside the square brackets goes towards $\partial^2 V/\partial x^2$ when $\partial x \to 0$:

$$\frac{\partial^2 V(x,t)}{\partial x^2} = \lim_{\partial x \to 0} \frac{1}{\partial x}\left(\frac{V(x+\partial x,t)-V(x,t)}{\partial x} - \frac{V(x,t)-V(x-\partial x,t)}{\partial x}\right)$$

With this inserted into the original equation we get:

$$C_m \frac{\partial V(x,t)}{\partial t} = \frac{E_m - V(x,t)}{R_m} + \frac{d}{4R_a}\frac{\partial^2 V(x,t)}{\partial x^2} + \frac{I_e(x,t)}{\pi d}$$

Which can be rewritten to:

$$\tau_m \frac{\partial V(x,t)}{\partial t} = E_m - V(x,t) + \frac{dR_m}{4R_a}\frac{\partial^2 V(x,t)}{\partial x^2} + \frac{I_e(x,t)R_m}{\pi d}$$

$$= E_m - V(x,t) + \lambda^2 \frac{\partial^2 V(x,t)}{\partial x^2} + \frac{I_e(x,t)R_m}{\pi d}$$

□

The length constant lambda λ

In the equation above, τ_m is the membrane time constant ($R_m C_m$), and lambda

$$\lambda = \sqrt{\frac{dR_m}{4R_a}} = \sqrt{\frac{r_m}{r_i}}$$

the electronic length constant (for a cylindrical cable), which is a typical length scale. In the equation, $r_m = dR_m$ is the membrane resistance per unit length, and $r_i = 4R_a$ the internal (axial) resistance per unit length.

Derivation of the input resistance for a semi-infinite cable with a constant current injected in one end from the steady-state solution

In the steady-state condition, the temporal derivative of membrane potential ($\partial V/\partial t$) is zero, which simplifies the equation significantly.

When looking at a semi-infinite cable, we can make some simplifications. Assuming we have a current I_e injected at one end of a semi-infinite cable, we get

$$\frac{\partial V}{\partial t} = E_m - V + \lambda^2 \frac{\partial^2 V}{\partial x^2} \quad (+ \text{ current at } x = 0)$$

$$0 = \frac{d^2 \hat{V}}{dx^2} - \frac{1}{\lambda^2} \qquad (\hat{V} = V - E_M \quad \rightarrow \quad \frac{\partial \hat{V}}{\partial x} = \frac{\partial V}{\partial x})$$

$$0 = \frac{\partial^2 \hat{V}}{\partial x^2} - \frac{1}{\lambda^2}\hat{V}$$

Solving this differential equation yields

$$\hat{V}(x) = \hat{V}(0)e^{-x/\lambda} \qquad (e^{-x/\lambda} \text{ diverges as } x \to \infty)$$

$$V(x) = E_m + (V(0) - E_m)e^{-x/\lambda}$$

$$R_\infty = \frac{\Delta V}{I_e} = \frac{V(0) - E_m}{I_e} \qquad \text{INPUT RESISTANCE}$$

□

Synapses

Functions for the postsynaptic conductance following the presynaptic arrival of an action potential

i. exponential decay	$g_{\text{syn}}(t) = \bar{g}_{\text{syn}} e^{-(t-t_s)/\tau} \Theta(t - t_s)$
ii. α-function	$g_{\text{syn}}(t) = \bar{g}_{\text{syn}} \frac{t - t_s}{\tau} e^{-(t-t_s)/\tau} \Theta(t - t_s)$
iii. β-function	$g_{\text{syn}}(t) = \bar{g}_{\text{syn}} \frac{\tau_1 \tau_2}{\tau_1 - \tau_2} \left(e^{-(t-t_s)/\tau_1} - e^{-(t-t_s)/\tau_2} \right) \Theta(t - t_s)$

Modelling spike-timing dependent plasticity

$$\Delta w_{ij} = \begin{cases} A_{\text{LTP}} e^{-\Delta t/\tau_{\text{LTP}}} & \text{if } \Delta \geq 0 \\ -A_{\text{LTD}} e^{-\Delta t/\tau_{\text{LTD}}} & \text{if } \Delta t < 0 \end{cases}$$

Where *LTP* is long-term potentiation and *LTD* long-term depression, and

$$\begin{aligned} t_i &= t_{\text{pre}} && \text{time of presynaptic spike} \\ t_j &= t_{\text{post}} && \text{time of postsynaptic spike} \end{aligned}$$

$$\begin{aligned} \Delta t &= t_i - t_j \\ &= t_{\text{pre}} - t_{\text{post}} \end{aligned}$$

Modelling electrical synapses

Two axons connected by a gap junction can be modelled by ohmic connections,

$$\begin{aligned} I_1 &= g_c(V_2 - V_1) \\ I_2 &= -I_1 = g_c(V_1 - V_2) \end{aligned}$$

Active ion channels

Structure of gating-particle models for ion channel currents

Gating-particle models assume that the opening and closing (i.e. gating) of ion channels are controlled by the movement of charged particles within the channel structure.

Ion currents modelled in the Hodgkin-Huxley way by means of *activating* (a) and *inactivating* particles (b). Their dynamics is described by:

$$I(t) = \bar{g}a^x b^y (V - E)$$

For potassium current, this is:

$$I_A = \bar{g}_A a^3 b (V - E_A)$$

In *thermodynamic models* the form of the rate coefficients are derived from "transition state theory".

$$x_\infty = \frac{1}{1 + e^{-(V - V_{1/2})/\alpha}}$$

$$\tau_x = \tau_x(V; V_{1/2}, \delta)$$

Intracellular signaling and calcium

Calcium concentration importance (in contrast to that of sodium, potassium and chloride)

When neurons spike, it's the calcium (Ca2+) concentration that determine neurotransmitter release. While sodium (Na+), potassium (K+) and chloride (Cl-) ions are important for action potential propagation, it's often just their current that is interesting (and therefore what's modelled). While for the synapse, its the calcium concentration that it vital.

The intracellular concentration of calcium being $\sim 10^{-3}$ magnitudes smaller than other ions, calcium currents will affect the intracellular concentrations significantly. The reversal potential (Nernst potential) for calcium is

$$E_{Ca} = \frac{RT}{Z_{Ca}F} \ln \frac{[Ca]_{\text{out}}}{[Ca]_{\text{in}}}$$

Diffusion

Diffusion is a slow process, that doesnt move molecules very far. It's therefore important for short-distance transport inside cells (and less so for larger distances).

$$\frac{\partial c}{\partial t} = D\left(\frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} + \frac{\partial^2 c}{\partial z^2}\right)$$

$$= D\nabla^2 c$$

where D is the diffusion coefficient (constant) describing how fast the substance diffuses, and $\nabla^2 c$ indicating that the net movement is from high to low concentration.

Solution of diffusion equation with N molecules (e.g. Ca) starting at $t = 0$:

$$c(\vec{r}, t) = \frac{N}{(4\pi D t)^{3/2}} e^{-r^2/(4Dt)}$$

$$= \frac{N}{(\sqrt{2\pi}\sigma(t))^3} e^{-r^2/(2\sigma(t)^2)}$$

with a Gaussian distribution with time-dependent standard deviation $\sigma(t) = \sqrt{2Dt}$. □

Dynamics of calcium concentration

$$\frac{\partial[Ca^{2+}]}{\partial t} = J_{\text{diff}} - J_{\text{buff}} + J_{\text{cc}} - J_{\text{pump}} - J_{\text{up}} + J_{\text{rel}} - J_{\text{leak}}$$

J_{diff} Diffusion flux.

J_{buff} Calcium buffering. Interactions (and bindings) to other molecules in the intracellular space.

J_{cc} Ionic calcium current (electric) I_{cc}.

J_{pump} Membrane-bound pumps contributing to maintaining low resting-level calcium concentrations (by pumping against gradient). E.g. Ca-Na-pumps or Ca-ATPase-pumps (PMCA):

$$J_{\text{pump}} = V_{\text{pump}} \frac{[Ca^{2+}]}{K_{\text{pump}} + [Ca^{2+}]}$$

J_{up} Calcium uptake into intracellular stores like the *endoplasmic reticulum* (ER) via Ca-ATPase-pumps (SERCA):

$$J_{\text{up}} = V_{\text{up}} \frac{[Ca^{2+}]^2}{K_{\text{up}}^2 + [Ca^{2+}]^2}$$

2 because two ions are bound and transported per ATP molecule.

J_{rel} Calcium release from intracellular stores like the ER. Either

$$J_{\text{rel}} = V_{\text{rel}} R([Ca^{2+}]) \left([Ca^{2+}]_{\text{store}} - [Ca^{2+}]\right)$$

or

$$R([Ca^{2+}]) = \frac{[Ca^{2+}]^n}{K_{\text{rel}}^n + [Ca^{2+}]^n}$$

where n is a suitable Hill coefficient.

J_{leak} Leakage.

Brain tissue

Extracellular diffusion

Diffusion of molecules in the ECS follows the same physical laws as "standard" diffusion, and is therefore bound by the same equations.
The flux density is given by Fick's law in 3D, $\vec{J}_{X,diff} = -D_X^* \nabla[X]$, where X is the ion. The continuity equation therefore becomes

$$\begin{aligned}\frac{\partial[X]}{\partial t} &= -\nabla \vec{J}_{X,diff} + \frac{s_X}{\alpha} \\ &= D_X^* \nabla^2[X] + \frac{s_X}{\alpha}\end{aligned}$$

□

where s_X is a source term (ions added per tissue volume), scaled by the volume fraction α.

Volume conductor theory electrical conductivity

The electrical conductivity of the ECS is a measure of how efficiently the electrical signal is propagated (i.e. drifts).

The flux density of ion X due to 3D electrical drift is given by

$$\vec{J}_{X,drift} = -\frac{FD_X^*}{RT} z_X [X] \nabla V_e$$

where V_e denotes the ECS potential.

The current associated with ion X is proportional to the flux;

$$\vec{I}_{X,drift} = F z_x \vec{J}_{X,drift}$$

And the total current density (for all ion-types) is therefore the sum of these:

$$\begin{aligned}
\vec{I} &= \sum_X \vec{I}_{X,drift} \\
&= -\frac{F^2}{RT} \left(\sum_X D_X^* z_X^2 [X] \right) \nabla V_e \\
&= \sigma \vec{E}_e \\
&= -\sigma \nabla V_e
\end{aligned}$$

□

where the conductivity is defined as

$$\sigma = \frac{F}{RT} \sum_X D_X^* z_X^2 [x]$$

The extracellular potential from a point current source when sigma is isotropic and constant

Since V_e only changes in the radial direction, we can rewrite the current as

$$\vec{I} = -\sigma \nabla V_e \quad \Rightarrow \quad I(r) = -\sigma \frac{dV_e}{dr}$$

and insert this into $i_e = 4\pi r^2 I(r)$ we get

$$\begin{aligned}
\frac{dV_e}{dr} &= \frac{-i_e}{4\pi\sigma r^2} \qquad \text{defining } V_e(\infty) = 0 \\
V_e(r) &= \frac{i_e}{4\pi\sigma r}
\end{aligned}$$

which expands to

$$V_e(\vec{r}) = \sum_k \frac{i_{e,k}}{4\pi\sigma |\vec{r} - \vec{r}_k|}$$

□

for several point sources.

Instead of working with point sources, we introduce a current-source density

$$\begin{aligned}
c(\vec{r}, t) &= \nabla \vec{I} \\
&= -\sigma \nabla^2 V_e \qquad \text{where} \qquad \int\int\int_{\text{volume}} c dV = i_e
\end{aligned}$$

Electrodiffusion

The dynamics of an ECS concentration is governed by both diffusion and electrical drift of ions, i.e., it's of electrodiffusive nature.

Electrodiffusion is modelled by the Nernst-Plancks continuity equation with the flux-density containing both a diffusive- and drift-term.

$$\begin{aligned}\frac{\partial [X]_e}{\partial t} &= -\nabla \vec{J}_{X,diff} + \frac{s_X}{\alpha} \\ &= -D_X^* \nabla^2 [X]_e - \frac{F D_X^*}{RT} z_X [X]_e \nabla V_e\end{aligned}$$

□

To solve the equation (numerically), V_e must be defined. The two ways this may be done are:

(i) Poisson-Nernst-Planck (PNP) framework

In the Poisson-Nernst-Planck framework, the extracellular charge density ρ_e is expressed as a function of extracellular ion concentrations $[X]_e$, while ϵ is the permittivity of the medium.

$$\nabla^2 V_e = -\frac{\rho_e}{\epsilon} \qquad \text{where} \qquad \rho_e = F \sum_X z_X [X]_e$$

(ii) Electroneutral frameworks (e.g. Kirchhoff-Nernst-Planck (KNP))

An alternative to *PNP* rather derive V_e from the constraint that the bulk solution should remain electroneutral.

$$\frac{\partial \rho_e}{\partial t} = \begin{cases} 0 & \text{everywhere without membrane} \\ -c_{\text{cap}} \propto -\frac{dV_m}{dt} & \text{where there is membrane} \end{cases}$$

Multi-compartmental model

For a general multicompartmental model (of N compartments), the equation generalizes.

$$V_e(\vec{r}, t) = \frac{1}{4\pi\sigma} \sum_n^N \frac{I_n(t)}{|\vec{r} - \vec{r}_n|}$$

$$\sum_n^N I_n(t) = 0$$

Integrate-and-fire neuron

The integrate-and-fire neuronal models, as the name explains, accumulates charge/potential, and fires when above a threshold.

The sub-threshold membrane dynamics

$$C_m \frac{dV}{dt} = -\frac{V - E_m}{R_m} + I$$

$$\tau_m \frac{dV}{dt} = -V + E_m + R_m I$$

Where the neuron fires when $V \to E_m + \Theta$, before reset to E_m.

f-I curve with an absolute refractory period

$$f(I) = \frac{1}{\tau_r - \tau_m(1 - \Theta/R_m I)}$$

Where τ_r is the absolute refractory period.

Conductance-based synapse

$$I_{\text{syn}}(t) = g_{\text{syn}}(t)(V(t) - E_{\text{syn}})$$
$$g_{\text{syn}}(t) = \bar{g}_{\text{syn}} e^{-(t-t_s)/\tau_{\text{syn}}} \Theta(t - t_s)$$

Current-based synapse

$$I_{\text{syn}}(t) = \bar{I}_{\text{syn}} e^{-(t-t_s)/\tau_{\text{syn}}} \Theta(t - t_s)$$

Integrate-and-fire models

Leaky integrate-and-fire **LIF**

$$C_m \frac{dV}{dt} = I - \frac{V - E_m}{R_m}$$

Quadratic integrate-and-fire **QIF**

$$C_m \frac{dV}{dt} = I - \frac{(V - E_m)(V_{\text{thresh}} - V)}{R_m(V_{\text{thresh}} - E_m)}$$

Exponential integrate-and-fire **EIF**

$$C_m \frac{dV}{dt} = I - \left(\frac{V - E_m}{R_m} - \frac{\Delta_T}{R_m} e^{(V - V_T)/\Delta_T} \right)$$

Firing-rate function

The firing-rate function tells what the steady-state firing rate is given a constant input current.

Piece-wise linear	$f(I) = k(I - \Theta)$	for $I \geq \Theta$, else 0
Sigmoid	$f(I) = \bar{f}/(1 + e^{-k(I-\Theta)})$	
Step	$f(I) = \bar{f}$	for $I \geq \Theta$, else 0
Static feed-forward	$I_j = \sum_i w_{ij} f_i$	where $f_j = f(I_j)$
Dynamic	$\tau_{\text{syn}} \frac{dI_j}{dt} = -I_j + \sum_i w_{ij} f_i$	where $f_j = f(I_j)$

Amit and Brunel network

Recurrent network of excitatory and inhibitory integrate-and-fire neurons. Here, N_E is the amount of excitatory neurons and N_I the number of inhibitory neurons.

$$\frac{dV_j}{dt} = -\frac{V_j}{\tau_j} + I_j^{\text{rec}} + I_j^{\text{ext}}$$

$$\text{where} \qquad I_j^{\text{rec}}(t) = \sum_{i \in E,k}^{N_E} c_{ij} w_{ij} \delta(t - \tau_{ij} - t_i^k) - \sum_{i \in I,k}^{N_I} c_{ij} w_{ij} \delta(t - \tau_{ij} - t_i^k)$$

Where c_{ij} is a random binary variable 0 or 1 denoting whether the connection exists, w_{ij} a randomly drawn Gaussian weight, τ_{ij} a randomly drawn delay $[0.5, 1.5]ms$ and I_j^{ext} the Poisson input.

Exercises

Exercise 1 — Biophysical properties of cell membranes

Calculations by hand.

Exercise 2 — Leaky integrate-and-fire: LIF

Plotting $F(I)$-curve based on formula.

Numerically obtaining firing-rate

```
R = 40e-3    # Membrane resistance
tau_m = 10.  # Time-constant
theta = 15.  # Threshold potential
h = 0.1
T = 1000
v_init = 0
v_reset = 0

num_tsteps = int(T / h + 1)
t = np.linspace(0, T, num_tsteps)
v = np.zeros(num_tsteps)

v[0] = v_init
if noisy:
    I = I0 * (np.ones(num_tsteps) + np.random.normal(0, 0.4, size=
        num_tsteps))
else:
    I = I0 * np.ones(num_tsteps)

for n in range(num_tsteps - 1):
    v[n + 1] = v[n] + h / tau_m * (-v[n] + R * I[n])
    if v[n + 1] >= theta:
        v[n + 1] = v_reset
```

Exercise 3 — Hodgkin-Huxley model

Setting parameters for the neuron, and adding stimuli.

Adding stimuli

```
    I = np.zeros(len(p['time']))

    for i, t in enumerate(p['time']):
        if p['t_stim_on'] <= t <= p['t_stim_off']:
            I[i] = p['I_amp']
```

Updating state

```
            g_Na = p['gbar_Na']*(m**3)*h
            g_K = p['gbar_K']*(n**4)
            g_l = p['gbar_l']
            m += (p['alpha_m'](Vm)*(1 - m) - p['beta_m'](Vm)*m) * p['dt']
            h += (p['alpha_h'](Vm)*(1 - h) - p['beta_h'](Vm)*h) * p['dt']
            n += (p['alpha_n'](Vm)*(1 - n) - p['beta_n'](Vm)*n) * p['dt']
            Vm += (I - g_Na*(Vm - p['E_Na']) - g_K*(Vm - p['E_K']) - g_l*(Vm - p
                ['E_l'])) / p['Cm'] * p['dt']
```

Where

- m, h, n : Activation of channels
- V_m : Membrane potential

Exercise 4 — Ball-and-stick neuron

```
        soma = nrn.Section('soma')
        soma.L = 15  # um; stored as a float number
        soma.diam = 15  # um
        soma.nseg = 1  # stored as an integer

        dend = nrn.Section('dend')
        dend.L = 1000
        dend.diam = 2
        dend.nseg = int(dend.L/10)

        dend.connect(soma, 1, 0)
```

Here a soma and dendrite is created. The soma is one section, and the dendrite 1000/10 compartments. These are then connected. The "start" (0) of the dendrite is connected to the "end" (1) of the soma.

```
        for sec in nrn.allsec():
            sec.insert('pas')
            sec.Ra = 100
            sec.cm = 1
            for seg in sec:
                seg.g_pas = 0.00003
                seg.e_pas = -65
```

Each section then is defined with passive channels, resistance and membrane capacitance. Each segment then gets its own passive conductance and reversal potential defined.

```
        stim = nrn.IClamp(input_site)
        stim.delay = 10
        stim.amp = amp
        stim.dur = dur
```

A current clamp is then inserted at the input-size. If *input_size = soma(0.5)*, the clamp is inserted in the middle of the soma. The duration and amplitude of the current pulse is defined, as well as after how long time it is inserted.

Exercise 5 — Hay model

```
        h('forall delete_section()')
        model_path = 'hay_model'
        #neuron.load_mechanisms(join(model_path, 'mod'))
        cell_parameters = {
```

```
        'morphology': join(model_path, 'cell1.hoc'),
        'v_init': -65,
        'passive': False,
        'nsegs_method': 'lambda_f',
        'lambda_f': 100,
        'dt': 2**-3,  # Should be a power of 2
        'tstart': -200,
        'tstop': 200,
        'custom_code': [join(model_path, 'custom_codes.hoc')],
        'custom_fun': [active_declarations],
        'custom_fun_args': [{'conductance_type': conductance_type}],
    }
    cell = LFPy.Cell(**cell_parameters)
    synapse_parameters = {
        # Returns compartment on cell closest to coordinates:
        'idx': cell.get_closest_idx(x=0., y=synaptic_y_pos, z=0.),
        'e': 0.,
        'syntype': 'ExpSyn',
        'tau': 10.,
        'weight': weight,
        'record_current': True,
    }
    synapse = LFPy.Synapse(cell, **synapse_parameters)
    synapse.set_spike_times(input_spike_train)
    cell.simulate(rec_imem=True, rec_vmem=True)
```

- **v_init**: This parameter represents the initial membrane potential of the neuron in millivolts (mV). In your code, v_init is set to -65, which is a typical resting membrane potential for many neurons. This value is crucial as it sets the starting electrical condition of the neuron before any simulation of neural activity begins. It's the baseline from which all changes in membrane potential are measured during the simulation.
- **e**: This parameter appears in the context of synapse parameters. It represents the reversal potential (in mV) for a particular synaptic conductance. In this case, e is set to 0.0, which could be indicative of a synaptic reversal potential for excitatory inputs (like those mediated by glutamatergic synapses) under certain conditions. The reversal potential is the membrane potential at which the net flow of the specific ions (that the synapse is permeable to) is zero. It's a critical factor in determining whether a synaptic input will be excitatory or inhibitory.

Exercise 6 — T-type Ca^{2+}-channels

Setting up the neuron. Conductance for potassium (gkbar_hh2) and sodium (gnabar_hh2) is initially set to zero for all sections other than somas.

```
    for sec in cell.allseclist:
        sec.v = Epas
        sec.e_pas = Epas
        sec.insert("pas")
        sec.e_pas = Epas
        sec.g_pas = 1/Rm
        sec.Ra = rall
        sec.cm = cap
        sec.gnabar_hh2 = 0
        sec.gkbar_hh2 = 0
        if sec.name().rfind('soma') >= 0:
            sec.gnabar_hh2 = gna
            sec.gkbar_hh2 = gkdr
```

Modifies the conductances based on the "tdist"-input.

```
    if tdist == 1:
        for sec in cell.allseclist:
```

```
                sec.gcabar_it2 = gcat
        elif tdist == 2:
            for sec in cell.allseclist:
                if sec.name().rfind('soma') >= 0:
                    sec.gcabar_it2 = gcat*0.1054
                else:
                    for seg in sec:
                        seg.gcabar_it2 = gcat*0.1054*(1 + 0.04 * h.distance(seg.x,
                            sec=sec))
        elif tdist == 3:
            for sec in cell.allseclist:
                sec.gcabar_it2 = gcat
                if sec.name().rfind('soma') >= 0:
                    sec.gnabar_hh2 = gna
                    sec.gkbar_hh2 = gkdr
                else:
                    sec.gnabar_hh2 = gna*0.05
                    sec.gkbar_hh2 = gkdr*0.05
```