# Ions demo

*Balazs B Ujfalussy*

*11/09/2017*

This is a demo for simulating the buildup of the equilibrium potential in a simple neuron. We use the package deSolve to simulate the differential equations.

Simple physical laws decribe the forces - electrical and chemical - that control the movements of the ions through the membrane. Here we will simulate these forces (in their simplified form) using differential equations. The basic idea is the following:

- the internal and the external concentrations of an ion define the *Nernst potential* of that ion, i.e., $E_K \propto \log \frac{C_{\text{out}}}{C_{\text{in}}}$ where $\propto$ means that we omitted some constants and log is the logarithm.
- the movement is driven by the *driving force*, that is the difference between the membrane potential and the Nernst potenital: $(E_K - V)$
- we assume that the flow of the ion is dictated by *Ohm's law*, so it is the product of the driving force and the conductance: $I_K = g_K(E_K - V)$
- the current changes the concentration of the ions inside and outside, and thus influences the Nernst potential of the ion
- the current (measured in Ampere = Coulumb per secundum) provides the rate of the change of the membrane potential, but also the changes of the ionic concentrations
- taken together we have a *differential equation* that connects the rate of change of something (the concentrations and the membrane potential) with some function of their actual value ($E_K$ and $V$).

We will simulate this differential equation by iteratively calculating the ionic currents from the actual values of the ionic concentrations, then updating the concentrations by the amount of ions that crossed the membrane.

## 0. Preparations

First we will load a couple of useful functions. These functions will be used later to simulate the movements of the ions through the cell membrane. The parameters of our model cell is defined in the ions_const.R file, which is also loaded by the ions_sim.R file.

```
source('./ions_sim.R', chdir=T)
```

Both electric and chemical changes are simulated. The goal is to compare the time course and magnitude of the changes under different conditions.

## 1. Illustrating the Nernst equation

We will simulate the flux of K ions through a simple membrane permeable only to K ions. Initially there are no electric potential difference between the two sides of the membrane but there is a difference in the ionic concentrations.

```
# to simulate the system, we need to provide the parameters of out model cell.
# parameters are stored in a vector, called params
# here the name 'gK' is assigned with the value of the variable gK defined in the parameter file.
params <- c(gK=gK, cm=cm, vi=vi, ve=ve) # parameters of the system. Parameters are:
# gK: K conductance in mS,
# cm: membrane capacitance in uF
# vi and ve: the intracellular and extracellular volume in cm3.
# For more details, see the demos/ions_consts.R file
```

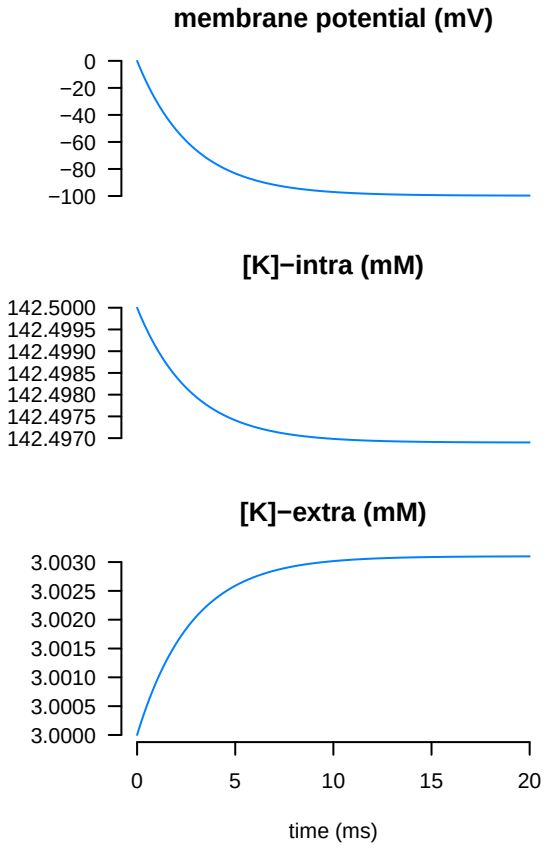
```r
# or use the command 'print(params)'

# we also need the provide initial values for the variables of the system -
# the membrane potential and the initial K concentrations inside and outside the cell
state <- c(v=0, C.Ki=C.Ki.init, C.Ke=C.Ke.init) # initial state of the system. State variables are:
# v - membrane potential - mV
# C.Ki - intracellular K concentration in mM
# C.Ke - extracellular K concentration in mM

# we define the time scale of the simulations -
# times is the vector containing the time points at
# which we are interested in the values of the variables
times <-  seq(0,20, by=1/10) # the time axis of the simulations

# the ode (= ordinary differential equations) is a function that simulates the differential equation
# its arguments are the initial state - state
# the time points for the output
# the description of the system to simulate - sim.Nernst function defined in the ions_sim.R file
# the parameters of the system
out <- ode(y = state, times = times, func = sim.Nernst, parms = params)
# out is a big matrix (table) with the values of the variables at the required time points
```

Now, we plot the resulting change in the membrane potential and the change in the ionic concentrations.



**Homework problem**

Interpret the results:

1. What are the magnitudes of changes you see in the membrane potential and in the K concentration? Do you think that these changes are large or small? Express the changes in the percentage of the final value! [2p]
2. What is the time scale required to achieve steady state - the state where the variables don't change any more? [2p]

Run new simulations:

3. Change some of the parameters of the system (e.g., change `gK`, `cm` or `C.Ki` by multiplying or dividing them with 2). Observe their effect on the steady state of the system or on the time required to achieve steady state. [4p]

## 2. Equilibrium potential

Next we simulate the buildup of the resting potantial of a neuron. The difference between this and the previous simulation is that now we simulate two ions (Na and K) and also an ion pump (K/Na pump).

Initially there are no electric or chemical gradients, since there are equal amount of K and Na ions on the two sides of the membrane. The membrane contains ion channels for both both Na and K, but the permeability is different for the two ions. Importantly, in this simulation the permeability for Na and K is constant.

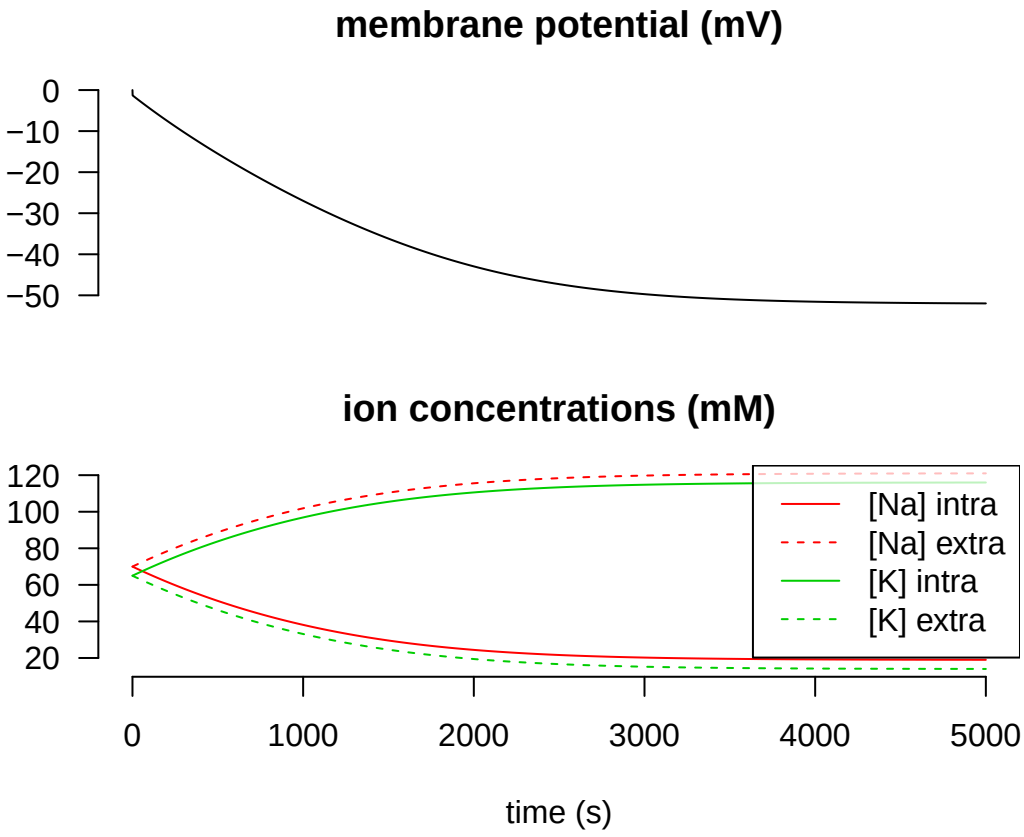
We also simulate the action of the K/Na pump, that uses energy (ATP) to move Na ions out of the cell and K ions into the cell. At t=0, we switch on the Na/K exchanger, that starts transporting ions.

```
# simulation
params <- c(gK=gK, gNa=gNa, cm=cm, vi=vi, ve=ve, I.pump.K=I.pump.K, I.pump.Na=I.pump.Na)# simulation pa
# gK, gNa: K and Na conductance in mS,
# cm: membrane capacitance in uF
# vi and ve: the intracellular and extracellular volume in cm3.
# I.pump.K, I.pump.Na: the current mediated by the Na/K exchanger in uA
# For more details, see the demos/ions_consts.R file
state <- c(v=0, C.Nai=70, C.Nae=70, C.Ki=65, C.Ke=65)
# initial state of the system. State variables are:
# v - membrane potential - mV
# C.Ki, C.Nai - intracellular K and Na concentration in mM
# C.Ke, C.Nae - extracellular K and Na concentration in mM

times <-  seq(0, 5000000, by=1000) # in ms!
out2 <- ode(y = state, times = times, func = sim.equilibrium, parms = params)

## we set the membrane potential to 0 to see how quickly it returns to the old value
st <- out2[3501,2:6]
st[1] <- 0
times <-  seq(0, 50, by=1) # in ms!
out3 <- ode(y = st, times = times, func = sim.equilibrium, parms = params)
```

Now look at the results



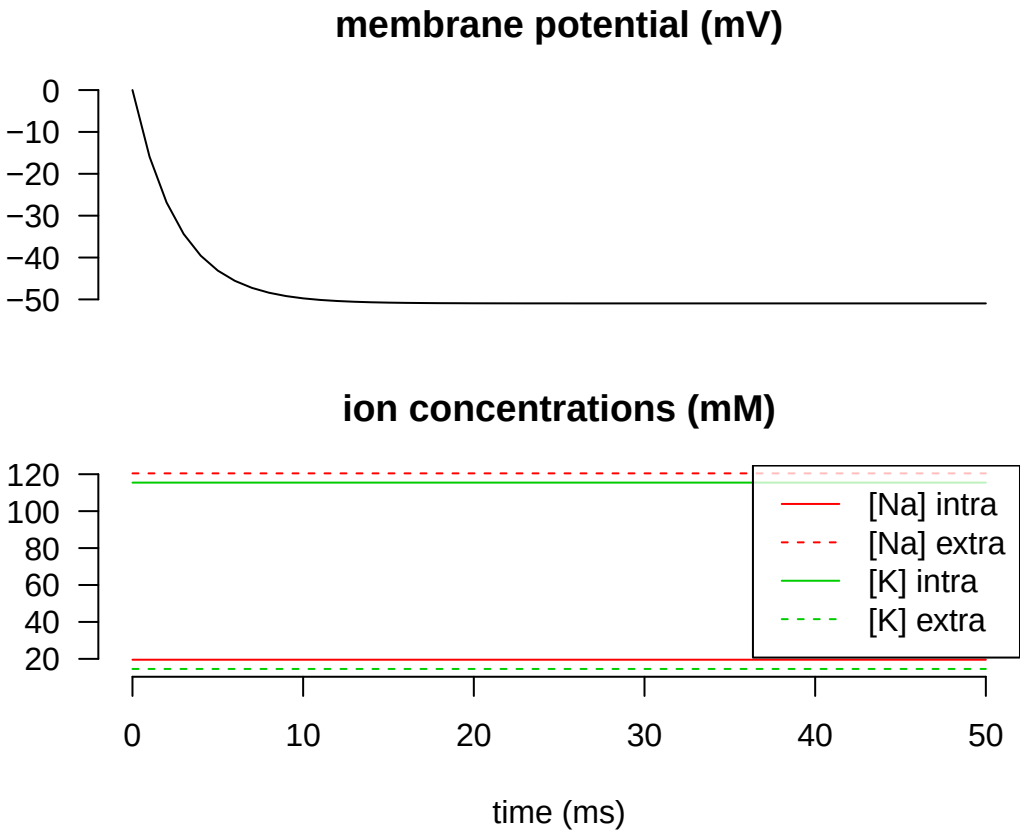

**Homework problem**

Interpret the results:

4. What are the magnitudes of changes you see in the membrane potential and in the K concentration? Do you think that these changes are large or small? Express the changes in the percentage of the final value! [1p]
5. What is the time scale required to achieve steady state - the state where the variables don't change any more? Compare it with the time scale you observed in the first simulation and in Section 3 (Repolarisation). [1p]
6. The equilibrium potential is sometimes approximated as $\tilde{V} = \frac{g_{\mathrm{K}}\, E_{\mathrm{K}} + g_{\mathrm{Na}}\, E_{\mathrm{Na}}}{g_{\mathrm{K}} + g_{\mathrm{Na}}}$. Check the accuracy of the approximation and explain the difference! [4p]
7. Initially there is a small, but quick drop in the membrane potential (remember, we started from 0 mV!). What could be the cause of this? Can you eliminate this by changing a single number in the code of the function `sim.equilibrium()`? What did you change and how did this eliminate the drop? [4p]
8. What happens if, after reaching the steady state, you switch off the Na/K pump? [2p]

## 3. Repolarisation

After reaching the steady state we can change the membrane potential to see how quickly it returns to the steady state.


membrane potential (mV)
0
−10
−20
−30
−40
−50
ion concentrations (mM)
120
100
80
60
40
20
[Na] intra
[Na] extra
[K] intra
[K] extra
0
10
20
30
40
50
time (ms)