Derivation of cell parameters

September 16, 2020

Ilaria Urbani, University of Padova

1 Abstract

In this report, a way for estimation of cell parameters such as membrane capacitance, membrane resistance and access resistance in whole-cell configuration is explained. This method is limited to cases where the cell-pipette system can be approximately described by a simple three-component circuit with linear properties. The parameters are derived from an analysis of capacitive transients during a square wave stimulation of $V = 10$ mV. The simulation is carried out with two different values of resting potential ($V_r = 0$ mV and $V_r = -80$ mV). Moreover a discussion about the application of 4-pole low-pass Bessel filter and a comparison between various cut-off frequency is presented.

2 Introduction

2.1 Membrane voltage and ionic currents

The membrane voltage V_m, or rather, the voltage across the cell membrane is the most fundamental electrophysiological property of almost all the cells. It is defined as the difference in electric potential between inside the cell (ϕ_{intra}) and outside the cell (ϕ_{extra}):

$$V_m = \phi_{intra} - \phi_{extra}$$

Most of the time, it is assumed that ϕ_{extra} remains constant and that any change to V_m is the result of changes to ϕ_{intra}. If there is not an external input, the value of V_m is called the resting potential. Ionic currents, that flow through the ion channels placed in the membrane, modulate the membrane voltage because they change the electric potential ϕ_{intra} inside the cell. The electric circuit model of an ion channel is a resistor with resistance R_m and in series a voltage source whose strength is the equilibrium (or resting) potential.

2.2 Voltage clamp: whole-cell configuration

The patch clamp recording is a technique used in electrophysiology to study ionic currents in individual isolated living cells, tissue sections, or patches of cell membrane. Two types of techniques can be used, depending on the aim of the experiment: current clamp and voltage clamp. In the first, the current passing across the membrane is maintained at a given value and the resulting changes in voltage are recorded; in the second, the voltage across the cell membrane is controlled from the outside and the resulting currents are recorded. The value of the membrane voltage is called V_m and an amplifier injects the current needed to balance any change to this voltage. Keeping the membrane voltage constant by the voltage clamp, it is possible to determine the time course of the ionic current as a function of the membrane voltage.

Various patch-clamp configurations can be used: for example cell-attached enables the recording of single-channel currents; whole-cell allows the control of intracellular potential, to see currents flowing through the totality of the cell membrane or to deliver ions and molecules into the cell. The whole-cell configuration, enabling continuity between the pipette solution and the cytoplasm, has the advantage that it makes possible to control the intracellular ionic concentration and the delivery of chemicals and peptides to the interior of the cell. However, it has also a disadvantage because it dilutes the cytoplasm, altering any response that depends on soluble intracellular second messengers or ionic gradients. Furthermore, the loss of cytoplasmic components, essential for maintaining the channel activity, normally results in progressive decrease of recorded currents.

2.3 Why use patch clamp?

The patch-clamp technique is used to observe in real time the variations in the activity of a single channel, like changes in conductance (the rate of ions going through the channel) and kinetic properties (the speed of opening and closing a channel) in response to a stimulus. In addition, through the patch-clamp technique, the detailed characteristic sensitivities of a multitude of ion-channel types have been discovered, giving information about the physiological modulation of these channels and also their alterations in disease states. Moreover, this technique is very flexible because it permits to record ionic currents in a range between 0.1-1.0 pA (a single channel transition) and 1-10 nA (the whole-cell current) and this is the reason why it is so widespread in electrophysiology.

3 Methods

The simulation is carried out using Matlab2020a. Initially the resting potential is set to zero for the first simulation and to $V_r = -80$ mV for the second; at $t_1 = 0.5$ ms, a voltage step of V=+10 mV begins, lasting 4.5 ms (until $t_2 = 5$ ms), after which I wait for the current signal to return to baseline. A noise (as explained in Section 3.1) is added to the simulated current, in order to reproduce better a signal that could be obtained by an experiment in lab. The sampling frequency is set to $f_{sam} = 100\ kHz$.

To model the whole-cell configuration resistances, capacitance and batteries can be used. If the whole-cell configuration is correctly achieved, R_L (leakage current) should be negligible and C_P should have been compensated by amplifier electronics at the bath phase; therefore the simplified circuit is that which can be seen in Figure 1.

The initial values set for resistances and capacitance:

- $R_a = 10\ M\Omega$
- $R_m = 100\ M\Omega$
- $C_m = 30\ pF$

The value of the membrane voltage V_m is related to V:

$$V_m = \frac{R_m}{R_m + R_a} V$$

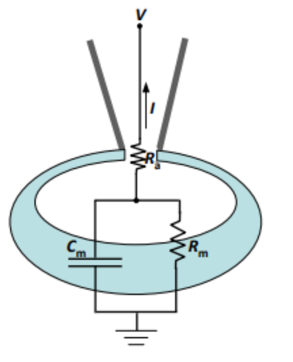

Figure 1: Simplified model for whole-cell configuration

Using Kirchoff's laws:

$$V = iR_a + V_m$$

$$\frac{R_a R_m}{R_a + R_m} C_m \frac{dV_m}{dt} = -V_m + V \frac{R_m}{R_m + R_a}$$

that is a first-order linear differential equation with a time constant:

$$\tau = \frac{R_a R_m}{R_a + R_m} C_m$$

simulated in the program (Appendix). The analytical solution is:

$$i_{(t_1,t_2)} = V \cdot \frac{1 + \frac{R_m}{R_a} \cdot e^{-\frac{(t-t_1)}{\tau}}}{R_a + R_m}$$

if $t - t_1 >> \tau$

$$i_\infty = \frac{V}{R_a + R_m}$$

From the experimental data, namely the current to which the noise has been added, these quantities can be extrapolated:

$$\begin{cases} R_a = \frac{V}{i_{peak}} \\ R_m = \frac{V}{i_\infty} - R_a \\ C_m = \tau \cdot \left(\frac{1}{R_a} + \frac{1}{R_m} \right) \end{cases}$$

The value of relaxation time (τ) can be obtained by fitting the exponential decay of $i(t)$. On the other hand, i_{peak} is simply the maximum of the current and i_∞ derives from the average of the final points (just before t_2).

For the second simulation a resting potential of V_r =-80 mV is introduced. The analysis of the signal is the same, but there are some changes in the formulas, taking into account V_r:

$$i_{(t_1,t_2)} = \frac{V - V_r}{R_a + R_m} + \frac{V \cdot \frac{R_m}{R_a} \cdot e^{-\frac{(t-t_1)}{\tau}}}{R_a + R_m}$$

if $t - t_1 >> \tau$

$$i_\infty = \frac{V - V_r}{R_a + R_m}$$

and

$$\begin{cases} R_{a,r} = R_{a,r} \cdot \frac{V - V_r}{V} \left(\frac{i_{peak}}{i_\infty} - 1 \right) = R_{a,r} \cdot c \\ R_{m,r} = \frac{V - V_r}{i_\infty (c+1)} \\ C_{m,r} = \tau \cdot \left(\frac{1}{R_{a,r}} + \frac{1}{R_{m,r}} \right) \end{cases}$$

Finally, a 4-pole low-pass Bessel filter with various cut-off frequencies is applied and the cell parameters are derived again. However, since the filtering procedure alters the shape of the current transient, the computation of R_a is different; another strategy is used that do not rely on the computation of i_{peak}:

$$R_a = \frac{\tau \cdot V}{Q + \tau \cdot i_\infty} \tag{1}$$

where Q is the charge moved by the capacitative current i_Q:

$$Q = \int_0^\infty i_Q \cdot dt$$

but this Q is a good approximation of the real charge only when $R_m >> R_a$ (so i_∞ is small). Otherwise Q should be estimated in another way, taking into account the properties of the Bessel filter, namely linearity and conservation of the total charge moved at and from the capacitance

$$i = Bess(i_{real}) = Bess(i_Q + i_\infty) = Bess(i_Q) + Bess(i_\infty)$$

$$Q_{real} = \int_0^\infty i_Q \cdot dt = \int_0^\infty Bess(i_Q) \cdot dt = \int_0^\infty i \cdot dt - \int_0^\infty Bess(i_\infty) \cdot dt$$

Therefore the correct estimation of Q requires subtraction of an additional term obtained from the temporal integration of the Bessel-filtered stationary current i_∞. Since there are not references that explain this formula, I demonstrated it in Section 8.

Concerning the errors, the mean error ($\frac{\sigma}{\sqrt{N}}$) is associated to i_∞, V is assumed to have null error and the other errors are computed by means of propagation.

3.1 Noise

Noise can be defined as any disturbance that interferes with the measurement of the signal. In electrophysiological measurements, these disturbances derive from the preparation itself, the electrodes that couple the preparation to the measurement instrument, the electronic instrumentation, interference from external sources, and from the digitization process. The noise associated with the whole cell configuration is dominated by current noise arising from the pipette resistance R_a, in conjunction with the membrane capacitance C_m. The power spectral density S^2_{wc} of the current noise is given by:

$$S^2_{wc} = \frac{4\pi^2 f^2 4k_B T R_a C_m^2}{1 + 4\pi^2 f^2 (R_{res} C_m)^2} \ A^2/Hz$$

where R_{res} is the residual (uncompensated) series resistance. For the first simulation both 90% and 70% of compensation levels are used, to understand better the difference in noise levels and the difficulties in estimating the values of resistances and capacitance in this two cases. The compensation is an electronic trick that can reduce the electrode resistance. This mechanism involves positive feedback and does change cell behaviour. The degree of positive feedback is adjusted as a percentage of the electrode resistance that was previously estimated by setting the cancellation controls, adding a fraction of the output to the command potential. For the second simulation, only 90% of compensation is used.

Since many noise processes have a Gaussian distribution of instantaneous amplitudes versus time, a Gaussian noise is applied to the simulated current. The other contributions of the noise are assumed to be null, because interference from external sources can be almost completely eliminated in a well-designed system and noise arising from digitization is often ignored, justifying that it can be negligible with respect to other sources of noise.

4 Results

The simulated current, with two different noises (as explained in Section 3.1) is shown in Figure 2. As can be seen, the more is the compensation, the more is the current noise.

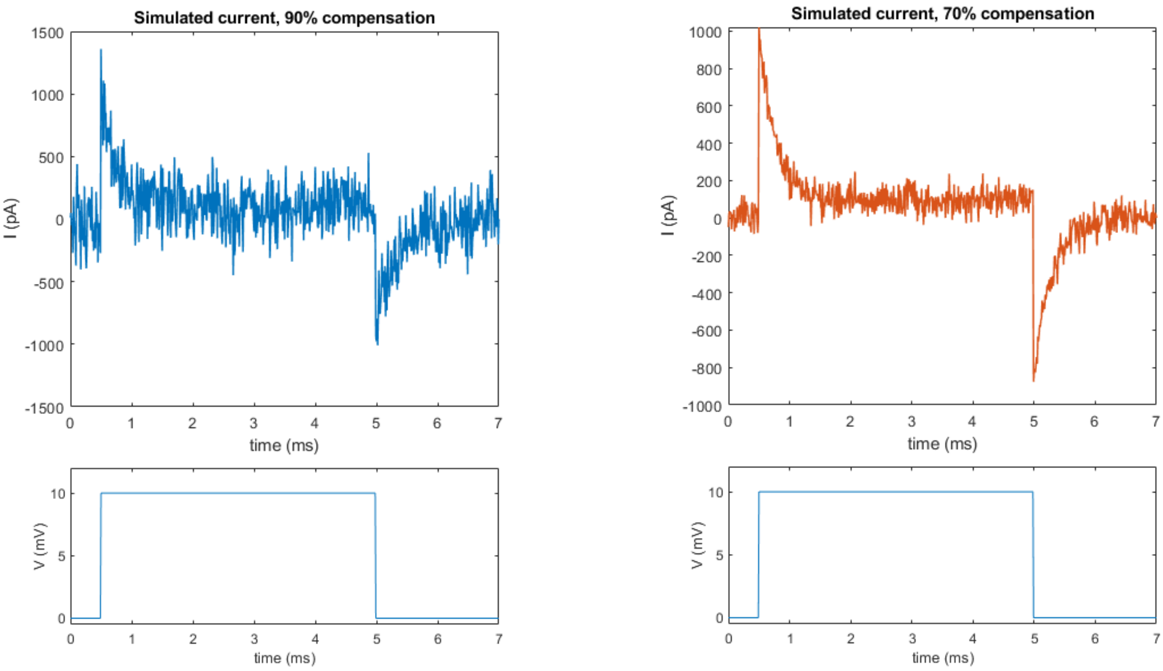

Figure 2: Simulated current and voltage

The values of the access resistance, membrane resistance and membrane capacitance are reported in Table 1. The computation of these quantities is performed according to what is written in Section 3, in particular the exponential fit to find the capacitance starts from $t_i = 0.52$ ms and lasts until $t_f = 1$ ms that is most of the exponential decreasing.

	Compensation: 90%	Compensation: 70%
R_a $(M\Omega)$	10 ± 1	10.0 ± 0.6
R_m $(M\Omega)$	108 ± 28	101 ± 8
C_m (pF)	29 ± 5	32 ± 4

Table 1: Quantities esteem for different compensations

Initially, another procedure to compute the value of the capacitance was used:

$$C_m = -t \cdot \left(\frac{1}{R_a} + \frac{1}{R_m} \right) \cdot \left[ln \left(\frac{i(R_a + R_m) - V}{V \dot{R}_m / R_a} \right) \right]^{-1}$$

however this was not accurate due to the noise added to the simulated current, in particular for the trace more noisy; for this reason the results are not reported.

For the second simulation, the procedure is the same as before, except for the introduction of the resting potential (Figure 3 and Table 2).

R_a $(M\Omega)$	10.1 ± 0.3
R_m $(M\Omega)$	98 ± 5
C_m (pF)	29 ± 2

Table 2: Quantities valued for second simulation (namely with resting potential)

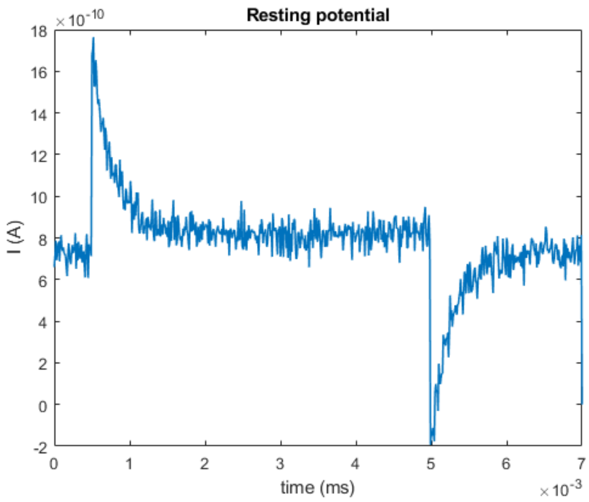

Figure 3: Simulation with resting potential

4.1 Filtering

Most patch-clamp amplifiers are equipped with 4-pole low-pass Bessel filters that are of sufficient quality to filter data, eliminating the high frequency components from the record. In theory, signals can be sampled at the highest possible frequency supported by the electronics. However there are practical limitations that cause the sample rate to be around 100-300 kHz. Since the sampling frequency is high, the signal has to be filtered with a reasonable bandwidth; this bandwidth is selected to reduce noise to acceptable levels so that the desired signal can be adequately observed. In many electrophysiological measurements wide bandwidths result in background noise levels that obscure the signals of interest. In those situations, it is necessary to make compromises between the amount of noise that can be tolerated and the time resolution that is achieved after filtering. For this reason, normally the cut off frequency is set between 10 and 1 kHz.

As can be seen in Figures 4 and 5, applying a filter results in a drastic reduction of the peak amplitude of the fast capacitance current transient and in a much slower time course of the remaining signal and this effect is amplified as the cut-off frequency decreases. Therefore the resulting i_{peak} value is too low and, consequently, computing R_a would produce a value that is too high. For this reason, a derivation of R_a independently from i_{peak} can be performed by the formula 1 in Section 3. The values computed for the specific simulations shown in Figures 4 and 5 are reported in Table 3.

90% compensation				
cut-off frequency (kHz)	10	5	2	1
R_a ($M\Omega$)	9 ± 1	9 ± 1	10.3 ± 0.7	11.2 ± 0.5
R_m ($M\Omega$)	104 ± 2	100 ± 5	102 ± 4	98 ± 2
C_m (pF)	31 ± 2	30 ± 2	30 ± 1	29 ± 1
70% compensation				
cut-off frequency (kHz)	10	5	2	1
R_a ($M\Omega$)	9.3 ± 0.5	9.2 ± 0.2	10.2 ± 0.7	11.1 ± 0.6
R_m ($M\Omega$)	102 ± 4	100 ± 6	101 ± 4	97 ± 2
C_m (pF)	31 ± 2	31 ± 1	30 ± 1	29 ± 1

Table 3: Values for various cut off frequencies

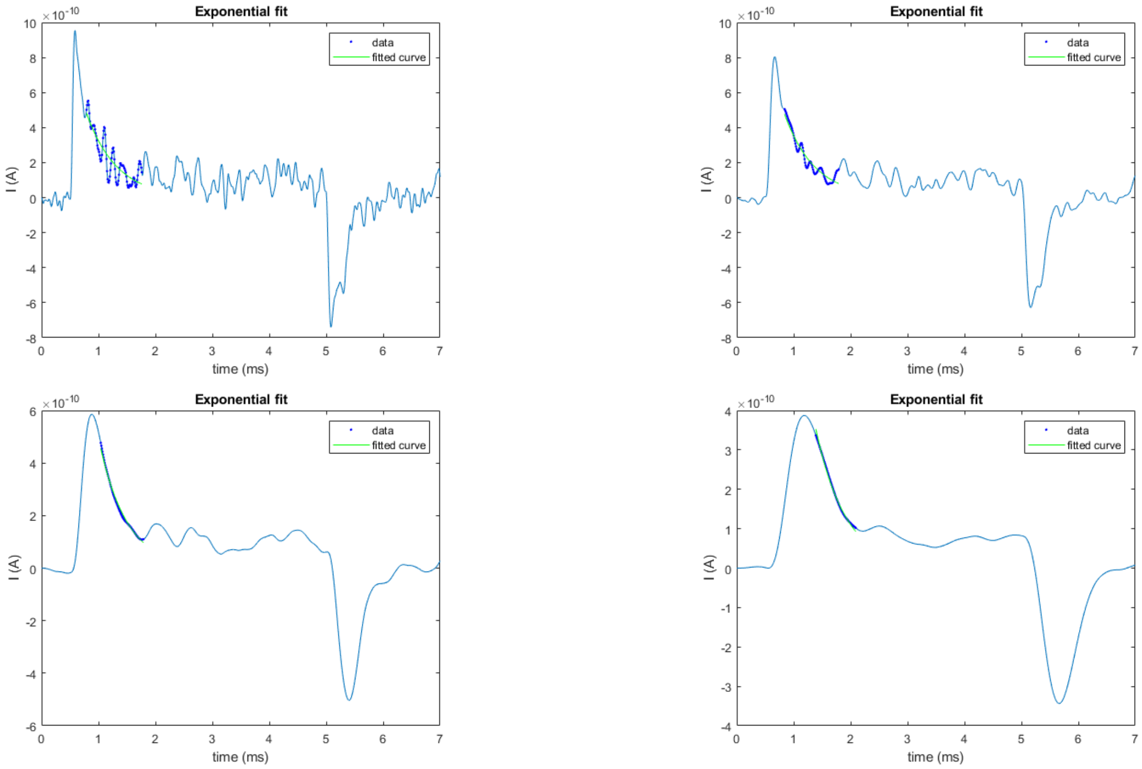

Figure 4: Comparison between various cut-off frequencies with 90% compensation. From top left to down right: 10 kHz, 5 kHz, 2 kHz, 1kHz

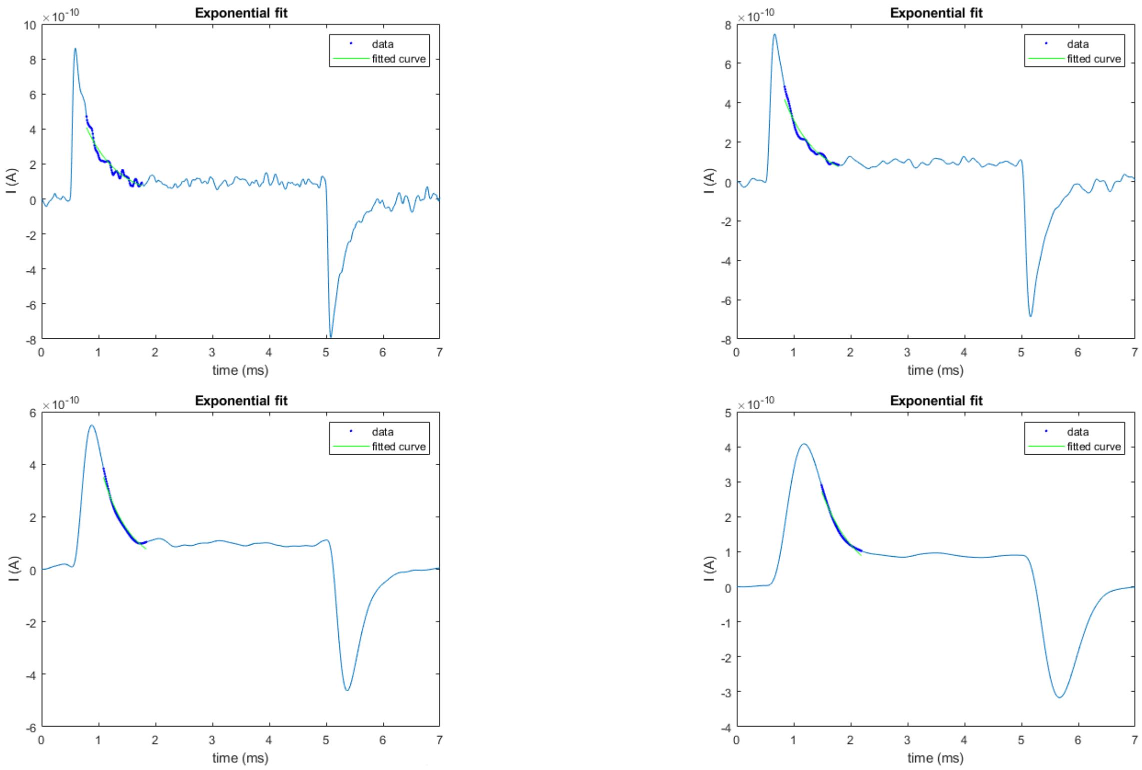

Figure 5: Comparison between various cut-off frequencies with 70% compensation. From top left to down right: 10 kHz, 5 kHz, 2 kHz, 1kHz

5 Discussion

Noise consequences

The values and the errors found in the previous section depend on the noise that is added to the simulation. In Figure 6 it is possible to visually see the difference between 90% and 70% of compensation. Considering the values reported in Table 1, the difference concerning errors is noticeable, since the seconds (70% compensation) are markedly lower than the others: this is due to the fact that increasing the compensation, the noise increases. Therefore, basing on this simulation, it is possible to understand that during patch-clamp recording, it is very important to balance the compensation and the noise produced. Since the errors are lower for 70% compensation, the second esteems are considered the most accurate and this is confirmed by the fact that these values are the nearest to the ideal ones.

Resting potential consequences

Introducing a resting potential affects the simulation in the sense that the current is higher than that computed setting $V_r = 0$, although the overall trend is the same. This is shown in Figure 3, in which the noise seems lower than the previous: this is not true, because its absolute value is the same as 70% compensation, but the ratio S/N (signal/noise) is lower. The direct effect is that the errors associated to resistances and capacitance are slightly smaller.

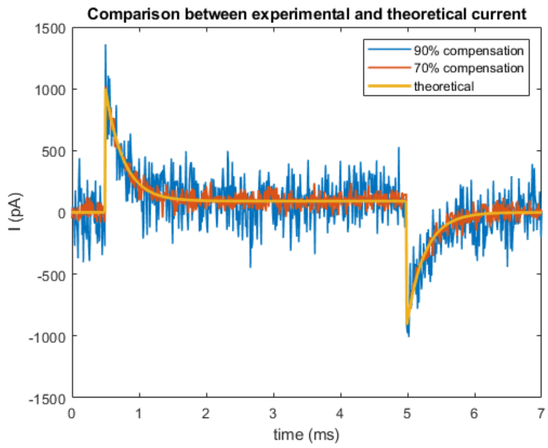

Figure 6: Comparison between compensations. As can be seen the simulation result match perfectly the analytical solution

Cut-off frequency consequences

Comparing the filtered signals (Figure 7), it is evident that the lower is the cut-off frequency, the higher is the smoothing. The shift as well as the height of the peak depends on the cut-off frequency. Even if the function of the filter is to reduce the noise, producing a trace that is more simple to be analyzed, the drawback of this procedure is the lost of some information about the signal and the quantities derived after filtering are not so accurate. Obviously this effect is major for a high noise (90% compensation).

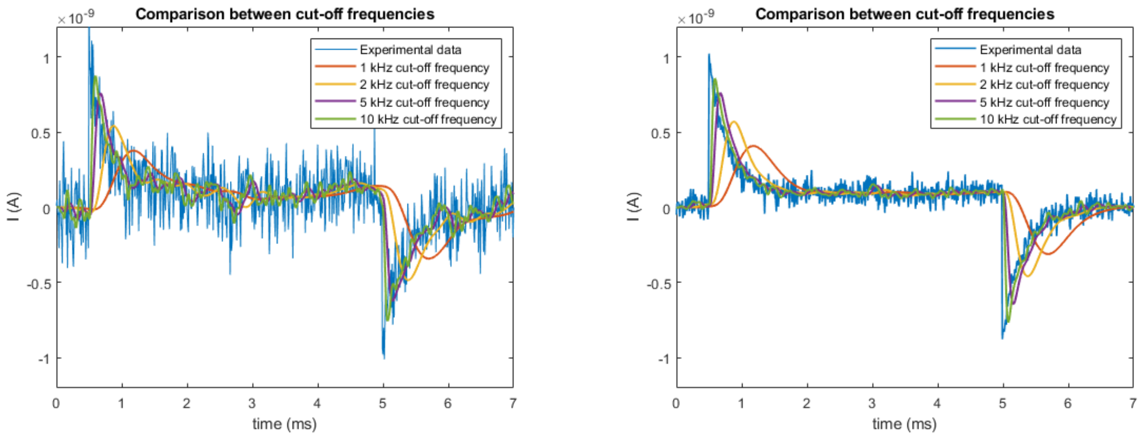

Figure 7: Left: comparison between various cut-off frequencies at 90% compensation, **Right:** comparison between various cut-off frequencies at 70% compensation

6 Conclusions

In general, simulations are very important to predict the behaviour of a system and to understand if the model built is coherent with the experimental data. This simulation correctly reproduces the signal measured in lab, however some differences could exist concerning the noise: the experimental conditions considerably affect the signal and the noise could be enhanced with respect to the simulation. Future studies can take into account other noisy effects which affect patch-clamp experiments, in order to create a simulation even closer to experimental data. Moreover it is clear that the errors are underestimated: a more detailed analysis of the errors would improve the simulation.

7 References

John F. MacDonald, Lu-Yang Wang, and Michael W. Salter, *Phosphorylation of Glutamate Receptors Using Electrophysiology Techniques*, Methods in Neurosciences, Volume 19, 1994

Flavio Fröhlich, *Network Neuroscience*, Chapter 1, Elsevier, 2016

Laura Conforti, *Cell Physiology Source Book, Essentials of Membrane Biophysics*, Chapter 20, Elsevier, 2011

A. L. Hodgkin and A. F. Huxley, *A quantitative description of membrane current and its application to conduction and excitation in nerve*, The Journal of Physiology, 1952

D. Ogden, D. Stanfield, *Patch clamp techniques for single channel and whole-cell recording*, Microelectrode Techniques (2nd edition). The Company of Biologists Ltd., 1999

Boris Barbour, *Electronics for electrophysiologists*, 2014

Dirk L. Ypey, Louis J. DeFelice, *Electrical Properties of Cells*, Springer, 2013

Manfred Lindau and Erwin Neher, *Patch-clamp techniques for time-resolved capacitance measurements in single cells*,European Journal of Physiology, 1988

Noise in electrophysiological measuraments, Axon guide

H. Sontheimer, *Whole-Cell Patch-Clamp Recordings*, Neuromethods, Vol 26 Patch-Clamp Applications and Protocols, Humana Press, 1995

8 Appendix

8.1 Demonstration of the formula for the access resistance

Here I demonstrate the equation (1). Starting from the analysis of the current

$$i_{Ra} = i_c + i_{Rm} = C_m \frac{dV_m}{dt} + \frac{V_m}{R_m}$$

with

$$V_m(t) = \frac{VR_m}{R_a + R_m}(1 - e^{-t/\tau})$$

from which

$$i_c = C_m \frac{dV_m}{dt} = \frac{VC_mR_m}{R_a + R_m}\frac{1}{\tau}e^{-t/\tau}$$

since

$$\tau = C_m \frac{R_mR_a}{R_a + R_m},$$

$$i_c = \frac{V}{R_a}e^{-t/\tau}$$

so the total current is

$$i_{Ra} = \left(\frac{V}{R_a} - \frac{V}{R_a + R_m}\right)e^{-t/\tau} + \frac{V}{R_a + R_m} = i_t e^{-t/\tau} + i_\infty$$

it is possible to notice that i_{Ra} is the sum of a transient current and a regime current. The charge associated to the transient current is

$$Q = \int_0^\infty \left(\frac{V}{R_a} - \frac{V}{R_a + R_m}\right)e^{-t/\tau}dt = \tau\left(\frac{V}{R_a} - \frac{V}{R_a + R_m}\right)$$

concluding

$$i_{Ra} = \frac{\tau \cdot i_t}{\tau} + i_\infty = \frac{Q}{\tau} + i_\infty$$

and so

$$R_a = \frac{V}{i_{Ra}} = \frac{\tau \cdot V}{Q + \tau \cdot i_\infty}$$

8.2 Matlab code

```
%cell parameters
R_a = 10^7; %Ohm
R_m = 10^8; %Ohm
C_m = 3e-11; %F
tau = C_m/(1/R_m+1/R_a);

fs=1e5; %sampling frequency (Hz)
s90 = (4*pi^2*C_m^2*4*1.38*10^(-23)*300*R_a*fs^2)/(1+4*pi*fs^2*10^12*C_m^2); ...
    %whole cell s^2 90%
noise90 = sqrt(s90*fs/2); %noise with 90% compensation
s70 = ...
    (4*pi^2*C_m^2*4*1.38*10^(-23)*300*R_a*fs^2)/(1+4*pi*fs^2*9*10^12*C_m^2); ...
    %whole cell s^2 70%
noise70 = sqrt(s70*fs/2);  %noise with 70% compensation

I_th = zeros(701,1);
V_m = zeros(701,1);
V_tot = zeros(701,1);
dt = 10^(-5);
t = zeros(701,1);
I_ex = zeros(701,1);
I_ex70 = zeros(701,1);
for i=1:700
    if i>50 && i< 500
        V_tot(i)=10^(-2);
    else
        V_tot(i)=0;
    end
    dV_m = (-V_m(i)+V_tot(i)*R_m/(R_m+R_a))*1/tau;
    I_th(i) = V_m(i)/R_m+C_m*dV_m;
    V_m(i+1) = V_m(i)+dV_m*dt;
    t(i+1) = t(i) + dt;
    I_ex(i) = I_th(i) + noise90*randn; %experimental current, with gaussian ...
        noise (90)
    I_ex70(i) = I_th(i) + noise70*randn; %experimental current, with gaussian ...
        noise (70)
end
```

```
%second part
% Experimental parameters (90% compensation)
R_a_ex = 0.01/max(I_ex);
I_inf = mean(I_ex(350:450));
R_m_ex = 0.01/I_inf − R_a_ex;

%fit to find Cm
I_ex_nuovo = I_ex−I_th(450);
f = fit(t(52:100),I_ex_nuovo(52:100),'exp1');
C = coeffvalues(f); %units: 10 microns
C_m_ex = 1/(−C(1,2))*(1/R_a_ex+1/R_m_ex);

%errors (90% compensation)
I_inf_er = std(I_ex(350:450))/sqrt(50);
R_a_ex_er = 0.01/max(I_ex)^2*noise90;
R_m_ex_er = sqrt((0.01*I_inf_er/I_inf^2)^2+R_a_ex_er^2);
interval = confint(f);
er_tau = (interval(2,2)−C(1,2))/2;
C_m_ex_er = sqrt((1/(−C(1,2)^2)*(1/R_a_ex+1/R_m_ex)*er_tau)^2+ ...
    (1/(−C(1,2))*(1/R_a_ex^2)*R_a_ex_er)^2+ ...
    (1/(−C(1,2))*(1/R_m_ex^2)*R_m_ex_er)^2);

% Experimental parameters (70% compensation)
R_a_ex70 = 0.01/max(I_ex70);
I_inf70 = mean(I_ex70(350:450));
R_m_ex70 = 0.01/I_inf70 − R_a_ex70;

%fit to find Cm
I_ex70_nuovo = I_ex70−I_th(450);
f70 = fit(t(52:120),I_ex70_nuovo(52:120),'exp1');
C70 = coeffvalues(f70);
C_m_ex70 = 1/(−C70(1,2))*(1/R_a_ex70+1/R_m_ex70);

%errors (70% compensation)
I_inf_er70 = std(I_ex70(350:450))/sqrt(50);
R_a_ex_er70 = 0.01/max(I_ex70)^2*noise70;
R_m_ex_er70 = sqrt((0.01*I_inf_er70/I_inf70^2)^2+R_a_ex_er70^2);
interval70 = confint(f70);
er_tau70 = (interval70(2,2)−C70(1,2))/2;
C_m_ex_er70 = sqrt((1/(−C70(1,2)^2)*(1/R_a_ex70+1/R_m_ex70)*er_tau70)^2+ ...
    (1/(−C70(1,2))*(1/R_a_ex70^2)*R_a_ex_er70)^2+ ...
    (1/(−C70(1,2))*(1/R_m_ex70^2)*R_m_ex_er70)^2);

%third part
I_thr = zeros(701,1);
I_exr = zeros(701,1);
Vr = −0.08*ones(701,1); %resting potential
V_mr = zeros(701,1); %membrane potential (Vm)
V_mr(1) = −0.08*R_a/(R_m+R_a); %initial condition
for i=1:700
    Vc = (R_m*V_tot(i)+R_a*(Vr(i)))/(R_a+R_m);
    dV_mr = −1/tau * (V_mr(i)−Vc);
    I_thr(i) = (V_mr(i)−Vr(i))/R_m+C_m*dV_mr;
    V_mr(i+1) = V_mr(i)+dV_mr*dt;
    I_exr(i) = I_thr(i) + noise70*randn; %experimental current, with gaussian ...
        noise (90)
end
```

```
% Experimental parameters (resting potential)
I_infr = mean(I_exr(350:450));
const = 0.09/0.01*((max(I_exr)/I_infr)-1);
R_a_exr = 0.09/(I_infr*(1+const));
R_m_exr = const*R_a_exr;

%fit to find Cm
I_exr_nuovo = I_exr-I_thr(450);
fr = fit(t(52:100),I_exr_nuovo(52:100),'exp1');
Cr = coeffvalues(fr);
C_m_exr = 1/(-Cr(1,2))*(1/R_a_exr+1/R_m_exr);

%errors (resting potential)
I_inf_err = std(I_exr(350:450))/sqrt(50);
err_const = 0.09/0.01*sqrt((max(I_exr)*I_inf_err/I_infr^2)^2+(noise70/I_infr)^2);
R_a_ex_err = ...
    0.09*sqrt((I_inf_err/(I_infr^2*(const+1)))^2+(err_const/(I_infr*(const+1)^2))^2);
R_m_ex_err = sqrt((R_a_exr*err_const)^2+(const*R_a_ex_er)^2);
intervalr = confint(fr);
er_taur = (intervalr(2,2)-Cr(1,2))/2;
C_m_ex_err = sqrt((1/(-Cr(1,2)^2)*(1/R_a_exr+1/R_m_exr)*er_taur)^2 ...
    +(1/(-Cr(1,2))*(1/R_a_exr^2)*R_a_ex_err)^2+ ...
     (1/(-Cr(1,2))*(1/R_m_exr^2)*R_m_ex_err)^2);

% fourth part:  4-pole low-pass Bessel filter
%90% compensation
y = zeros(701,10);
order=4; %4-pole
figure(1)
plot(t*10^3,I_ex)
hold on
for i=[1 2 5 10] %different cutoff frequency
s=i*1e3; %Hz
[b, a] = besself(order,s*2*pi);
[bz, az] = impinvar(b,a,fs);
y(:,i)=filter(bz,az,I_ex);
hold on
plot(t*10^3,y(:,i),'LineWidth',1.5) %plot of the filtered signal
end
ylim([-1.2e-9 1.2e-9]);
title('Comparison between cut-off frequencies');
legend('Experimental data', '1 kHz cut-off frequency', '2 kHz cut-off ...
    frequency', '5 kHz cut-off frequency', '10 kHz cut-off frequency');
xlabel('time (ms)');
ylabel('I (A)');
hold off

%70% compensation
y70 = zeros(701,10);
figure(2)
plot(t*10^3,I_ex70, 'LineWidth',1.2)
hold on
for i=[1 2 5 10] %different cutoff frequency
s=i*1e3; %Hz
[b, a] = besself(order,s*2*pi);
[bz, az] = impinvar(b,a,fs);
y70(:,i)=filter(bz,az,I_ex70); %plot of the filtered signal
hold on
plot(t*10^3,y70(:,i),'LineWidth',1.5)
```

```
end
ylim([-1.2e-9 1.2e-9]);
title('Comparison between cut-off frequencies');
legend('Experimental data', '1 kHz cut-off frequency', '2 kHz cut-off ...
    frequency', '5 kHz cut-off frequency', '10 kHz cut-off frequency');
xlabel('time (ms)');
ylabel('I (A)');
hold off

% see functions in the other file
% here are simulated fr 90% compensation, if you change with "noise70" and
% "y70" you obtain for 90% compensation
[R_a_10k, R_m_10k, C_m_10k, R_a_10k_er, R_m_10k_er, C_m_10k_er] = ...
    fit10k(t,y,noise90,I_th(450));
[R_a_5k, R_m_5k, C_m_5k, R_a_5k_er, R_m_5k_er, C_m_5k_er] = ...
    fit5k(t,y,noise90,I_th(450));
[R_a_2k, R_m_2k, C_m_2k, R_a_2k_er, R_m_2k_er, C_m_2k_er] = ...
    fit2k(t,y,noise90,I_th(450));
[R_a_1k, R_m_1k, C_m_1k, R_a_1k_er, R_m_1k_er, C_m_1k_er] = ...
    fit1k(t,y,noise90,I_th(450));

% other plots
figure(3)
plot(t*10^(3),I_exr,'LineWidth',1.1)
title('Resting potential');
xlabel('time (ms)');
ylabel('I (A)');
hold off

figure(4)
plot(t*10^(3),I_ex*10^(12),'LineWidth',1.1)
hold on
plot(t*10^(3),I_ex70*10^(12),'LineWidth',1.4)
hold on
plot(t*10^(3),I_th*10^(12),'LineWidth',2)
title('Comparison between experimental and theoretical current');
legend('90% compensation', '70% compensation', 'theoretical');
xlabel('time (ms)');
ylabel('I (pA)');
hold off

figure(5)
subplot(3,1,[1 2]);
plot(t*10^(3),I_ex*10^(12),'LineWidth',1.1)
title('Simulated current, 90% compensation');
xlabel('time (ms)');
ylabel('I (pA)');
subplot(3,1,3);
plot(t*10^(3),V_tot*10^3)
xlabel('time (ms)');
ylabel('V (mV)');
ylim([-0.5; 12]);
hold off

figure(6)
subplot(3,1,[1 2]);
plot(t*10^(3),I_ex70*10^(12),'Color','#D95319','LineWidth',1.1)
xlabel('time (ms)');
ylabel('I (pA)');
title('Simulated current, 70% compensation');
```

```
subplot(3,1,3);
plot(t*10^(3),V_tot*10^3)
xlabel('time (ms)');
ylabel('V (mV)');
ylim([-0.5; 12]);
hold off
```

And here are reported the functions used in the program. It is not possible to incorporate all the four functions into one since the points needed to compute τ are different because the delay introduced by the filter depends on the cut-off frequency.

```
function [R_a_10k, R_m_10k, C_m_10k, R_a_10k_er, R_m_10k_er, C_m_10k_er] = ...
    fit10k(t,y,noise,I_th)

%fit to find R_a (10 k)
Q_10k = 0;
y_n = y - I_th;
f_10k = fit(t(70:170),y_n(70:170,10),'exp1');
C_10k = coeffvalues(f_10k);
I_inf_10k = mean(y(350:495,10));

for j=1:230
   if y(j,10) > I_inf_10k
    Q_10k = Q_10k + (y(j,10)-I_inf_10k)*10^(-5);
    end
end

R_a_10k = (-0.01*(1/C_10k(1,2)))/(Q_10k-I_inf_10k*(1/C_10k(1,2))); %Ohm
R_m_10k = 0.01/I_inf_10k-R_a_10k; %Ohm
C_m_10k = 1/(-C_10k(1,2))*(1/R_a_10k+1/R_m_10k); %F

%errors
I_inf_10k_er = std(y(300:490,10));
y_err = (noise+I_inf_10k_er)*10^(-5);
interval = confint(f_10k);
er_tau = (interval(2,2)-C_10k(1,2))/2;
R_a_10k_er = ...
    sqrt(((-0.01*er_tau*(1/C_10k(1,2)^2))/(Q_10k-I_inf_10k*(1/C_10k(1,2))))^2+...
    ((-0.01*y_err*(1/C_10k(1,2)))/(Q_10k-I_inf_10k*(1/C_10k(1,2)))^2)^2+...
    ((-0.01*I_inf_10k_er*(1/C_10k(1,2))*(1/C_10k(1,2)))/ ...
        (Q_10k-I_inf_10k*(1/C_10k(1,2)))^2)^2+...
    ((-0.01*er_tau*(1/C_10k(1,2))*I_inf_10k*(1/C_10k(1,2)^2))/ ...
        (Q_10k-I_inf_10k*(1/C_10k(1,2)))^2)^2);
R_m_10k_er = sqrt((0.01*I_inf_10k_er/I_inf_10k^2)^2+R_a_10k_er^2);
C_m_10k_er = sqrt((1/(-C_10k(1,2)^2)*(1/R_a_10k+1/R_m_10k)*er_tau)^2+ ...
    (1/(-C_10k(1,2))*(1/R_a_10k^2)*R_a_10k_er)^2+ ...
    (1/(-C_10k(1,2))*(1/R_m_10k^2)*R_m_10k_er)^2);

%plot
figure(8)
plot(t,y(:,10))
title('10 kHz cut-off');
xlabel('time (ms)');
ylabel('I (A)');
hold off

end
```

```
function [R_a_5k, R_m_5k, C_m_5k, R_a_5k_er, R_m_5k_er, C_m_5k_er] = ...
    fit5k(t,y,noise,I_th)

%fit to find R_a (5 k)
Q_5k = 0;
y_n = y - I_th;
f_5k = fit(t(75:170),y_n(75:170,5),'exp1');
C_5k = coeffvalues(f_5k);
I_inf_5k = mean(y(300:490,5));

for j=1:230
   if y(j,5) > I_inf_5k
    Q_5k = Q_5k + (y(j,5)-I_inf_5k)*10^(-5);
   end
end

R_a_5k = (-0.01*(1/C_5k(1,2)))/(Q_5k-I_inf_5k*(1/C_5k(1,2))); %Ohm
R_m_5k = 0.01/I_inf_5k-R_a_5k; %Ohm
C_m_5k = 1/(-C_5k(1,2))*(1/R_a_5k+1/R_m_5k); %F

%errors
I_inf_5k_er = std(y(300:490,5));
y_err = (noise+I_inf_5k_er)*10^(-5);
interval = confint(f_5k);
er_tau = (interval(2,2)-C_5k(1,2))/2;
R_a_5k_er = ...
    sqrt(((-0.01*er_tau*(1/C_5k(1,2)^2))/(Q_5k-I_inf_5k*(1/C_5k(1,2))))^2+...
     ((-0.01*y_err*(1/C_5k(1,2)))/(Q_5k-I_inf_5k*(1/C_5k(1,2)))^2)^2+...
     ((-0.01*I_inf_5k_er*(1/C_5k(1,2))*(1/C_5k(1,2)))/ ...
         (Q_5k-I_inf_5k*(1/C_5k(1,2)))^2)^2+...
     ((-0.01*er_tau*(1/C_5k(1,2))*I_inf_5k*(1/C_5k(1,2)^2))/ ...
         (Q_5k-I_inf_5k*(1/C_5k(1,2)))^2)^2);
R_m_5k_er = sqrt((0.01*I_inf_5k_er/I_inf_5k^2)^2+R_a_5k_er^2);
C_m_5k_er = sqrt((1/(-C_5k(1,2)^2)*(1/R_a_5k+1/R_m_5k)*er_tau)^2+ ...
     (1/(-C_5k(1,2))*(1/R_a_5k^2)*R_a_5k_er)^2+ ...
     (1/(-C_5k(1,2))*(1/R_m_5k^2)*R_m_5k_er)^2);

%plot
figure(9)
plot(t,y(:,5))
title('5 kHz cut-off');
xlabel('time (ms)');
ylabel('I (A)');
hold off

end
```

```
function [R_a_2k, R_m_2k, C_m_2k, R_a_2k_er, R_m_2k_er, C_m_2k_er] = ...
    fit2k(t,y,noise,I_th)

%fit to find R_a (2 k)
Q_2k = 0;
y_n = y - I_th;
f_2k = fit(t(100:195),y_n(100:195,2),'exp1');
C_2k = coeffvalues(f_2k);
I_inf_2k = mean(y(300:490,2));

for j=1:230
```

```
    if y(j,2) > I_inf_2k
     Q_2k = Q_2k + (y(j,2)-I_inf_2k)*10^(-5);
    end
end

R_a_2k = (-0.01*(1/C_2k(1,2)))/(Q_2k-I_inf_2k*(1/C_2k(1,2))); %Ohm
R_m_2k = 0.01/I_inf_2k-R_a_2k; %Ohm
C_m_2k = 1/(-C_2k(1,2))*(1/R_a_2k+1/R_m_2k); %F


%errors
I_inf_2k_er = std(y(300:490,2));
y_err = (noise+I_inf_2k_er)*10^(-5);
interval = confint(f_2k);
er_tau = (interval(2,2)-C_2k(1,2))/2;
R_a_2k_er = ...
    sqrt(((-0.01*er_tau*(1/C_2k(1,2)^2))/(Q_2k-I_inf_2k*(1/C_2k(1,2))))^2+...
     ((-0.01*y_err*(1/C_2k(1,2)))/(Q_2k-I_inf_2k*(1/C_2k(1,2)))^2)^2+...
     ((-0.01*I_inf_2k_er*(1/C_2k(1,2))*(1/C_2k(1,2)))/ ...
         (Q_2k-I_inf_2k*(1/C_2k(1,2)))^2)^2+...
     ((-0.01*er_tau*(1/C_2k(1,2))*I_inf_2k*(1/C_2k(1,2)^2))/ ...
         (Q_2k-I_inf_2k*(1/C_2k(1,2)))^2)^2);
R_m_2k_er = sqrt((0.01*I_inf_2k_er/I_inf_2k^2)^2+R_a_2k_er^2);
C_m_2k_er = sqrt((1/(-C_2k(1,2)^2)*(1/R_a_2k+1/R_m_2k)*er_tau)^2+ ...
    (1/(-C_2k(1,2))*(1/R_a_2k^2)*R_a_2k_er)^2+ ...
    (1/(-C_2k(1,2))*(1/R_m_2k^2)*R_m_2k_er)^2);

%plot
figure(10)
plot(t,y(:,2))
title('2 kHz cut-off');
xlabel('time (ms)');
ylabel('I (A)');
hold off

end
```

```
function [R_a_1k, R_m_1k, C_m_1k, R_a_1k_er, R_m_1k_er, C_m_1k_er] = ...
    fit1k(t,y,noise,I_th)

%fit to find R_a (1 k)
Q_1k = 0;
y_n = y - I_th;
f_1k = fit(t(140:230),y_n(140:230,1),'exp1');
C_1k = coeffvalues(f_1k);
I_inf_1k = mean(y(300:490,1));

for j=1:230
    if y(j,1) > I_inf_1k
     Q_1k = Q_1k + (y(j,1)-I_inf_1k)*10^(-5);
    end
end

R_a_1k = (-0.01*(1/C_1k(1,2)))/(Q_1k-I_inf_1k*(1/C_1k(1,2))); %Ohm
R_m_1k = 0.01/I_inf_1k-R_a_1k; %Ohm
C_m_1k = 1/(-C_1k(1,2))*(1/R_a_1k+1/R_m_1k); %F

%errors
```

```
I_inf_1k_er = std(y(300:490,1));
y_err = (noise+I_inf_1k_er)*10^(-5);
interval = confint(f_1k);
er_tau = (interval(2,2)-C_1k(1,2))/2;
R_a_1k_er = ...
    sqrt(((-0.01*er_tau*(1/C_1k(1,2)^2))/(Q_1k-I_inf_1k*(1/C_1k(1,2))))^2+...
    ((-0.01*y_err*(1/C_1k(1,2)))/(Q_1k-I_inf_1k*(1/C_1k(1,2)))^2)^2+...
    ((-0.01*I_inf_1k_er*(1/C_1k(1,2))*(1/C_1k(1,2)))/ ...
        (Q_1k-I_inf_1k*(1/C_1k(1,2)))^2)^2+...
    ((-0.01*er_tau*(1/C_1k(1,2))*I_inf_1k*(1/C_1k(1,2)^2))/ ...
        (Q_1k-I_inf_1k*(1/C_1k(1,2)))^2)^2);
R_m_1k_er = sqrt((0.01*I_inf_1k_er/I_inf_1k^2)^2+R_a_1k_er^2);
C_m_1k_er = sqrt((1/(-C_1k(1,2)^2)*(1/R_a_1k+1/R_m_1k)*er_tau)^2+ ...
    (1/(-C_1k(1,2))*(1/R_a_1k^2)*R_a_1k_er)^2+ ...
    (1/(-C_1k(1,2))*(1/R_m_1k^2)*R_m_1k_er)^2);

%plot
figure(11)
plot(t,y(:,1))
title('1 kHz cut-off');
xlabel('time (ms)');
ylabel('I (A)');
hold off

end
```