Enabling real-time interaction with an electrophysiological cancer cell model

a BioTechMed-Graz Lab Rotation Report

of Peter Julius Waldert

supervised by **Prof. Christian Baumgartner** at the
Institute of Health Care Engineering with European Testing Center for Medical Devices,
Graz University of Technology, Austria,

21st October 2024.

1 Summary

Lung cancer is one of the most widespread pathologies worldwide and its mechanisms, specifically at the level of individual cells, are not well understood. We improve on the A549 electrophysiological cancer cell model introduced in [1, 2], combining numerical methods with an efficient implementation to reduce simulation time to a level where it is feasible for live interaction. More specifically, we were able to accelerate the simulation with adaptive timestepping and a highly efficient implementation in the Rust programming language, while we also managed to approach the corresponding inverse problem using a quadratic program, solving it within milliseconds. We introduce a visualisation approach of the entire model in the form of a live simulation dashboard available at https://in-silico.hce.tugraz.at/ running directly in the browser. The entire source code is freely available on GitHub and reusable through three different channels: the simulation interface (powered by compilation to WebAssembly), the Rust linkable library implementation and a Python package (simply run: `pip install in-silico-cancer-cell`). Our aim behind a distribution in this way is to make the topic and simulation as accessible as possible.

2 Introduction

The mechanisms behind lung adenocarcinoma, specifically those of individual A549 cells. Computational techniques can help with a better understanding of the behaviour of these cancer cells. We work with the A549 model introduced in [1, 2], reimplementing the model in the Rust programming language and performing a number of numerical optimizations such as adaptive timestepping. Physical measurements are obtained using a *Patch-Clamp System*, where one records the current through the membrane given a voltage protocol. In order to find appropriate parameters for the model, an optimization procedure is performed. Multiple optimization approaches for the solution of the corresponding inverse problem (fitting model parameters to measurement data) are put in comparison.

3 Methods

A single cell's membrane consists of multiple ion channels, categorized into $M \in \mathbb{N}$ different types. Each ion channel is represented in one of $N_{s,k} \in \mathbb{N}$ states, which, in physical terms, is related to a positional configuration of a protein within the ion channel. Only some states can be observed directly, and its development only depends on the one previous state. Hence, we are working with a Hidden Markov Model (HMM). For many ion channel categories, their transition probabilities are voltage or ion-concentration dependent.

The whole cell current $I : T \to \mathbb{R}$ over time $t \in T \subset \mathbb{R}^+$ is then obtained as the sum of all individual channel contributions $I_k, k \in \{1, ..., M\}$ over $M \in \mathbb{N}$ channel types

$$I(t) := \sum_{k=1}^{M} N_k I_k(t) = \sum_{k=1}^{M} N_k g_k p_{o,k} \left(V(t) - E_k\right) , \quad (1)$$

where N_k is the number of channels of type $k \in \{1, ..., M\}$, g_k is the respective ion channel's conductivity, $p_{o,k} \in [0, 1]$ is the probability of observing the channel in a state where an ion current can flow ("open states"), $V : T \to \mathbb{R}$ is the voltage across the membrane and $E_k \in \mathbb{R}$ the reversal potential.

Within the simulation, we sample the state and current at discrete time points $T_{\text{meas}} \subset T$, for example

$$T_{\text{meas}} := \left\{ t_n := \sum_{i=0}^{n} (\Delta t)_i \;\middle|\; n \in \mathbb{N}_0 \;\middle|\; n < N_t \right\}$$

for N_t measurements with step size $(\Delta t)_n$, which may be chosen equally large for all $n \in \{0, ..., N_t - 1\}$. We adapt this time interval $(\Delta t)_n \in \mathbb{R}^+$ per simulation step based on a state change heuristic, cf. Section 3.1.

At each time step,

$$\boldsymbol{s}_{k,n+1} = H_k \left(V(t_n), \boldsymbol{C}(t_n), t_n\right) \boldsymbol{s}_{k,n} \quad (2)$$

where $\boldsymbol{s}_{k,n} \in [0, 1]^{N_{s,k}}$ is the state vector of ion channel type k at the n-th time step, $H_k \left(V, \boldsymbol{C}, t_n\right) \in [0, 1]^{N_{s,k} \times N_{s,k}}$ the transition matrix for type k with $\sum_{j=1}^{N_{s,k}} \{H_k\}_{i,j} = 1 \;\forall\, i$, $V(t_n)$ the voltage across the membrane at time t_n and $\boldsymbol{C}(t_n) \in \mathbb{R}^4$ the concentrations of Kalium, Calcium, Sodium and Chlorine at time t_n. We initialize the simulation at $t_0 = 0$ with $\boldsymbol{s}_{k,0} = (1, 0, ..., 0)^T$ for all k.

Given the state $\boldsymbol{s}_k$, current measurements are then simply

$$\boldsymbol{I} := (I(t_0), I(t_1), ..., I(t_{N_t-1}))^T \in \mathbb{R}^{N_t} ,$$

with I as stated above in Equation (1) and

$$p_{o,k} = \sum_{j \in \mathcal{S}_{o,k}} \{\boldsymbol{s}_{n,k}\}_j ,$$

where $\mathcal{S}_{o,k}$ is the set of all states contributing to the ion channel current, the "open states".

3.1 Adaptive Timestepping

In order to accelerate the simulation in areas where there is little change to the dynamics, we choose an adaptive step size based on

$$(\Delta t)_{n+1} = (\Delta t)_n \left(\frac{\Delta^{\text{tol}}}{\sum_{k=1}^{M} N_k \left\| \boldsymbol{s}_{k,n+1} - \boldsymbol{s}_{k,n} \right\|_2} \right)^{1/2} , \tag{3}$$

for all n, where $\Delta^{\text{tol}} \in \mathbb{R}^+$ is a measure for the allowed state change in between steps. When the state changes too quickly in between time steps, the above heuristic will decrease $(\Delta t)_{n+1}$ and vice-versa. In principle, it would be feasible to apply the adaptive timestepping to each ion channel type individually, however this would make current sampling and data synchronization between channels hard to realize, considering the ion concentration dependence of H_k. Within this paper, we set $\Delta^{\text{tol}} = 2 \cdot 10^{-7}$.

3.2 Inverse Problem

When regarding the cell model as a whole, the number of ion channels N_k per type k may be put into a configuration vector $\boldsymbol{N} := (N_1, ..., N_M)^T \in \mathbb{N}_0^M$ and the total simulated current I sampled at measurement points T_{meas} can be expressed as a matrix-vector product

$$\boldsymbol{I} = \sum_{k=1}^{M} N_k \boldsymbol{I_k} = R\boldsymbol{N} , \tag{4}$$

where $R \in \mathbb{R}^{N_t \times M}$ is the matrix of all current measurements per channel type.

Given the individual ion channel type models' parameters, which we know from literature (cf. Table 1), the question that remains is how many channels there are of each type to fit the measurements. This problem can be solved using a number of optimization approaches.

However, the formulation in Equation (4) also gives rise to a least-squares formulation, by projecting the measured current into the space of all individual channel currents. More specifically, we want to find

$$\boldsymbol{N}_{\text{opt}} = \underset{\boldsymbol{N} \in \mathbb{N}_0^M}{\arg\min} \frac{1}{2} \left\| R\boldsymbol{N} - \boldsymbol{I}_{\text{meas}} \right\|_2^2 , \tag{5}$$

with $\boldsymbol{I}_{\text{meas}} \in \mathbb{R}^{N_t}$ the experimentally measured current. The most important constraint here is that of integer non-negativity, $\boldsymbol{N}_{\text{opt}} \in \mathbb{N}_0^M$, which makes this problem hard to solve directly.

Table 1: Ion Channel Types: Most fitting configuration and corresponding references.

Channel Type	N_k [1]	**Our** N_k	**Reference**
Kv13	22	13	[4]
Kv31	78	247	[5]
Kv34	5	10	[6]
Kv71	1350	1176	[7]
KCa11	40	38	[8]
KCa31	77	7	[9]
Task1	19	24	[10]
CRAC1	200	188	[11]
TRPC6	17	15	[12]
TRPV3	12	10	[13]
CLC2	13	234	[14]

The unconstrained least-squares problem could be solved very efficiently using a QR-decomposition of the current basis $R = Q_b R_b$, its solution would be $\boldsymbol{N}_{opt} = \lfloor R_b^{-1} Q_b^* \boldsymbol{I}_{\text{meas}} \rceil \in \mathbb{Z}^M$.

3.3 Formulation as a Quadratic Program

Relaxing the integer condition on the solution, and letting $\boldsymbol{d} := \boldsymbol{I}_{\text{meas}}$ for brevity, we can reformulate Equation (5),

$$\boldsymbol{N}_{\text{opt}} \approx \underset{\boldsymbol{x} \in \mathbb{R}_+^M}{\arg\min}\, f(\boldsymbol{x}) = \underset{\boldsymbol{x} \in \mathbb{R}_+^M}{\arg\min}\, {}^1\!/_2 \left\| R\boldsymbol{x} - \boldsymbol{d} \right\|_2^2 ,$$

with cost function $f : \mathbb{R}^M \to \mathbb{R}^+$, which we manipulate to

$$\begin{aligned} f(\boldsymbol{x}) &= {}^1\!/_2 (R\boldsymbol{x} - \boldsymbol{d})^T (R\boldsymbol{x} - \boldsymbol{d}) \\ &= {}^1\!/_2 \left(\boldsymbol{x}^T R^T R \boldsymbol{x} - \boldsymbol{x}^T R^T \boldsymbol{d} - \boldsymbol{d}^T R \boldsymbol{x} + \boldsymbol{d}^T \boldsymbol{d} \right) \\ &= {}^1\!/_2 \left(\boldsymbol{x}^T P \boldsymbol{x} + \boldsymbol{x}^T \boldsymbol{q} + \boldsymbol{q}^T \boldsymbol{x} \right) + \mathcal{O}(1) \\ &= {}^1\!/_2 \boldsymbol{x}^T P \boldsymbol{x} + \boldsymbol{q}^T \boldsymbol{x} + \mathcal{O}(1) \end{aligned}$$

where we let $P := R^T R \in \mathbb{R}^{M \times M}$ and $\boldsymbol{q} := -R^T \boldsymbol{d} \in \mathbb{R}^M$ and leave out the constant $\boldsymbol{d}^T \boldsymbol{d}$ as $\mathcal{O}(1)$. We can express the nonnegativity constraint $\boldsymbol{x} \geq \boldsymbol{0}$ as an equality constraint using a slack variable $\boldsymbol{s} \in \mathbb{R}_+^M$,

$$-\boldsymbol{x} + \boldsymbol{s} = \boldsymbol{0} \quad \Leftrightarrow \quad A\boldsymbol{x} + \boldsymbol{s} = \boldsymbol{b} ,$$

where we set $A := -\mathbb{1} \in \mathbb{R}^{M \times M}$ and $\boldsymbol{b} := \boldsymbol{0} \in \mathbb{R}^M$. This leaves us with a constrained *quadratic program*,

$$\min_{\boldsymbol{x} \in \mathbb{R}^M} \frac{1}{2} \boldsymbol{x}^T P \boldsymbol{x} + \boldsymbol{q}^T \boldsymbol{x}, \tag{6}$$

$$s.t.\ A\boldsymbol{x} + \boldsymbol{s} = \boldsymbol{b} ,\ \boldsymbol{s} \in \mathbb{R}_+^M . \tag{7}$$

We solve the quadratic problem in this exact form using Clarabel [3]. Note that in Clarabel notation, the slack variable is to be taken as an element of the nonnegativity cone.

The integer solution can then be obtained from rounding,

$$\boldsymbol{N}_{\text{opt}} = \lfloor \boldsymbol{x} \rceil \in \mathbb{N}_0^M .$$

3.4 Implementation and Usage

The core simulation is implemented in the Rust programming language [15]. Each channel is represented as its own struct, implementing the `HasTransitionMatrix` trait. The number of states per channel type is constant by design, allowing the compiler to optimize the instruction hierarchy and memory alignment of the simulation. All individual ion channel types are then combined into the full `A549CancerCell` struct, whose most important function is `cell.simulate();`.

For ease of use, we provide a Python interface to the simulation, ready to install via the Python Package Index (PyPI) as precompiled wheels for the most common platforms.

```python
# pip install in-silico-cancer-cell
from in_silico_cancer_cell import *

# to evaluate on existing parameters
cell = A549CancerCell.new()
error = cell.evaluate(
  PatchClampProtocol.Ramp,
  CellPhase.G1
)

# to obtain the current basis matrix
meas = PatchClampData.pyload(
  PatchClampProtocol.Activation,
  CellPhase.G0
)
p = ChannelCountsProblem.new(meas)
p.precompute_single_channel_currents()
single_channels =
 ↪  p.get_current_basis()
```

3.5 Live Simulation

In addition to the programming interfaces, we also provide a live simulation web dashboard, available on https://in-silico.hce.tugraz.at. A screenshot is provided in Figure 6. After choosing the voltage protocol and cell cycle phase, the dashboard runs the full simulation live, within the browser. Computation time is well below a second for all protocol & phase measurement configurations. The dashboard shows the full current through the membrane at the top, compared to the measurement, while also breaking it down into individual channel type contributions. Each channel type is shown as the evolution of its state space and resulting current over time.

4 Results

The new simulation reproduces the measurements documented in [1] accurately (cf. Figures 2 and 3, the remaining five cell cycle phase & voltage protocol configurations were ommitted in this paper but can be verified via the online dashboard). The simulation itself is accessible via its code interface, available from GitHub[1] and the live simulation dashboard. Our method offers a set of significant numerical improvements in terms of the simulation runtime and optimization procedure, as presented in the previous chapter.

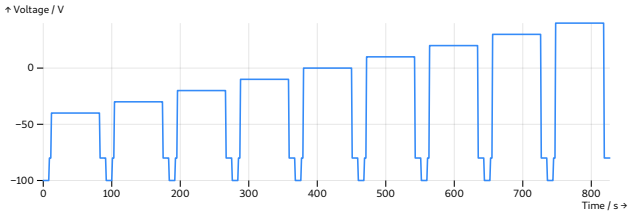

Figure 1: The *activation* voltage pulse protocol $V(t)$ used for the measurement and simulation. This is the voltage set across the A549 cell's membrane using a Patch-Clamp-System. The corresponding current is then recorded and depicted in Figure 2 along with its simulation result.

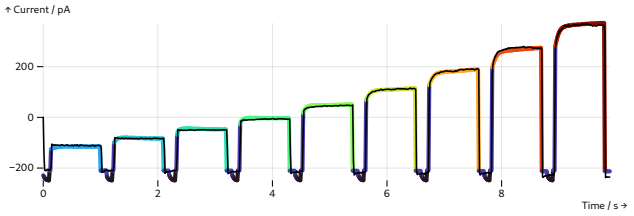

Figure 2: The simulated and measured current response $I(t)$ across the membrane of an A549 cancer cell, given the *activation* voltage protocol $V(t)$ in Figure 1, recorded in the G0 phase of the cell cycle.

4.1 Adaptive Timestepping

With the standard forward iteration approach, the simulation takes 412 million steps to simulate over all voltage protocols and cell cycle phases, while the adaptive timestepping only requires 9 million steps for the same configuration. While keeping the same level of accuracy when matching with the experimental data, our adaptive timestepping method is therefore 45 times faster than the standard

[1] https://github.com/MrP01/InSilicoCancerCell

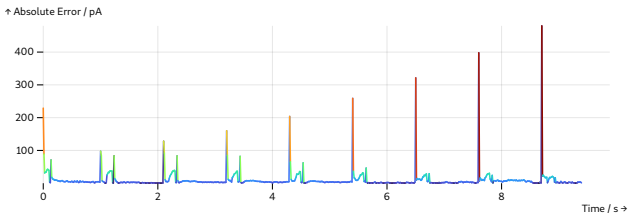

Figure 3: Pointwise error between the measured current $\boldsymbol{I}_{\text{meas}}$ and the simulated current $\boldsymbol{I}$, showing potential systematic problems within the computational model's representation of the ion channels as compared to the experimental results.

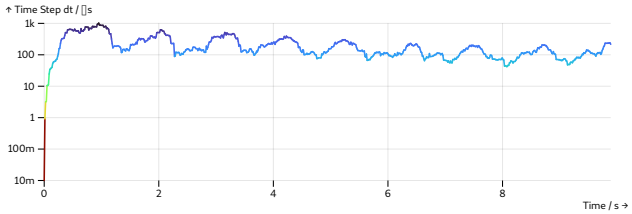

Figure 4: Timestep Δt throughout the duration of the simulation. $(\Delta t)_n$ is chosen in between simulation steps based on the heuristic given in Equation (3). Red areas (small dt) indicate a high resolution within the simulation, blue areas indicate lower resolution (and therefore higher simulation speed).

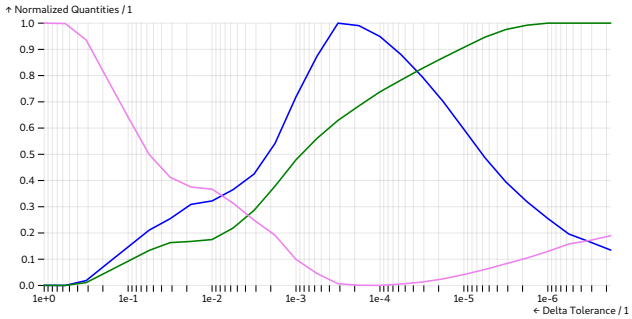

Figure 5: Relative change of the average timestep Δt (in blue), simulation runtime (in violet) and step acceptance rate (in green) when varying the delta tolerance Δ^{tol} on a log-scale. All three quantities were normalized from their individual extent to $[0, 1]$. The most effective Δ^{tol} is arguably on the order of 10^{-4}.

approach, on average. The time increment Δt chosen at each point in the simulation can be found in Figure 4.

In order to quantitatively establish the best possible state change tolerance Δ^{tol}, we recorded a number of simulation snapshots across different values for Δ^{tol}. The results of this numerical experiment are documented in Figure 5, which proves $\Delta^{\text{tol}} = 10^{-4}$ to be the best choice.

4.2 Inverse Problem

We compare different optimization approaches based on their runtime and root mean square error,

$$\Delta_{\text{RMS}} = \sqrt{1/N_t \sum_{j=1}^{N_t} (I_j - I_{\text{meas},j})^2} = \frac{\|\boldsymbol{I} - \boldsymbol{I}_{\text{meas}}\|_2}{\sqrt{N_t}} .$$

Specifically, the best performing formulation for the channel counts problem in this context is the Quadratic Program (QP), cf. Table 2. The program was solved using the Clarabel [3] optimization library (also written in Rust), which completes the procedure in just 18 milliseconds. Measurements and simulation were compared based on an FFT-accelerated Gauss-kernel convolution smoothed version of the 180,000 point measurements.

4.3 Live Simulation Dashboard

A screenshot of the simulation environment built upon the methods described in this paper is documented in Figure 6.

5 Outlook

Efforts to further enhance and complete the first electrophysiological cancer cell model simulation were fruitful, resulting in a new library implementation, an improved inverse problem solution technique and a live in-browser simulation dashboard.

Numerically, the stability of the simulation varies greatly with the time step state change tolerance Δ^{tol}, this could be improved using a higher-order integration scheme.

Regarding the simulation dashboard, there are still many adjustments that could improve and enable further usage perspectives. Specifically, it would be effective to add visual handles for adjusting the number of channels per type and weigh their individual contributions, for instance.

Another useful extension would be to enable the interface-based integration of further ion channel types as part of the simulation, without the need to specifically modify the pre-compiled Rust source code.

References

Langthaler, Sonja, Theresa Rienmüller, Susanne Scheruebel, Brigitte Pelzmann, Niroj Shrestha, Klaus Zorn-Pauly, Wolfgang Schreibmayer, Andrew Koff and Christian Baumgartner (June 2021). 'A549 in-silico 1.0: A first computational model to simulate cell cycle dependent ion current modulation in the human lung adenocarcinoma'. In: *PLoS Comput. Biol.* 17.6, e1009091. DOI: 10.1371/journal.pcbi.1009091.

Langthaler, Sonja, Christian Zumpf, Theresa Rienmüller, Niroj Shrestha, Julia Fuchs, Rui Zhou, Brigitte Pelzmann, Klaus Zorn-Pauly, Eleonore Fröhlich, Seth H. Weinberg and Christian Baumgartner (May 2024). 'The bioelectric mechanisms of local calcium dynamics in cancer cell proliferation: an extension of the A549 in silico cell model'. In: *Front. Mol. Biosci.* 11, p. 1394398. DOI: 10.3389/fmolb.2024.1394398.

Goulart, Paul J. and Yuwen Chen (2024). *Clarabel: An interior-point solver for conic programs with quadratic objectives*. arXiv: 2405.12762 [math.OC].

Hou, Panpan, Rong Zhang, Yongfeng Liu, Jing Feng, Wei Wang, Yingliang Wu and Jiuping Ding (Mar. 2014). 'Physiological Role of Kv1.3 Channel in T Lymphocyte Cell Investigated Quantitatively by Kinetic Modeling'. In: *PLoS One* 9.3, e89975. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0089975.

Wang, Wei, Jie Luo, Panpan Hou, Yimei Yang, Feng Xiao, Ming Yuchi, Anlian Qu, Luyang Wang and Jiuping Ding (Oct. 2013). 'Native Gating Behavior of Ion Channels in Neurons with Null-Deviation Modeling'. In: *PLoS*

Table 2: Comparison of Optimization Approaches, evaluated on the G0 cell cycle phase with the activation voltage protocol (cf. Figure 1). Runtime estimates were obtained on an Intel™i7-5600U CPU.

Algorithm	**Abbreviation**	**Runtime** / ms	**RMSE** / pA
Particle Swarm Optimization	PSO	22571	27.69
Gradient Descent + More Thuente	GD	18924	32.34
Limited-Memory BFGS + Hager Zhang	LBFGS	4845	32.20
Non-Negative Least Squares [16]	NNLS	318	28.00
Quadratic Program	QP	18	28.13

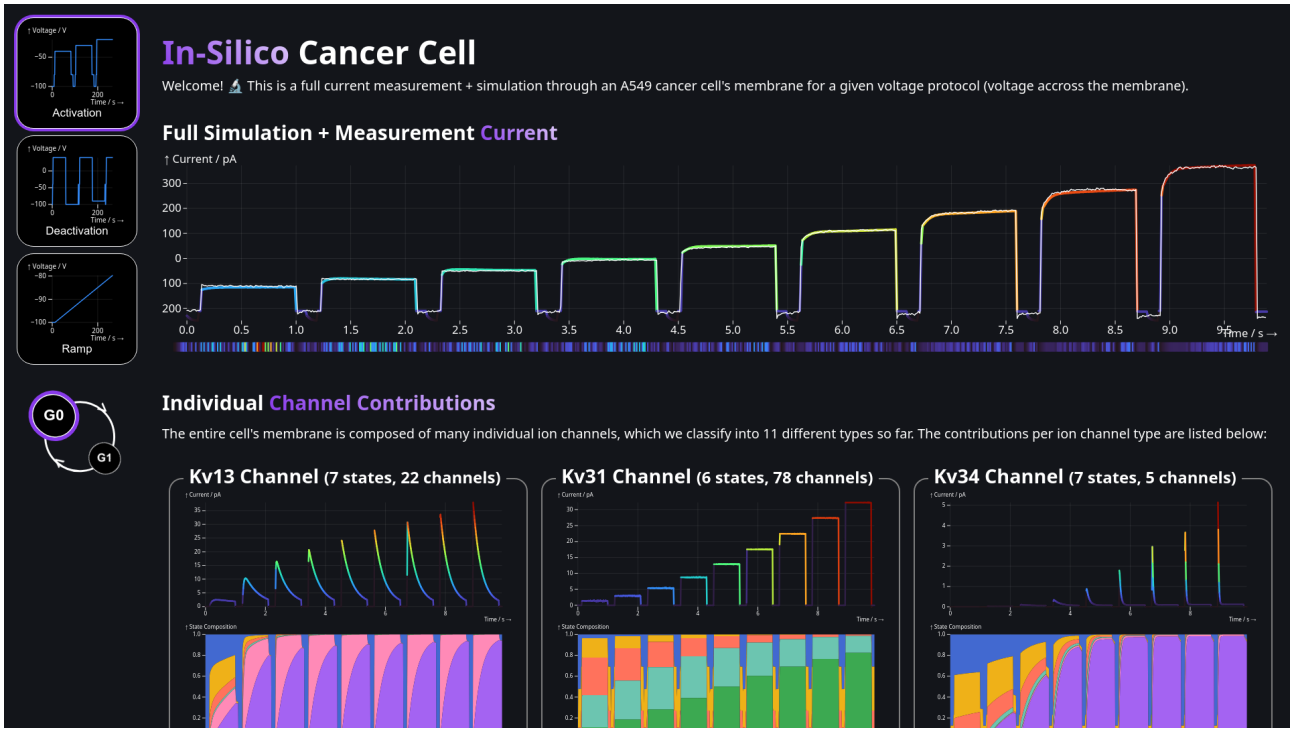

Figure 6: Screenshot of the live simulation dashboard, available here: https://in-silico.hce.tugraz.at/. This is the graphical user interface to the simulation written in Rust. The full simulation, completing within around 500ms, runs live in the browser using Rust's bindings to WebAssembly. One can choose the voltage protocol and cell cycle phase (G0/G1) in the menu on the left, while the full current and individual channel currents and internal states can be seen on the right. The remaining ion channel types have been cut off on this screenshot.

One 8.10, e77105. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0077105.

Schroter, Klaus-Hasso, Johann Peter Ruppersberg, Frank Wunder, Jens Rettig, Martin Stocker and Olaf Pongs (Jan. 1991). 'Cloning and functional expression of a TEA-sensitive A-type potassium channel from rat brain'. In: *FEBS Lett.* 278.2, pp. 211–216. ISSN: 0014-5793. DOI: 10.1016/0014-5793(91)80119-N.

Werry, Daniel, Jodene Eldstrom, Zhuren Wang and David Fedida (Mar. 2013). 'Single-channel basis for the slow activation of the repolarizing cardiac potassium current, IKs'. In: *Proc. Natl. Acad. Sci. U.S.A.* 110.11, E996–E1005. DOI: 10.1073/pnas.1214875110.

Wang, Wei, Feng Xiao, Xuhui Zeng, Jing Yao, Ming Yuchi and Jiuping Ding (Apr. 2012). 'Optimal Estimation of Ion-Channel Kinetics from Macroscopic Currents'. In: *PLoS One* 7.4, e35208. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0035208.

Grissmer, S., A. N. Nguyen and M. D. Cahalan (Oct. 1993). 'Calcium-activated potassium channels in resting and activated human T lymphocytes. Expression levels, calcium dependence, ion selectivity, and pharmacology.' In: *J. Gen. Physiol.* 102.4, pp. 601–630. ISSN: 0022-1295. DOI: 10.1085/jgp.102.4.601.

Papreck, Justin R., Elizabeth A. Martin, Ping Lazzarini, Dawon Kang and Donghee Kim (Nov. 2012). 'Modulation of K2P3.1 (TASK-1), K2P9.1 (TASK-3), and TASK-1/3 heteromer by reactive oxygen species'. In: *Pflügers. Arch. - Eur. J. Physiol.* 464.5, pp. 471–480. ISSN: 1432-2013. DOI: 10.1007/s00424-012-1159-y.

Kerschbaum, Hubert H. and Michael D. Cahalan (Feb. 1999). 'Single-Channel Recording of a Store-Operated Ca2+ Channel in Jurkat T Lymphocytes'. In: *Science* 283.5403, pp. 836–839. ISSN: 0036-8075. DOI: 10.1126/science.283.5403.836.

Vazquez, Guillermo, Barbara J. Wedel, Omar Aziz, Mohamed Trebak and James W. Putney (Dec. 2004). 'The mammalian TRPC cation channels'. In: *Biochimica et Biophysica Acta (BBA) - Molecular Cell Research* 1742.1, pp. 21–36. ISSN: 0167-4889. DOI: 10.1016/j.bbamcr.2004.08.015.

Xu, Haoxing, I. Scott Ramsey, Suhas A. Kotecha, Magdalene M. Moran, Jayhong A. Chong, Deborah Lawson, Pei Ge, Jeremiah Lilly, Inmaculada Silos-Santiago, Yu Xie, Peter S. DiStefano, Rory Curtis and David E. Clapham (July 2002). 'TRPV3 is a calcium-permeable temperature-sensitive cation channel'. In: *Nature* 418.6894, pp. 181–186. ISSN: 1476-4687. DOI: 10.1038/nature00882.

Weinreich, F. and T. J. Jentsch (Jan. 2001). 'Pores formed by single subunits in mixed dimers of different CLC chloride channels'. In: *J. Biol. Chem.* 276.4, pp. 2347–2353. ISSN: 0021-9258. DOI: 10.1074/jbc.M005733200. eprint: 11035003.

Matsakis, Nicholas D. and Felix S. Klock (Oct. 2014). 'The rust language'. In: *Ada. Lett.* 34.3, pp. 103–104. ISSN: 1094-3641. DOI: 10.1145/2692956.2663188.

Bro, Rasmus and Sijmen De Jong (Sept. 1997). 'A fast non-negativity-constrained least squares algorithm'. In: *J. Chemom.* 11.5, pp. 393–401. ISSN: 0886-9383. DOI: 10.1002/(SICI)1099-128X(199709/10)11:5<393::AID-CEM483>3.0.CO;2-L.

Signatures

This project was carried out as part of a BioTechMed-Graz *Lab Rotation*. The aim of the programme is to give interested graduates a chance to broaden their scientific interests beyond the topic of their master's thesis.

(Peter Waldert)	Place and Date
(Prof. Christian Baumgartner)	Place and Date