Peter Waldert*, Sonja Langthaler, Theresa Rienmüller, and Christian Baumgartner

Reducing runtime of an electrophysiological cancer cell model from X to Y and live simulation dashboard

https://doi.org/10.1515/

Abstract: Through a reimplementation in Rust and numerical optimization approaches we were able to reduce the runtime of the A549 electrophysiological cancer cell model [1] from X to Y.

Keywords: Please insert your keywords here, separated by commas.

1 Introduction

Lung cancer is one of the most widespread pathologies worldwide and its mechanisms, specifically those of individual A549 cells, are not well understood. Computational techniques can help with a better understanding of the behaviour of these cancer cells. We work with the A549 model introduced in [1], together with a calcium channel extension introduced in [2], reimplementing the model in the Rust programming language and performing a number of numerical optimizations such as adaptive timestepping. We also verify the model's performance in *Floating* mode, as compared to an individual simulation of the estimated number of channels. We introduce a visualisation approach of the entire model in the form of a live simulation dashboard[1]. The entire source code behind this simulation is freely available on GitHub[2], and reusable through three different channels: the simulation interface (powered by WebAssembly), the Rust linkable library implementation and a Python package. These three interfaces all originate from the same source code and our aim behind a distribution in this way is to make the simulation as accessible as possible. 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. The measurements are obtained using a *Patch-Clamp System*, where one records the current through the membrane given a voltage protocol.

2 Methods

A 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 2.1.

At each time step,

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

1 https://in-silico-cancer-cell.waldert.at/

2 https://github.com/MrP01/InSilicoCancerCell

***Corresponding author: Peter Waldert,** Institute of Health Care Engineering with European Testing Center for Medical Devices, Graz University of Technology, Graz, Austria, e-mail: peter.waldert@tugraz.at
Sonja Langthaler, Theresa Rienmüller, Christian Baumgartner, Institute of Health Care Engineering with European Testing Center for Medical Devices, Graz University of Technology, Graz, Austria

where $\boldsymbol{s}_{k,n} \in \mathbb{R}^{N_{s,k}}$ is the state vector of ion channel type k at the n-th time step, $H_k(V, \boldsymbol{C}, t_n) \in \mathbb{R}^{N_{s,k} \times N_{s,k}}$ the transition matrix for type k, $V(t_n)$ 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

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

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

2.1 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

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

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 $\Delta^{\text{tol}} = 2 \cdot 10^{-7}$.

2.2 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

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

where $A \in \mathbb{R}^{N_t \times M}$ 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

Tab. 1: Ion Channel Types

Channel Type	Channel Count N_k	Reference
Kv13	1	[3]
Kv31	1	[3]
Kv34	1	[3]
Kv71	1	[3]
KCa11	1	[3]
KCa31	1	[3]
Task1	1	[3]
CRAC1	1	[3]
TRPC6	1	[3]
TRPV3	1	[3]
CLC2	1	[3]

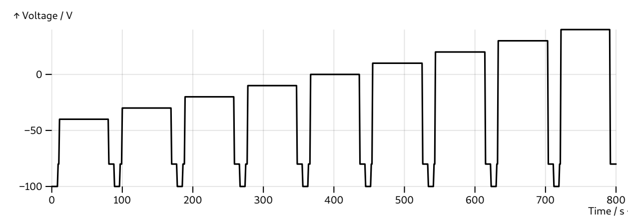

Fig. 1: Please insert your figure caption here.

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

$$\boldsymbol{N}_{\text{opt}} = \underset{\boldsymbol{N} \in \mathbb{N}_0^M}{\operatorname{argmin}} \|A\boldsymbol{N} - I_{\text{meas}}\|_2^2 , \tag{5}$$

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.

2.3 Implementation Architecture

The core simulation is implemented in the Rust programming language [4].

2.4 Live Simulation

tbd

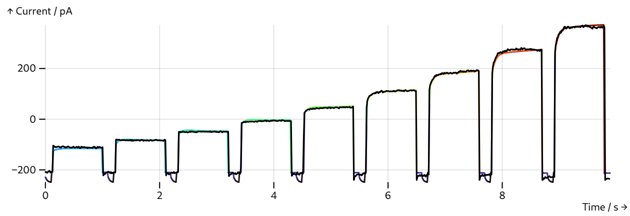

Fig. 2: Please insert your figure caption here.

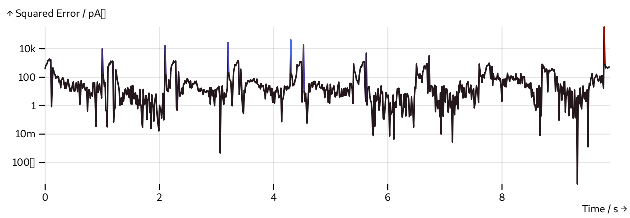

Fig. 3: Please insert your figure caption here.

3 Results

4 Outlook

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.

Author Statement

Research funding: This work was funded through a BioTechMed-Graz Lab Rotation Research Fellowship. Conflict of interest: Authors state no conflict of interest. Informed consent: Informed consent has been obtained from all individuals included in this study. Ethical approval: The research related to human use complies with all the relevant national regulations, institutional policies and was performed in accordance with the tenets of the Helsinki Declaration, and has been approved by the authors' institutional review board or equivalent committee.

Tab. 2: Comparison of Optimization Approaches

Algorithm	Accuracy
Particle Swarm Optimization	1
Gradient Descent	2
LBFGS	3
QR-based LSQ	4
NNLS	5

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. ISSN: 1553-7358. 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. ISSN: 2296-889X. DOI: 10.3389/fmolb.2024.1394398.

Pusch, Michael, Raffaella Magrassi, Bernd Wollnik, and Franco Conti (Aug. 1998). "Activation and Inactivation of Homomeric KvLQT1 Potassium Channels." In: *Biophys. J.* 75.2, pp. 785–792. ISSN: 0006-3495. DOI: 10.1016/S0006-3495(98)77568-X.

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.

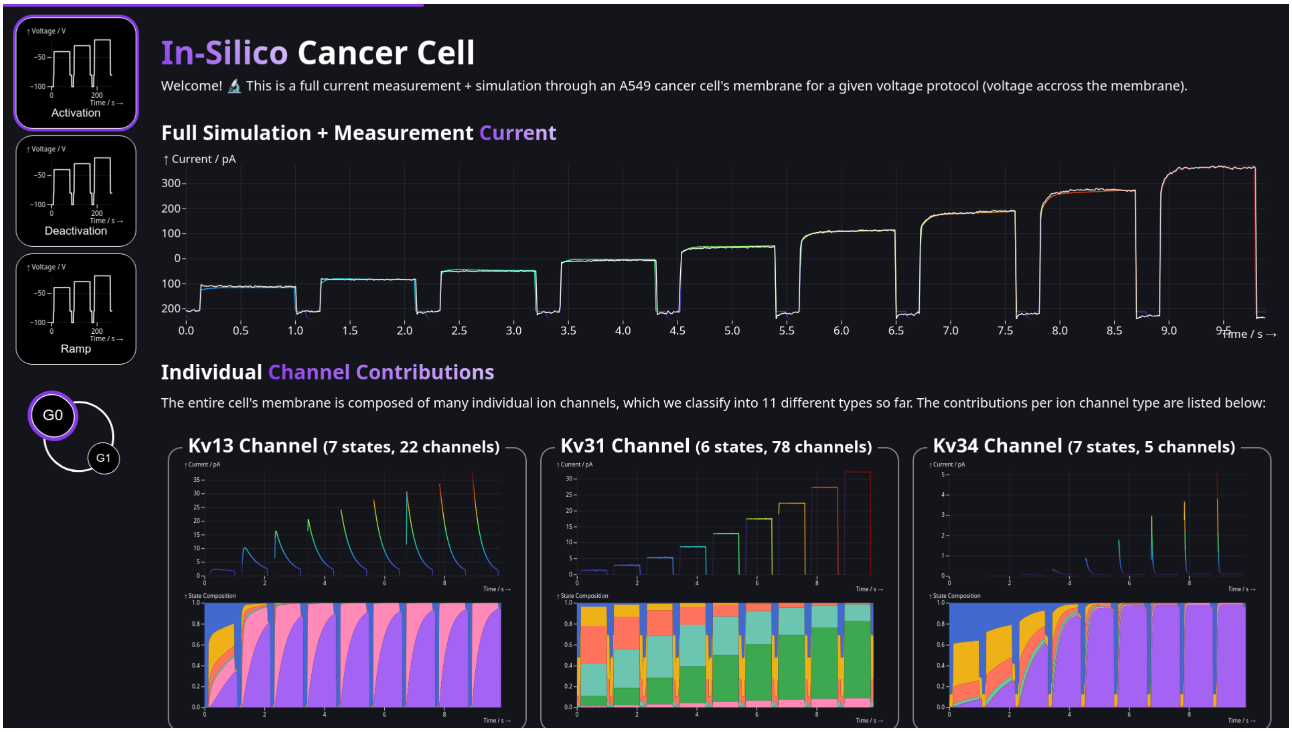

Fig. 4: Please insert your figure caption here.