Electrodiffusion Simulations of Neurons and Extracellular Space

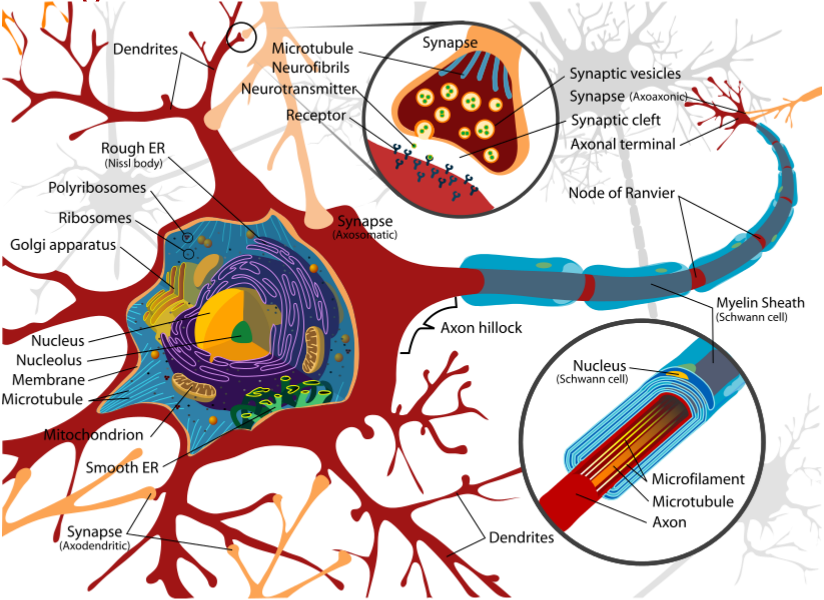
Models of the Axonal Membrane and Surrounding Fluids Based on the PNP Equations of Electrodiffusion

Jurgis Pods

Interdisziplinäres Zentrum für wissenschaftliches Rechnen, Universität Heidelberg

October 10, 2013

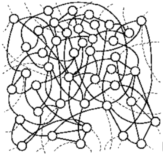
Biological Neuron Models: Levels of Abstraction



http://en.wikipedia.org/wiki/Myelin_sheath_gap

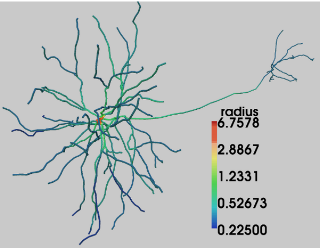
Biological Neuron Models: Levels of Abstraction

Point neurons (“0D”)



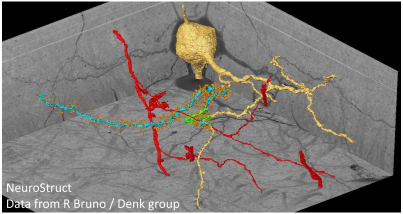
University of Minnesota, 2004

Compartment models (“1D”)



Dan Popovic, IWR

Panos Drouvelis, Stefan Lang, IWR

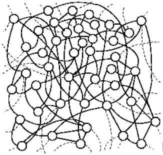


Detailed 2D and 3D electrodiffusion models

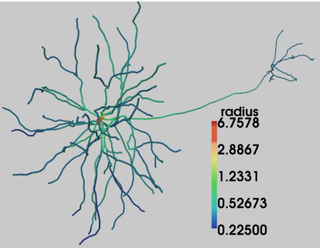
IWR

Biological Neuron Models: Levels of Abstraction

Point neurons ("0D")



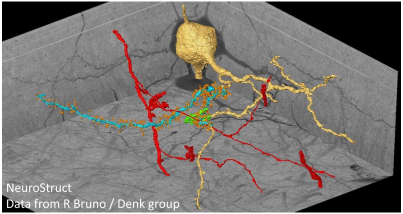
University of Minnesota, 2004



Dan Popovic, IWR

Compartment models ("1D")

Panos Drouvelis, Stefan Lang, IWR



Detailed 2D and 3D electrodiffusion models

IWR

The Nernst-Planck-Poisson System

describes ion drift through a static solvent.

Nernst-Planck: $$\frac{\partial n_i}{\partial t} - \nabla \cdot \mathbf{F}_i = 0$$ (conservation of charge)

with the ion flux $$\mathbf{F}_i = D_i \left(\nabla n_i + z_i n_i \nabla \phi \right)$$

Poisson: $$\nabla \cdot (\epsilon \nabla \phi) = -\frac{e^2 n^*}{\epsilon_0 k T} \sum_i z_i n_i$$

- For ion species i:
  - n_i: relative concentration (with respect to scaling concentration n^*)
  - z_i: valence ($\pm$ 1)
  - D_i: (position-dependent) diffusion coefficient
- ϕ: relative electric potential energy with respect to the thermal energy ($\phi = \frac{e}{kT} U$ with U in Volts)
- ϵ: (position-dependent) relative permittivity
- T: temperature of the solvent
- e, ϵ_0, k: natural constants

IWR

The Nernst-Planck-Poisson System

describes ion drift through a static solvent.

Nernst-Planck: $\frac{\partial n_i}{\partial t} - \nabla \cdot \mathbf{F}_i = 0$ (conservation of charge)

with the ion flux $\mathbf{F}_i = D_i \left(\nabla n_i + z_i n_i \nabla \phi \right)$ diffusion

Poisson: $\nabla \cdot (\epsilon \nabla \phi) = -\frac{e^2 n^*}{\epsilon_0 k T} \sum_i z_i n_i$

- For ion species i:
  - n_i: relative concentration (with respect to scaling concentration n^*)
  - z_i: valence ($\pm$ 1)
  - D_i: (position-dependent) diffusion coefficient
- ϕ: relative electric potential energy with respect to the thermal energy ($\phi = \frac{e}{kT} U$ with U in Volts)
- ϵ: (position-dependent) relative permittivity
- T: temperature of the solvent
- e, ϵ_0, k: natural constants

IWR

The Nernst-Planck-Poisson System

describes ion drift through a static solvent.

Nernst-Planck: $$\frac{\partial n_i}{\partial t} - \nabla \cdot \mathbf{F}_i = 0$$ (conservation of charge)

with the ion flux $$\mathbf{F}_i = D_i \left(\nabla n_i + z_i n_i \nabla \phi \right)$$ drift

Poisson: $$\nabla \cdot (\epsilon \nabla \phi) = -\frac{e^2 n^*}{\epsilon_0 k T} \sum_i z_i n_i$$

- For ion species i:
  - n_i: relative concentration (with respect to scaling concentration n^*)
  - z_i: valence ($\pm$ 1)
  - D_i: (position-dependent) diffusion coefficient
- ϕ: relative electric potential energy with respect to the thermal energy ($\phi = \frac{e}{kT} U$ with U in Volts)
- ϵ: (position-dependent) relative permittivity
- T: temperature of the solvent
- e, ϵ_0, k: natural constants

The Nernst-Planck-Poisson System

describes ion drift through a static solvent.

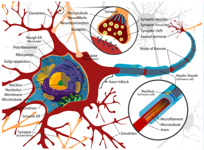
Nernst-Planck: $$\frac{\partial n_i}{\partial t} - \nabla \cdot \mathbf{F}_i = 0$$ (conservation of charge)

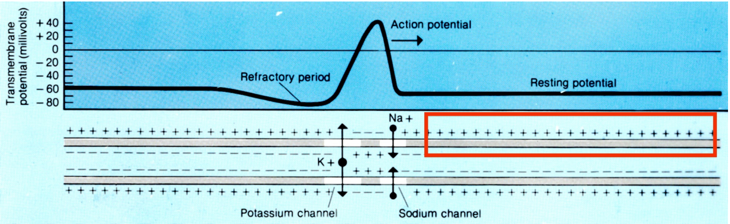
with the ion flux $$\mathbf{F}_i = D_i \left(\nabla n_i + z_i n_i \nabla \phi \right)$$

Poisson: $$\nabla \cdot (\epsilon \nabla \phi) = -\frac{e^2 n^*}{\epsilon_0 k T} \sum_i z_i n_i$$ charge density

- For ion species i:
  - n_i: relative concentration (with respect to scaling concentration n^*)
  - z_i: valence ($\pm$ 1)
  - D_i: (position-dependent) diffusion coefficient
- ϕ: relative electric potential energy with respect to the thermal energy ($\phi = \frac{e}{kT} U$ with U in Volts)
- ϵ: (position-dependent) relative permittivity
- T: temperature of the solvent
- e, ϵ_0, k: natural constants

Assumptions of a reduced model: Single Axon

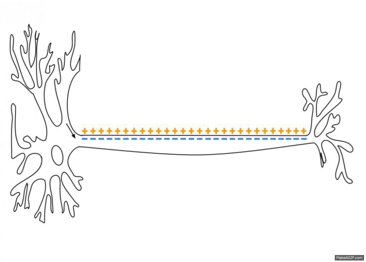


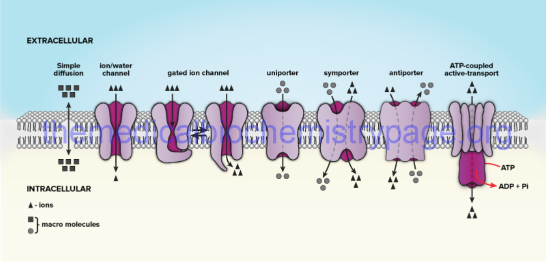


http://courses.washington.edu/psy222/psy222actionpotential.htm

- 3 ion species: Na^+, K^+, Cl^-
- homogeneous intra- and extracellular medium: water
- membrane thickness: 5 nm
- No ions inside the membrane $\Rightarrow$ Additional boundary conditions at membrane interfaces representing ion channels
- In a first approximation, an axon is a cylinder
  Assume rotational symmetry $\Rightarrow$ Reduction to 2D problem!

Special Part: Membrane





Membrane dynamics are taken from Hodgkin-Huxley model with a one voltage dependent one (voltage-independent) leak channel for each cation (Na^{+}, K^{+})

The Hodgkin-Huxley System

For each channel type C:

$I_C = g_C[\phi]$ $\qquad g_C$: conductance, $[\phi]$: membrane potential

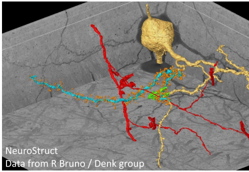
$$g_{\mathrm{K}_v} = \bar{g}_{\mathrm{K}_v} n^4$$
$$g_{\mathrm{Na}_v} = \bar{g}_{\mathrm{Na}_v} m^3 h$$

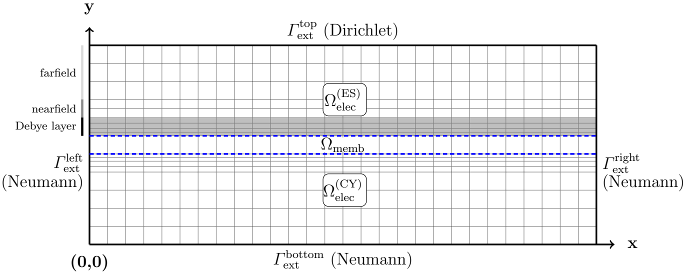
with maximum conductances $\bar{g}_{\mathrm{K}_v}, \bar{g}_{\mathrm{Na}_v}$ and gating particles $n, m, h \in [0\ 1]$, following time- and voltage-dependent kinetics

$$\frac{\mathrm{d}n}{\mathrm{d}t} = \alpha_n(\phi)(1-n) - \beta_n(\phi)n$$
$$\frac{\mathrm{d}m}{\mathrm{d}t} = \alpha_m(\phi)(1-m) - \beta_m(\phi)m$$
$$\frac{\mathrm{d}h}{\mathrm{d}t} = \alpha_n(\phi)(1-h) - \beta_h(\phi)h$$

Computational Domain / Boundary Conditions

Multidomain setup / Dimension reduction

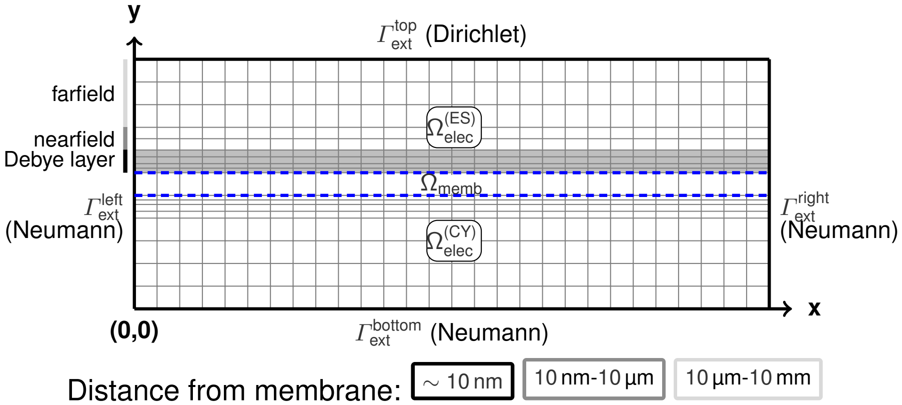




Internal Neumann boundary for concentrations:

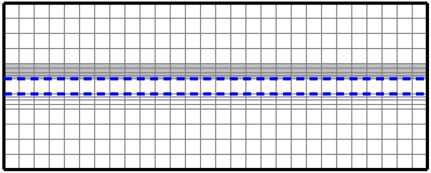
$$F_i(\mathbf{x}) = f_i^{memb}(\mathbf{x}) = g_i(\phi, t)\frac{kT}{e^2 z^2 n^*}\left(z[\phi] + \ln\frac{n_i^{ES}}{n_i^{CY}}\right)$$

Computational Domain



- Debye length needs to be resolved close to the membrane (Debye length $<<$ membrane thickness $\Rightarrow$ electrolytes are electrically decoupled)
- Grid is highly anisotropic ($dx = 100\,\mu m$, $dy_{min} = 0.5\,nm$, factor 200,000)

Implementation in DUNE



Grid:

- 2D Tensor grid
  - Sequential: `UGGrid`
  - Parallel: `YaspGrid` + `GeometryGrid`
  - Test stage: Tensor-`YaspGrid`
- Multiple domains: `MultidomainGrid`

Discretization:

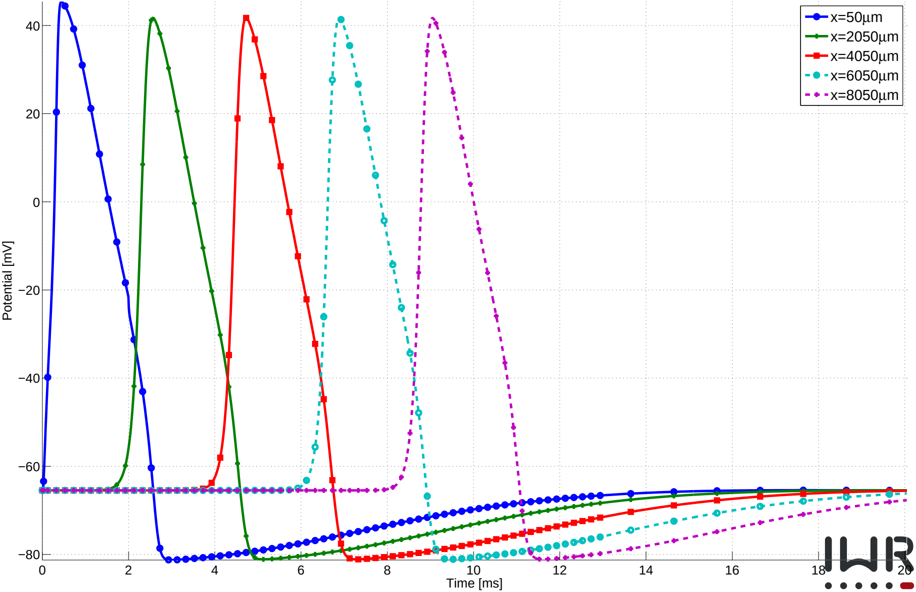
- Heidelberg ⇒ PDELab

Used modules: Core modules, `dune-multidomaingrid`, `dune-multidomain`, `dune-pdelab`

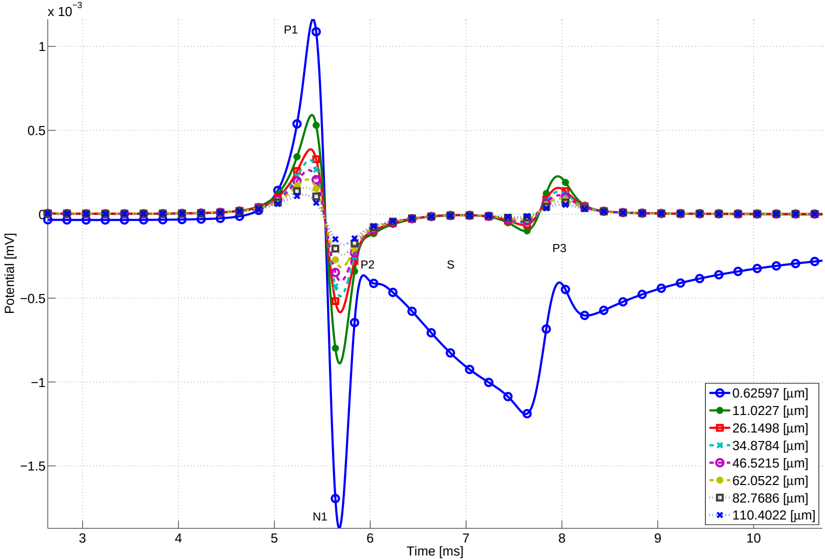
Numerical methods

- Linear Finite Elements (Continuous Galerkin)
- Time stepping: Implicit Euler
- PDE system is solved fully-coupled (Newton's method with line search) $\Rightarrow$ significantly higher time step!
- ODE system for HH kinetics (membrane flux) is solved separately (implicit Euler), once at the beginning of each time step
- linear solvers (ISTL):
  - direct solver (SuperLU)
  - BiCGStab + ILUn preconditioner
  - Restarted GMRes + AMG preconditioner (ILU smoother)
  - In parallel: Use overlapping versions of the above solvers

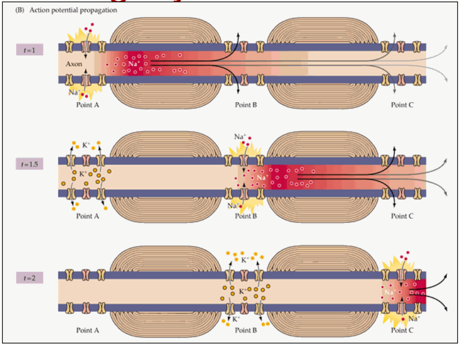
Intracellular potential



LFP signals for increasing distance from the membrane

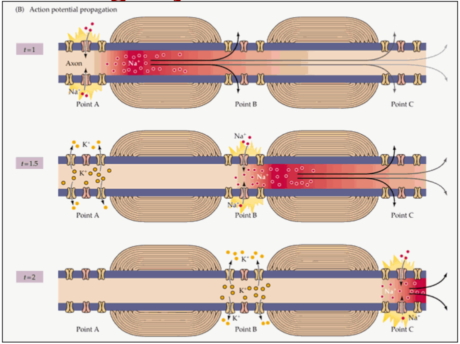


Adding myelin to the axon

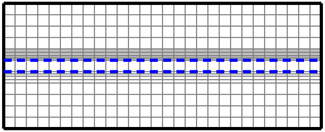


- Myelin serves as an insulating layer around the axon $\Rightarrow$ faster conduction
- Areas between myelin sheats are called nodes of Ranvier
- Action ptential is “refreshed” at nodes by membrane currents
- “Saltatory conduction” from node to node

Adding myelin to the axon



?



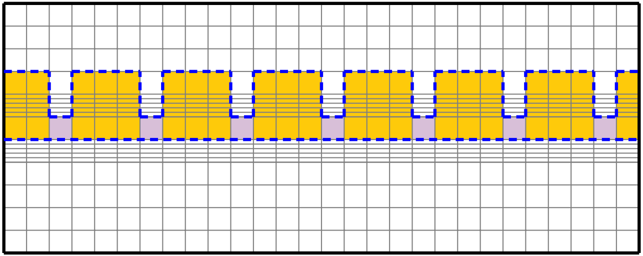
How to get the new geometry into the existing setup?

- Myelin serves as an insulating layer around the axon $\Rightarrow$ faster conduction
- Areas between myelin sheats are called nodes of Ranvier
- Action ptential is “refreshed” at nodes by membrane currents
- “Saltatory conduction” from node to node

How to get the new geometry into the existing setup?

Precondition: I want to keep my beloved tensor grid!

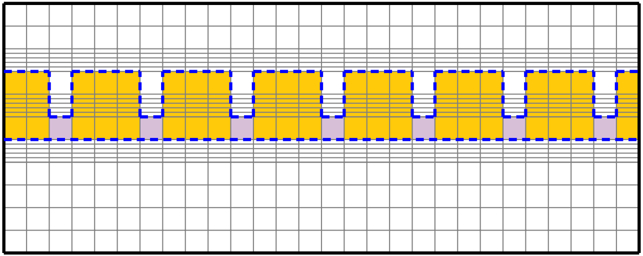
- One could explicitly include the myelin sheath



How to get the new geometry into the existing setup?

Precondition: I want to keep my beloved tensor grid!

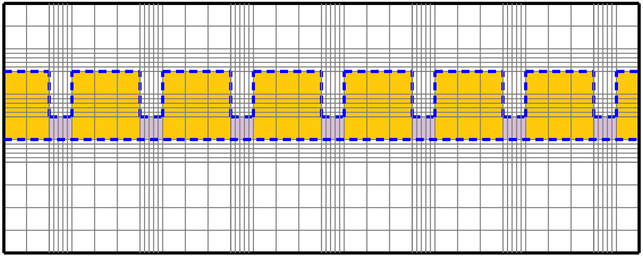
- One could explicitly include the myelin sheath
- ... but we would have to resolve the Debye layer twice!



How to get the new geometry into the existing setup?

Precondition: I want to keep my beloved tensor grid!

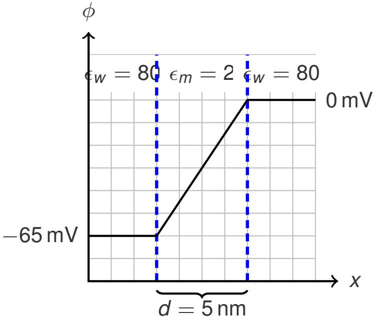
- One could explicitly include the myelin sheath
- ... but we would have to resolve the Debye layer twice!
- A vertical part of the membrane appears, how to handle that?



How to get the new geometry into the existing setup?

Observation: Potential is approximately a linear function over the membrane

$\Rightarrow$ A membrane with permittivity $\epsilon = 2$ and thickness $d = 5\,\text{nm}$ will cause the same potential decay than one with $\epsilon = 4$ and $d = 10\,\text{nm}$



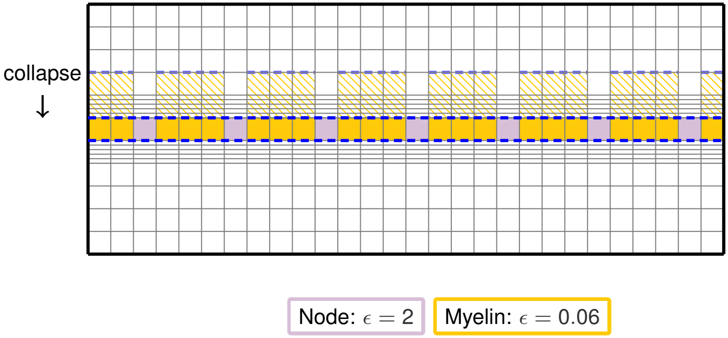
$C = \frac{\epsilon_w}{\epsilon_m}$ is the factor by which the potential gradient changes

How to get the new geometry into the existing setup?

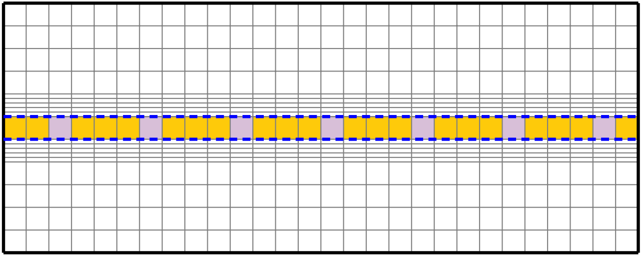
Idea: Do not include node geometry explicitly, but calculate effective permittivities for a “collapsed” myelin sheath!

- Membrane: Thickness $d_{\text{node}} = 5\,\text{nm}$, permittivity $\epsilon_{\text{node}} = 2$
- Myelin: Thickness $d_{\text{myelin}} = 500\,\text{nm}$, permittivity $\epsilon_{\text{myelin}} = 6$

$\Rightarrow$ Effective permittivity $\epsilon^{\text{eff}}_{\text{myelin}} = \epsilon_{\text{myelin}} \frac{d_{\text{node}}}{d_{\text{myelin}}} = 0.06$



Grid for the myelinated axon

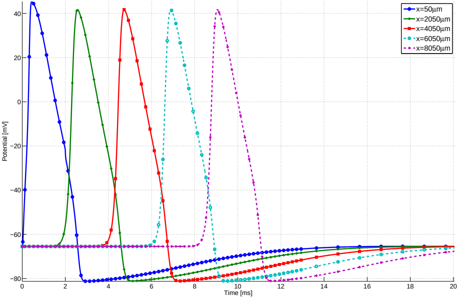


- Length of a node of Ranvier: 1 µm
- Length of the axon: 10 mm
- Equidistant spacing in x-direction not possible anymore
- Finer resolution at nodes of Ranvier and transitions to myelin, coarser spacing at myelin sheath

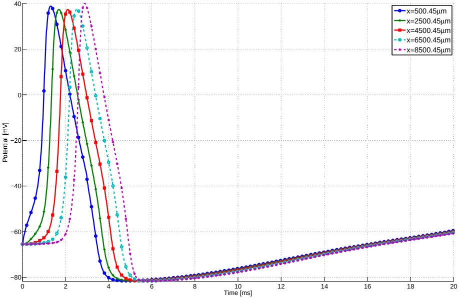
⇒ Strongly varying grid sizes in both x- and y-direction

Intracellular potential

Propagation is faster!



$v \approx 1 \, \frac{\text{m}}{\text{s}}$

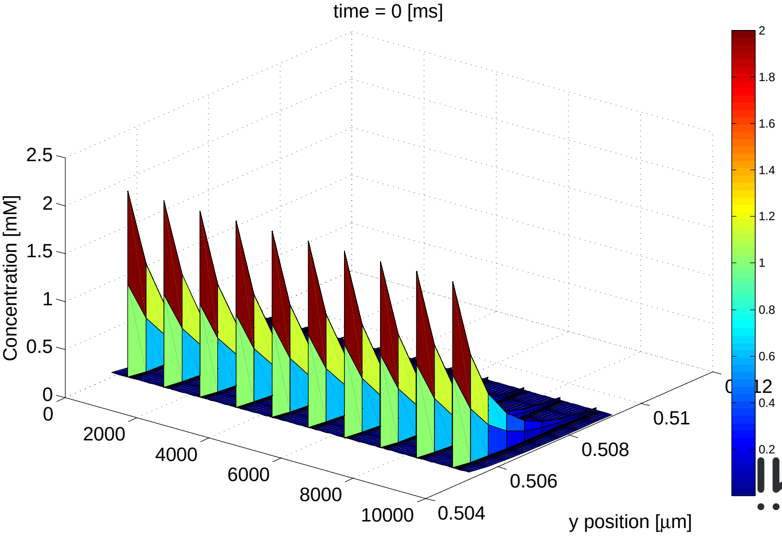


$v \approx 5 \, \frac{\text{m}}{\text{s}}$

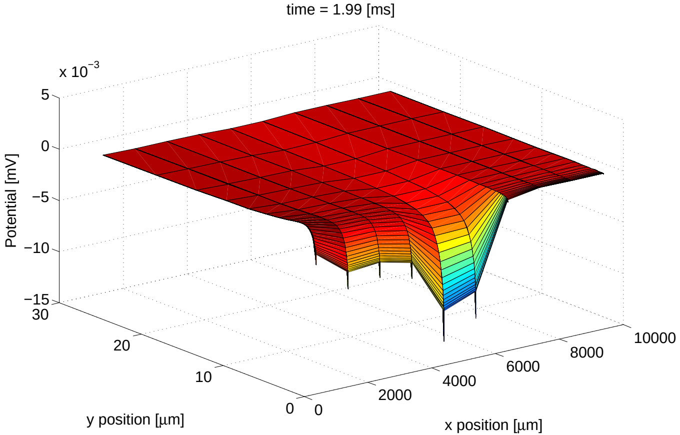
Equilibrium State

Varying permittivities ⇒ different membrane potentials

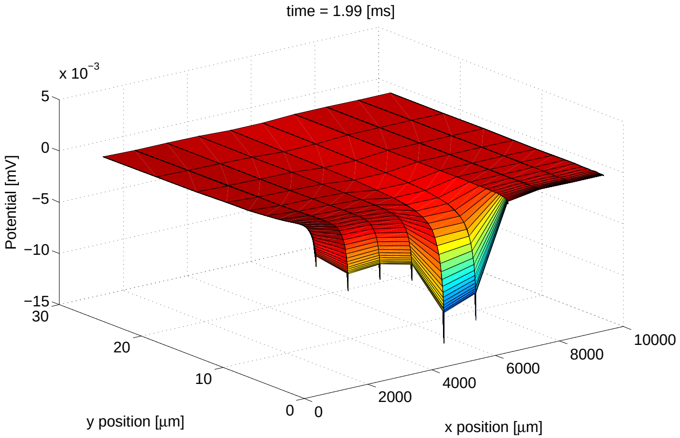
⇒ “comb-shaped” equilibrium concentration profile



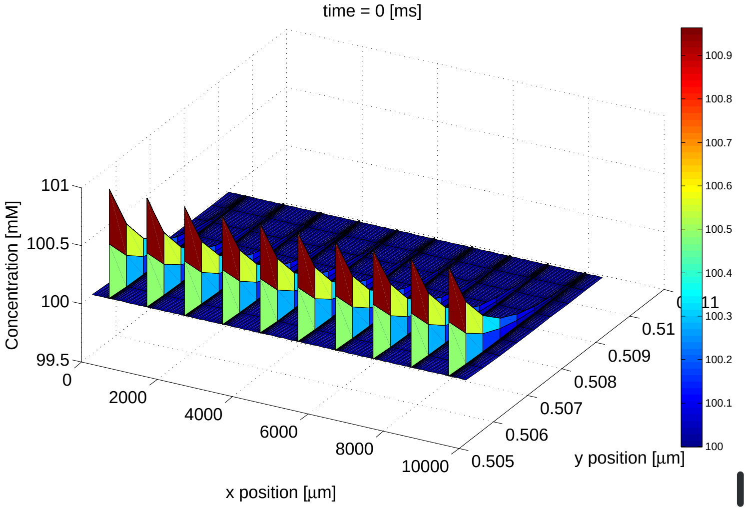
Extracellular potential near membrane (nodes of Ranvier)



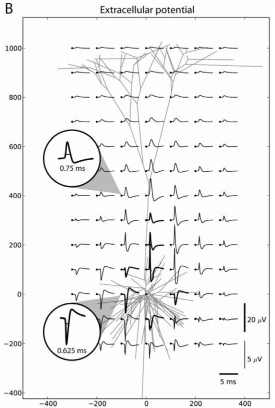
Extracellular potential near membrane (myelin)



Extracellular concentrations near membrane (myelin + nodes)



Comparison with an effective model: Line Source Approximation (LSA)



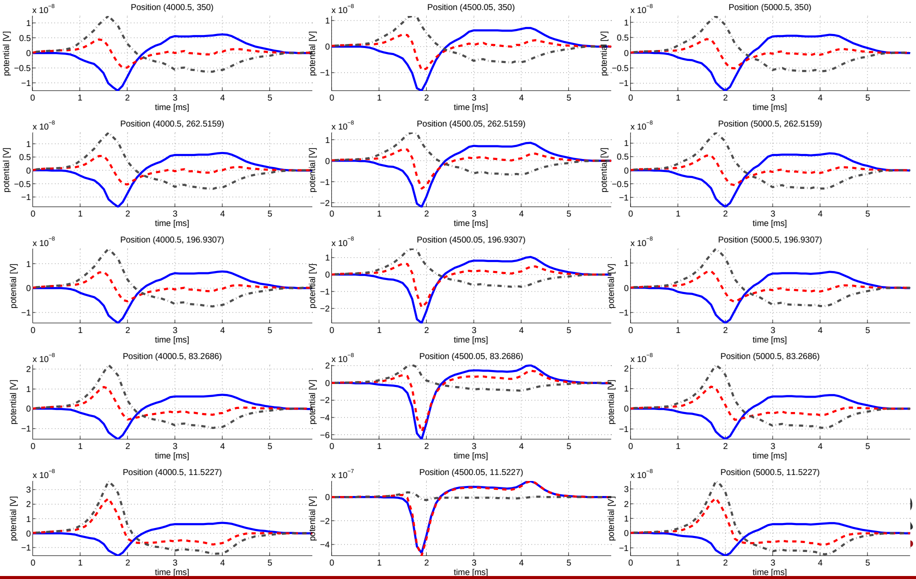
Pettersen & Einevoll 2009

Potential of one line segment:

$$\Phi(r,h) = \frac{\rho I}{4\pi \Delta s} \log \left| \frac{\sqrt{h^2 + r^2} - h}{\sqrt{l^2 + r^2} - l} \right|$$

- ρ: resistivity of the extracellular medium
- I: Total membrane current of line segment
- Δs: length of line segment
- r: radial distance from the line
- h: longitudinal distance from the end of the line
- $l = \Delta s + h$: distance from the start of the line

LFP signals for increasing distance from the membrane



Summary

- Electrodiffusion simulations of the brain using Poisson-Nernst-Planck equations
- Usage of a stacked meta grid hierarchy (`YaspGrid` in a `GeometryGrid` in a `MultidomainGrid`)
- Parallelization with custom overlapping partitioning (`YLoadBalance`) on a low number of processors $p \leq 10$
- 3D results, but only 2D cost by exploiting cylinder symmetry
- Results generally show deviations from effective LSA model for both unmyelinated and myelinated axon, good agreement at nodes of Ranvier

Outlook

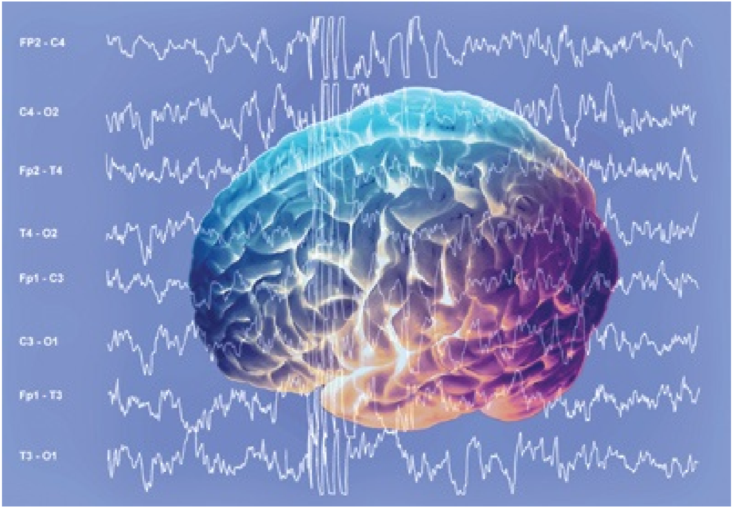
Next steps:

- Improve the modeling of the intra- and extracellular medium, i.e. effective diffusion coefficients and permittivities

- Run larger simulations of myelinated axon
  - larger domain
  - finer resolution (remove spurious oscillations, check grid convergence)

- Improve effective LSA model: Include AP echo as an additional term into the equation

Thank you for your attention!



IWR