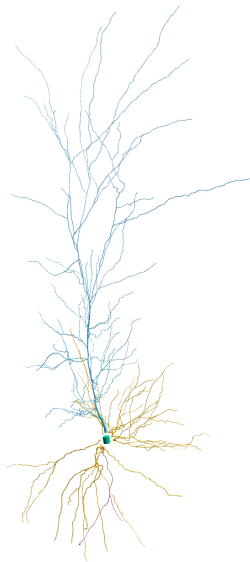


ARBOR

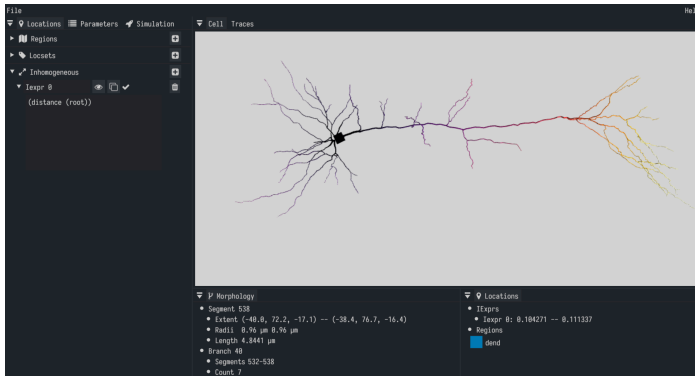
A TUTORIAL

Mar 2025 | S. Díaz, T. Hater, H. Lu | Forschungszentrum Jülich



- Design goals:
Usability, performance portability, and strong separation of concerns.
- Made to exploit Multicore, SIMD, and GPU
- Scales extremely well: 768 nodes \times 4 A100 on JUWELS booster. 70 Million cells.
- HPB since 2016 by CSCS and FZJ
- C++17 and Python3
- Linux and MacOS
- FOSS with a permissive BSD3 license
- Modern dev cycle in the open:
Git, Code Review, CI, tests, sanitizers, ...

The Larger Ecosystem



- NeuroML2
github.com/thorstenhater/nmlcc
- SONATA
github.com/thorstenhater/cantata

- GUI github.com/arbor-sim/gui
- Benchmarks and Validation github.com/arbor-sim/nsuite
- BluePyOpt
docs.arbor-sim.org/en/latest/tutorial/single_cell_bluepyopt.html

Getting started v0.10

```
# isolate
python -mvenv arb
source arb/bin/activate
# install
pip install arbor
# test
python -c \
    'import arbor as A; print(A.config())'
```

Note

For technical (pip) reasons, this will only enable threads and MPI **not** GPU and SIMD. For these features, a (simple) local build is needed.
See our documentation.

Getting started v0.10

```
# isolate
python -menv arb
source arb/bin/activate
# install
pip install arbor
# test
python -c \
    'import arbor as A; print(A.config())'
```

Note

For technical (pip) reasons, this will only enable threads and MPI **not** GPU and SIMD. For these features, a (simple) local build is needed. See our documentation.

```
import arbor as A
from arbor import units as U
# Define labels and cell
lbl = A.label_dict().add_swc_tags()
ctr = '(location 0 0.5)'
dec = (A.decor()
    # Set initial membrane potential
    .set_property(Vm=-55 * U.mV)
    # Assign mechanisms to soma and dendrite
    .paint("soma", A.density("hh"))
    .paint("dend", A.density("pas"))
    # Attach stimulus and spike detector
    .place(ctr, A.iclamp(tstart=10*U.ms,
        ↳ duration=1*U.ms, current=5*U.nA), "ic")
    .place(ctr, A.threshold_detector(-10*U.mV),
        ↳ "det"))
# Load morphology data
mrf = A.load_swc_arbor("cell.swc")
# Make and run a single cell model
sim = A.single_cell_model(mrf, dec, lbl)
sim.run(tfinal=100 * U.ms, dt=0.0025 * U.ms)
```

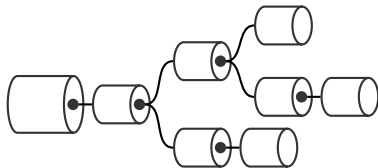
The Bio-Physical Model

Consider a cell as

- **charges** a liquid solution of ions
- **membrane** a bi-lipid layer, impermeable to these ions
- **ion channels** a set of proteins acting as gates

Note

The interplay of ion channels, in particular their dependence on the the membrane potential, creates action potentials and dynamics of the cells.

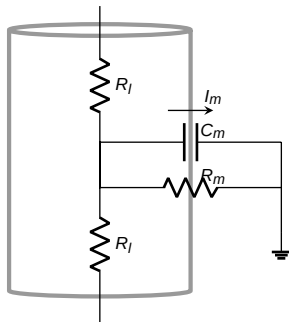


The Cable Equation

$$\frac{1}{C_m} \partial_t U_m = \partial_x \left(\frac{1}{R_I} \partial_x U_m \right) + I_m$$

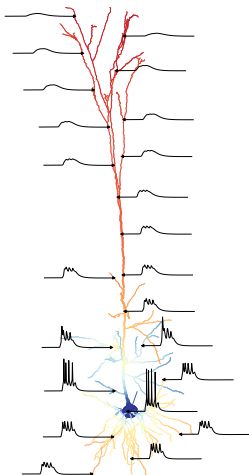
- 1st principles, Maxwell and lots of approximations.
- Similar to the equation for telegraph cables.
- C_m : Membrane capacitance.
- I_m : transmembrane current.
- R_I : axial resistance.

- Applied math occurs: FVM, implicit Euler, Lie-split.



- I_m captures ion channels

Feature Highlights



Neuroscience

- Stochastic Differential Equations
- Ionic diffusion along the dendrite
- Advanced gap junction dynamics
- Plastic connection table

Infrastructure

- Plugin API for defining ion channels
- Bi-directional spike-exchange API
- Serialisation/Deserialisation

Upcoming in 0.10

- Units at API level
- High-level network builder

Compatibility and Interoperability

Morphology

- Neurolucida ASC
- Neuromorpho SWC
- NeuroML2 NML

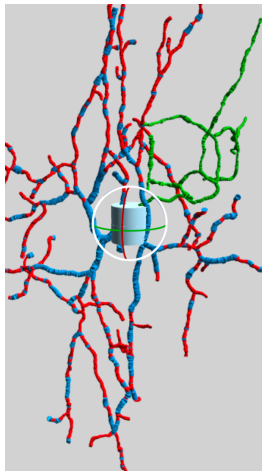
Ion channels

- Neuron NMODL
- NeuroML2 NML

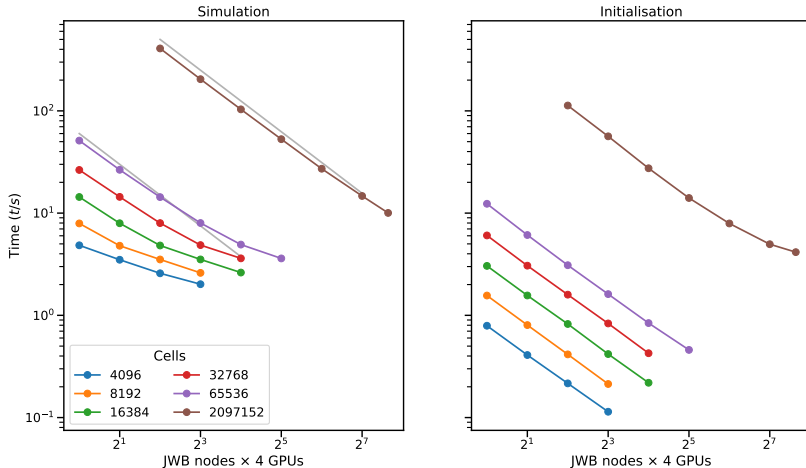
Parametrisations and Networks

- NeuroML2
- SONATA
- both realized as external translators

+ internal formats



Scalability and Performance



Scaling on JUWELS Booster using a benchmark with cells from the Allen Brain Atlas.

Direct comparison against NEURON on CPU workloads indicates a speed-up of roughly $10\times$.

Our interests

- Large scale simulations, especially multi-scale science.
- Vertical integration with other tools.
- Getting more people on board.

Contact, Code, and Documentation

- <https://gitter.im/arbor-sim/community>
- <https://github.com/arbor-sim/arbor>
- <https://docs.arbor-sim.org>



Our story continues: <https://go.fzj.de/arbor-for-ebrains>

Chat: <https://app.gitter.im/#/room/#arbor-tutorials:gitter.im>

Our interests

- Large
- Vertical
- Getting people on board.



scale science.

Contact, Code, and Documentation

- <https://gitter.im/arbor-sim/community>
- <https://github.com/arbor-sim/arbor>
- <https://docs.arbor-sim.org>



Our story continues: <https://go.fzj.de/arbor-for-ebrains>

Chat: <https://app.gitter.im/#/room/#arbor-tutorials:gitter.im>