IBM Research Division

The Neural Tissue Simulator: How to specify and scale an arbitrary number of compartment variables over an arbitrary number of compartments

Presentation to NeuroML workshop
Convergence in Computational Neuroscience 2012

James Kozloski, Research Staff Member
IBM T. J. Watson Research Center, Yorktown Heights, NY
kozloski@us.ibm.com, ibm.co/kozloski

March 2012

tinyurl.com/neuraltissuesim

frontiers in
NEUROINFORMATICS

METHODS ARTICLE
published: xx August 2011
doi: 10.3389/fninf.2011.00015

An ultrascalable solution to large-scale neural tissue simulation

James Kozloski[1]* *and John Wagner*[2]

[1] *Biometaphorical Computing, Computational Biology Center, IBM Research Division, IBM T. J. Watson Research Center, Yorktown Heights, NY, USA*
[2] *Victorian Life Sciences Computation Initiative, IBM Research Collaboratory for Life Sciences – Melbourne, Carlton, VIC, Australia*

Edited by:
Markus Diesmann, RIKEN Brain Science Institute, Japan

Reviewed by:
Michael Hines, Yale University, USA
Abigail Morrison, Bernstein Center Freiburg, Germany

Correspondence:
James Kozloski, Biometaphorical Computing, Computational Biology Center, IBM Research Division, IBM T. J. Watson Research Center, 1101 Kitchawan Road, Room 05-144, Yo Heights, NY, USA.
e-mail: kozloski@us.ibm.com

Neural tissue simulation extends requirements and constraints of previous neuronal and neural circuit simulation methods, creating a tissue coordinate system. We have developed a novel tissue volume decomposition, and a hybrid branched cable equation solver. The decomposition divides the simulation into regular tissue blocks and distributes them on a parallel multithreaded machine. The solver computes neurons that have been divided arbitrarily across blocks. We demonstrate thread, strong, and weak scaling of our approach on a machine with more than 4000 nodes and up to four threads per node. Scaling synapses to physiological numbers had little effect on performance, since our decomposition approach generates synapses that are almost always computed locally. The largest simulation included in our scaling results comprised 1 million neurons, 1 billion compartments, and 10 billion conductance-based synapses and gap junctions. We discuss the implications of our ultrascalable Neural Tissue Simulator, and with our results estimate requirements for a simulation at the scale of a human brain.

Keywords: neural tissue, simulation, parallel computing, distributed computing, Hodgkin–Huxley, numerical methods, ultrascalable, whole-brain

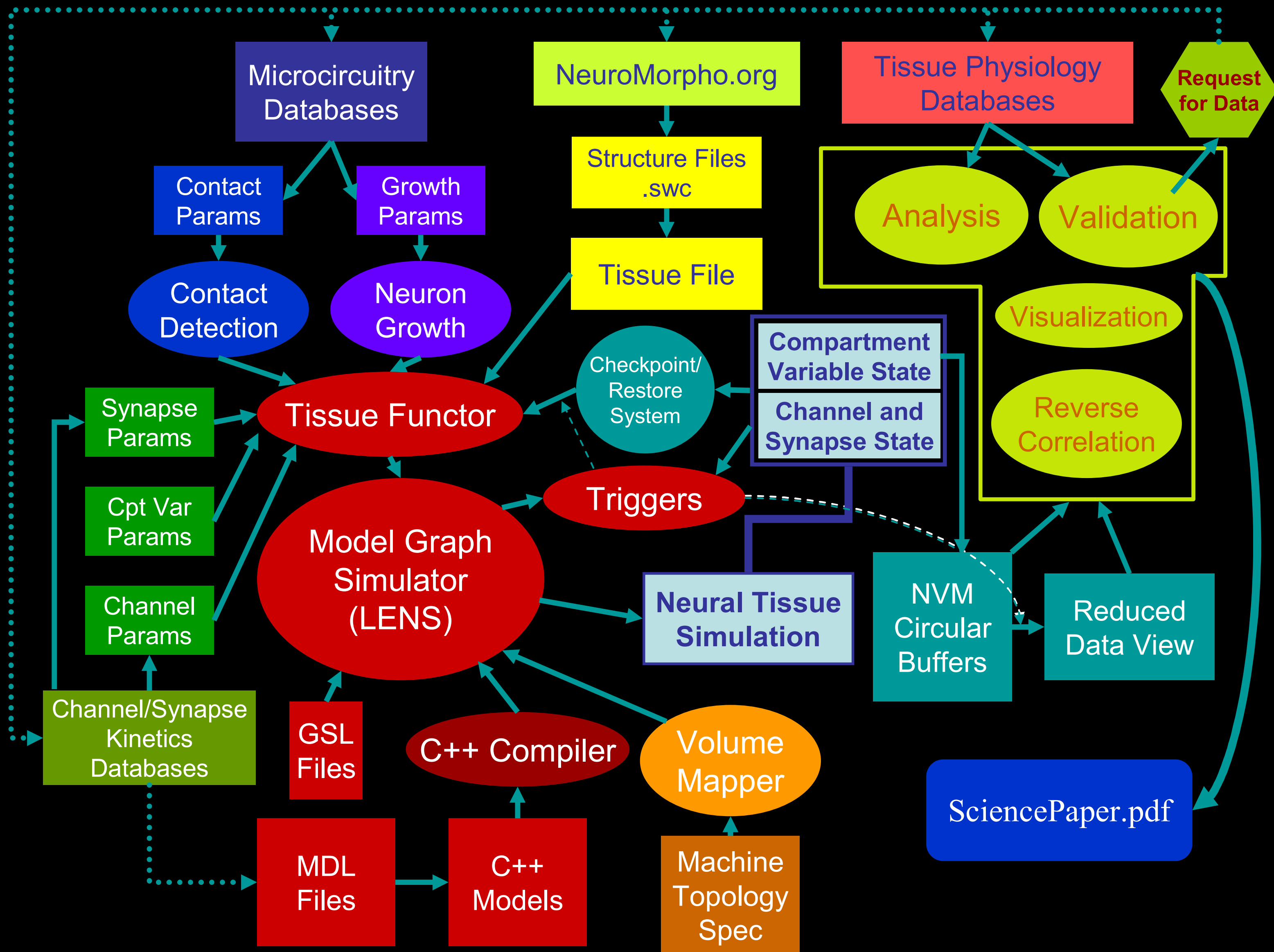
Neural Tissue Simulator: Goals

- Develop a simulator capable of testing mappings to various machine architectures
  - Parallel, Multithreaded
- Support for high level, abstract model definitions and simulation specifications
  - Model Graphs
- Extensible simulator, able to map arbitrary, domain level models directly to a variety of data arrangements and computational implementations
  - Code Generation

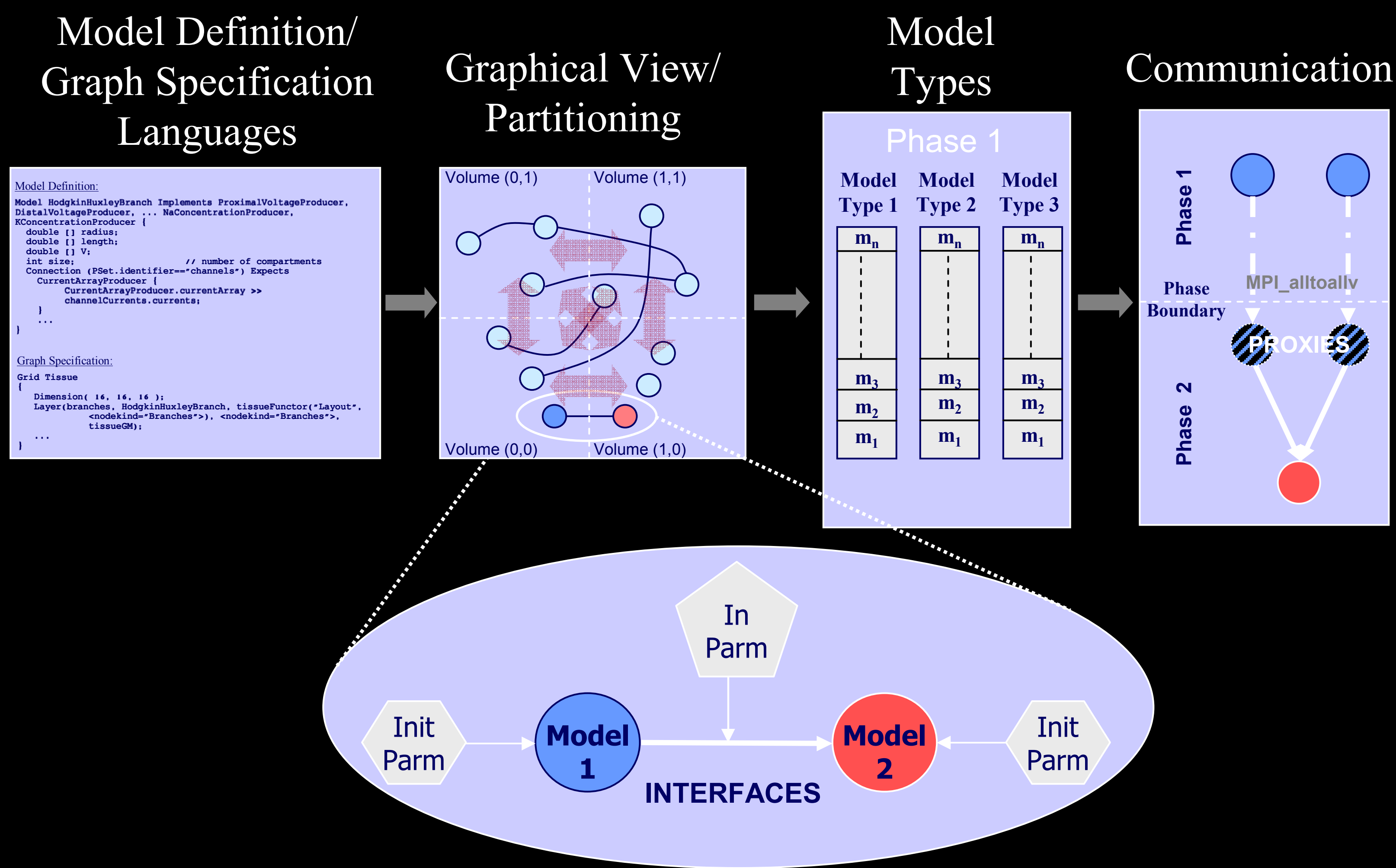
Overview

- Model Graph Simulator
  - Model Definition
  - Graph Specification
- Scaling

Neural Tissue Simulator Workflow



Model Graph Simulator: Infrastructure

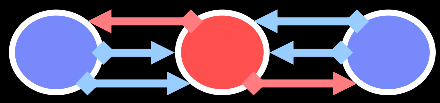


Model Graph Simulator: Infrastructure

Architectural Overview

- Language for expressing model state, computational phases, communicated state, and model interfaces (MDL)
- Language for composing arbitrary parameterized graphs (GSL)
- Automatic partitioning into work units for multi-threaded execution (SMP)
- Dynamically constructed, simulation-specific MPI collective communication for multi-process execution of computational phases (MPP)

Model Definition: Interfaces



```
Node NaChannel Implements ConductanceArrayProducer, ReversalPotentialProducer
{
  double [] m;
  double [] h;
  double [] g;
  double [] gbar;
  double []* V;
          ...

  Connection Pre Node (PSet.identifier=="compartment") Expects VoltageArrayProducer
  {
    VoltageArrayProducer.voltageArray >> V;
  }

  Connection Pre Node (PSet.identifier=="IC") Expects NaConcentrationProducer {
    NaConcentrationProducer.Na >> Shared.Na_IC;
  }
```

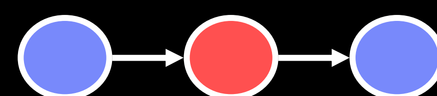
Model Definition: Phases

```
InitPhases = { initialize };
RuntimePhases = { run1, run2, run3, run4, run5, run6 };

NodeType HHJunction { predictState->run1,
                      correctState->run6 };

NodeType HHBranch { forwardEliminateCO0->run2,
                    forwardEliminateCO1->run3,
                    backSubstituteCO1->run4,
                    backSubstituteCO0->run5 };
```

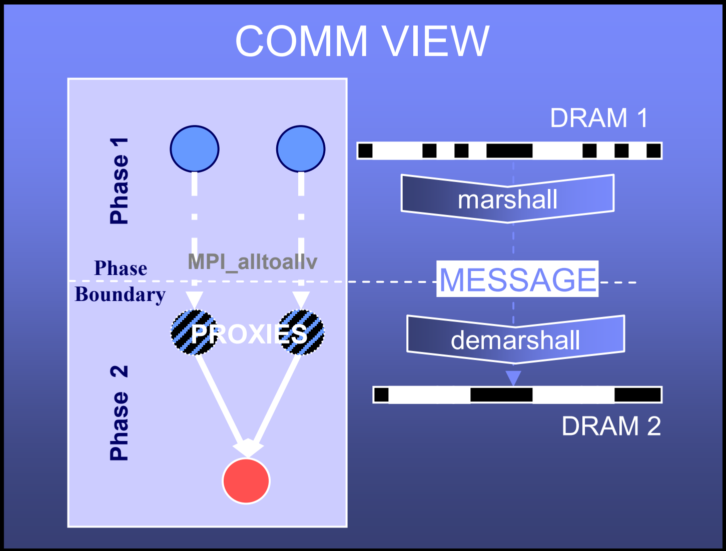
Time Step

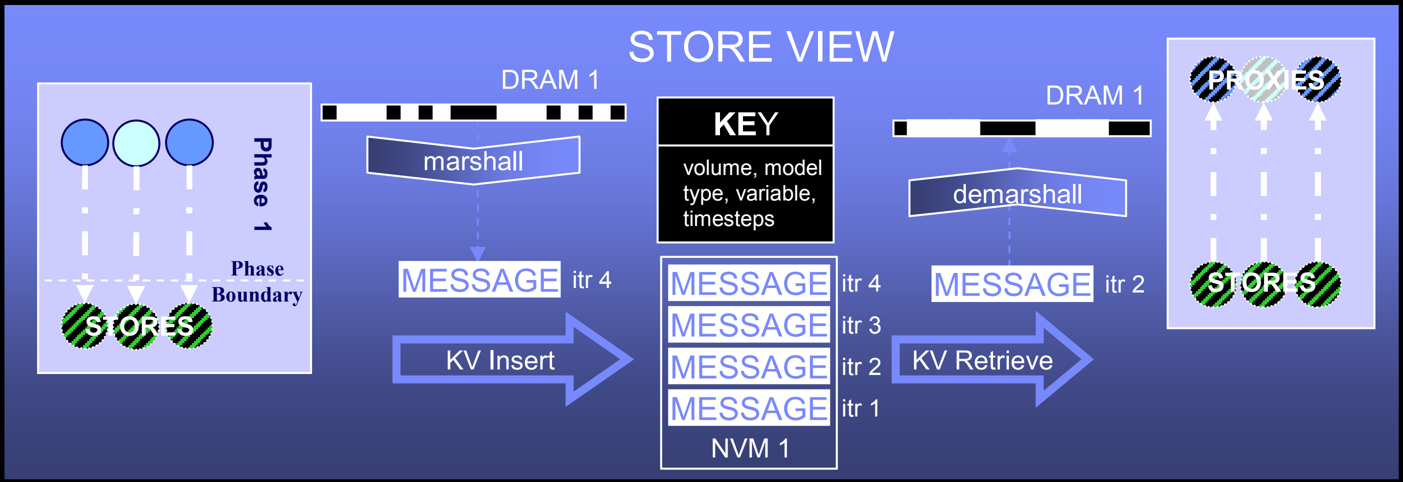
≈

Multiphase Algorithm



Data Movement: Marshalling and Demarshalling





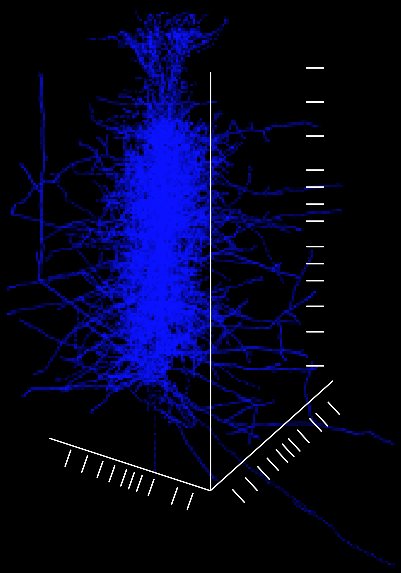
Graph Specification: Tissue Composition

SIMULATION APPROACH:

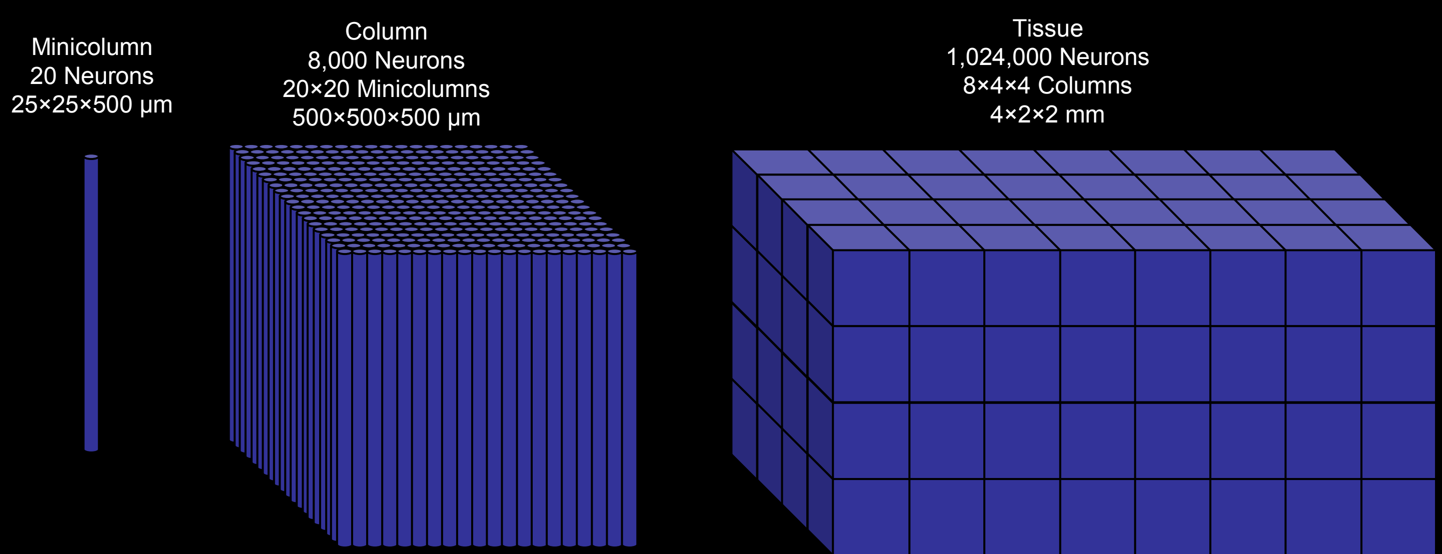
• Distribute tissue points weighted by computational complexity

• Scale out tissue simulation across all three dimensions

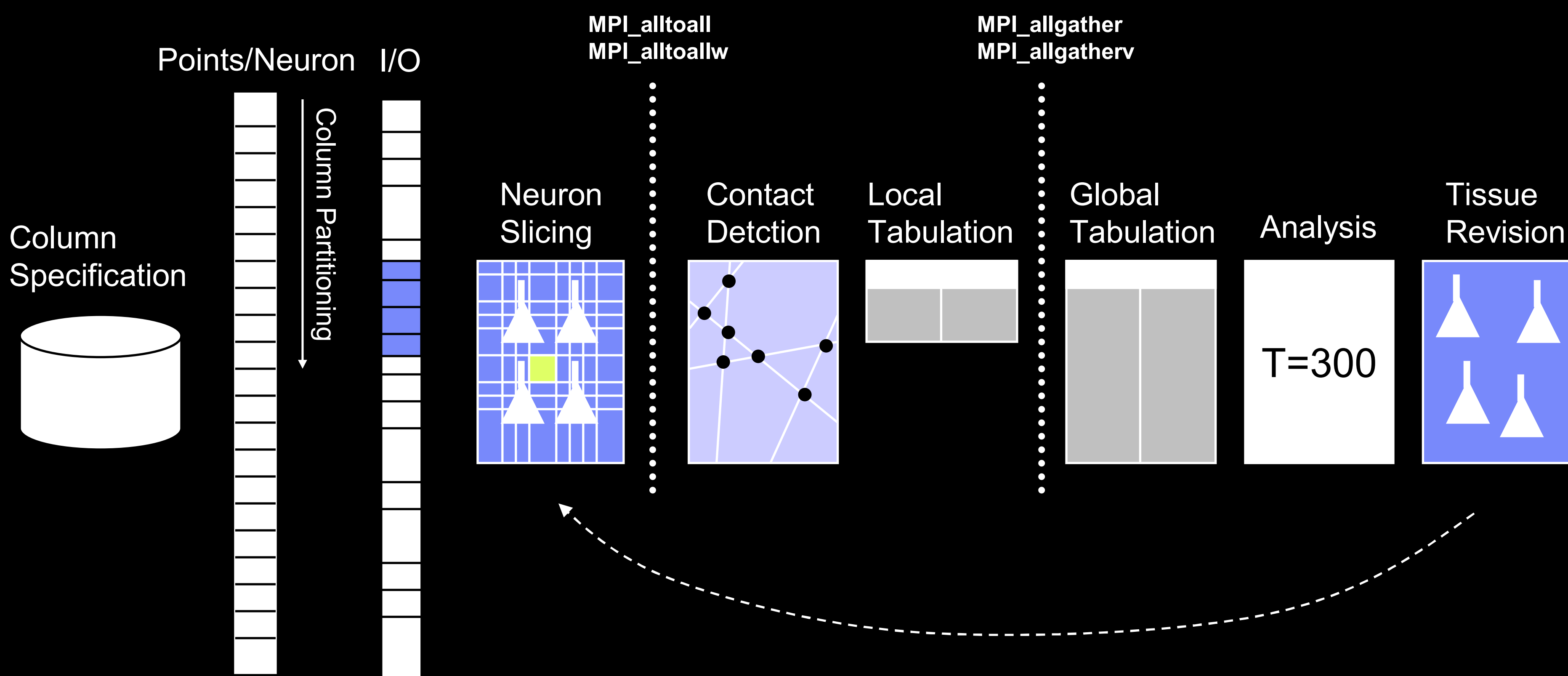
• Maintain realistic neuron and synapse densities at each scale



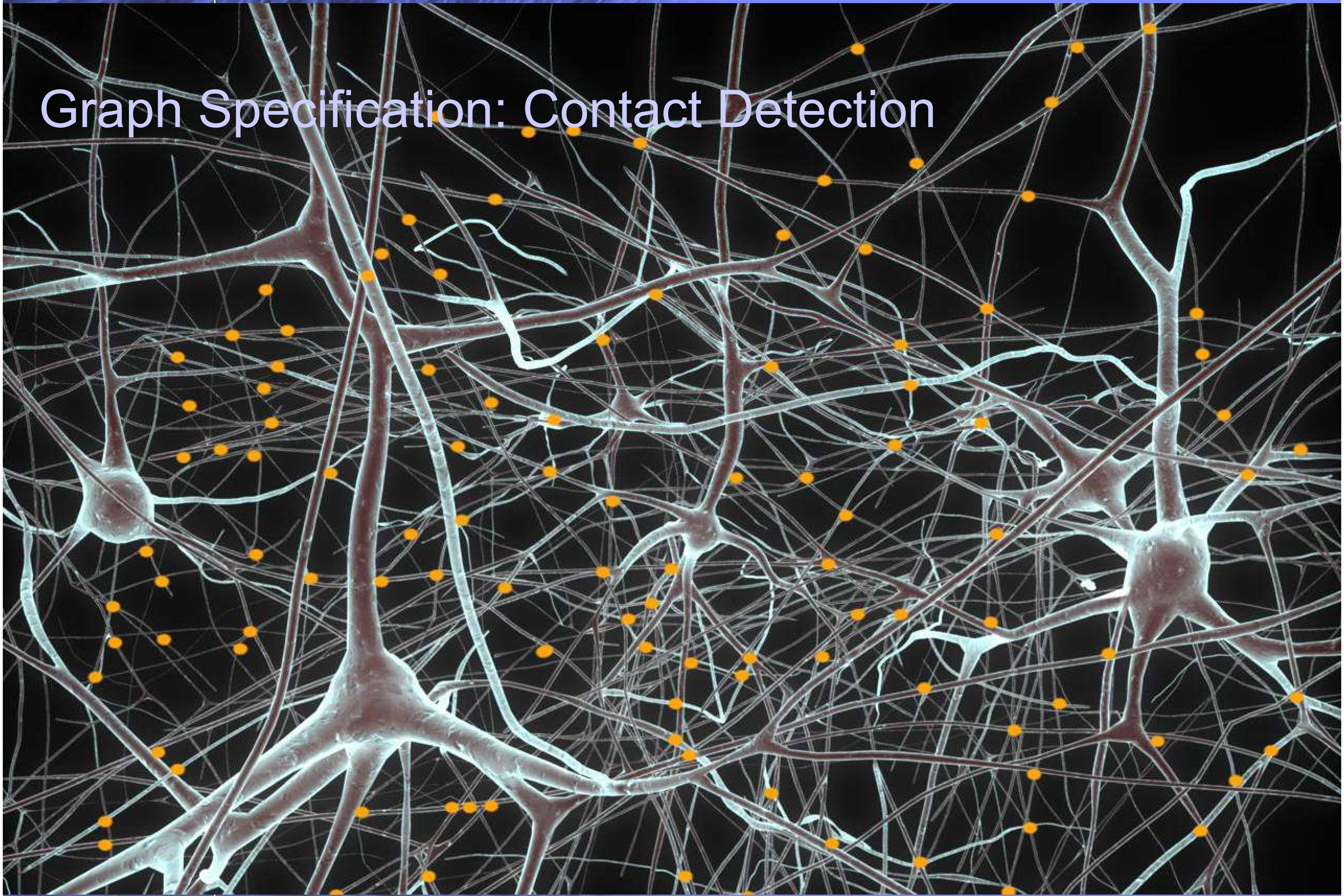
Simulation Element	Number	Processor Balance
Neurons	1,024,000,	N/A
Branches	344,474,059	84,100 ± 7,406
Junctions	208,947,659	51,012 ± 4,026
Compartments	1,083,289,600	264,475 ± 7,582
Na Channels	330,613,914	80,716 ± 7,440
KDR Channels	330,613,914	80,716 ± 7,440
AMPA Synapses	8,186,972,360	1,998,772 ± 720,155
GABAA Synapses	2,255,068,948	550,553 ± 169,064
Connexons	7,626,124	1,861 ± 820



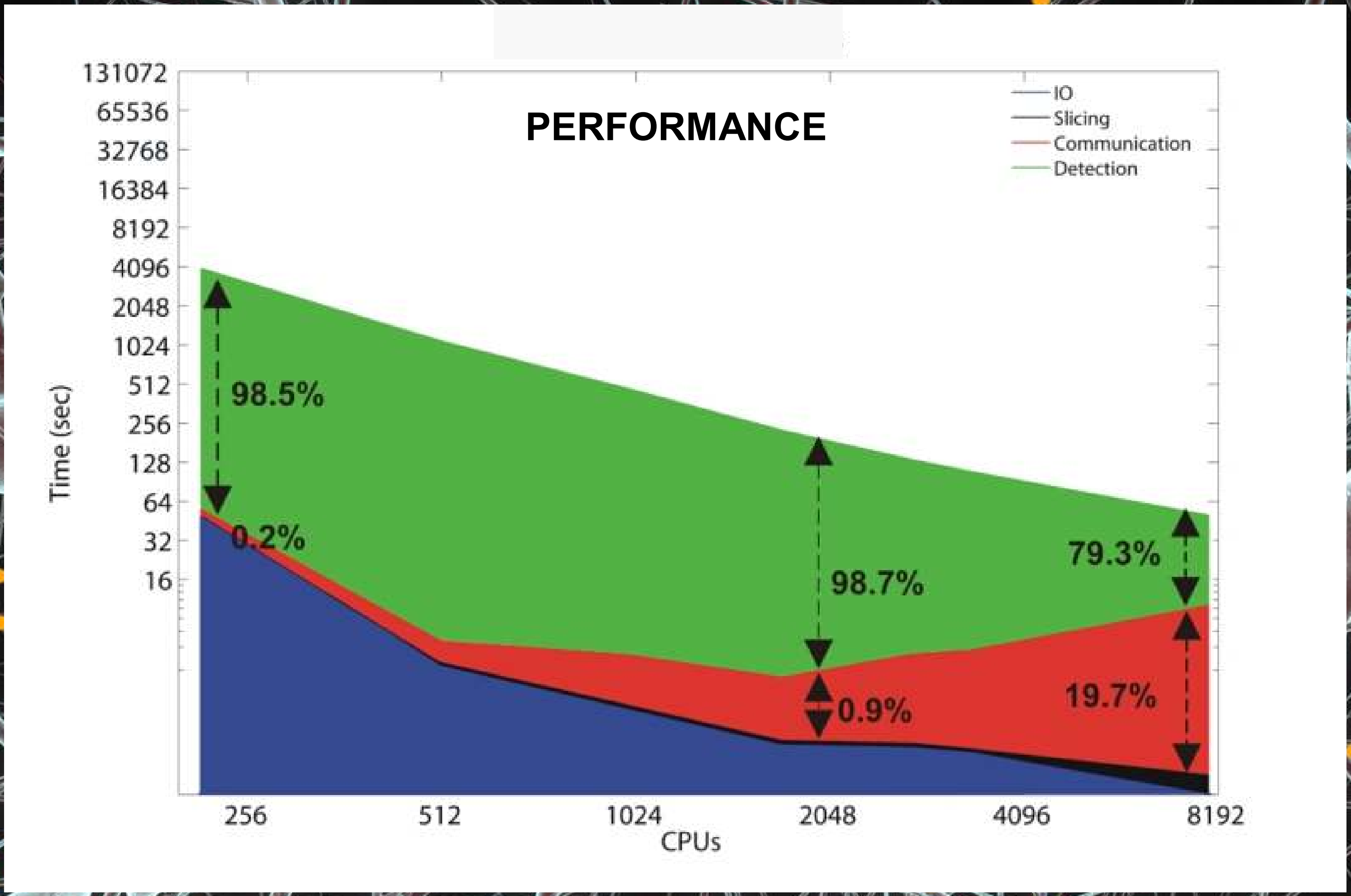
Graph Specification: Contact Detection



Graph Specification: Contact Detection



Graph Specification: Contact Detection

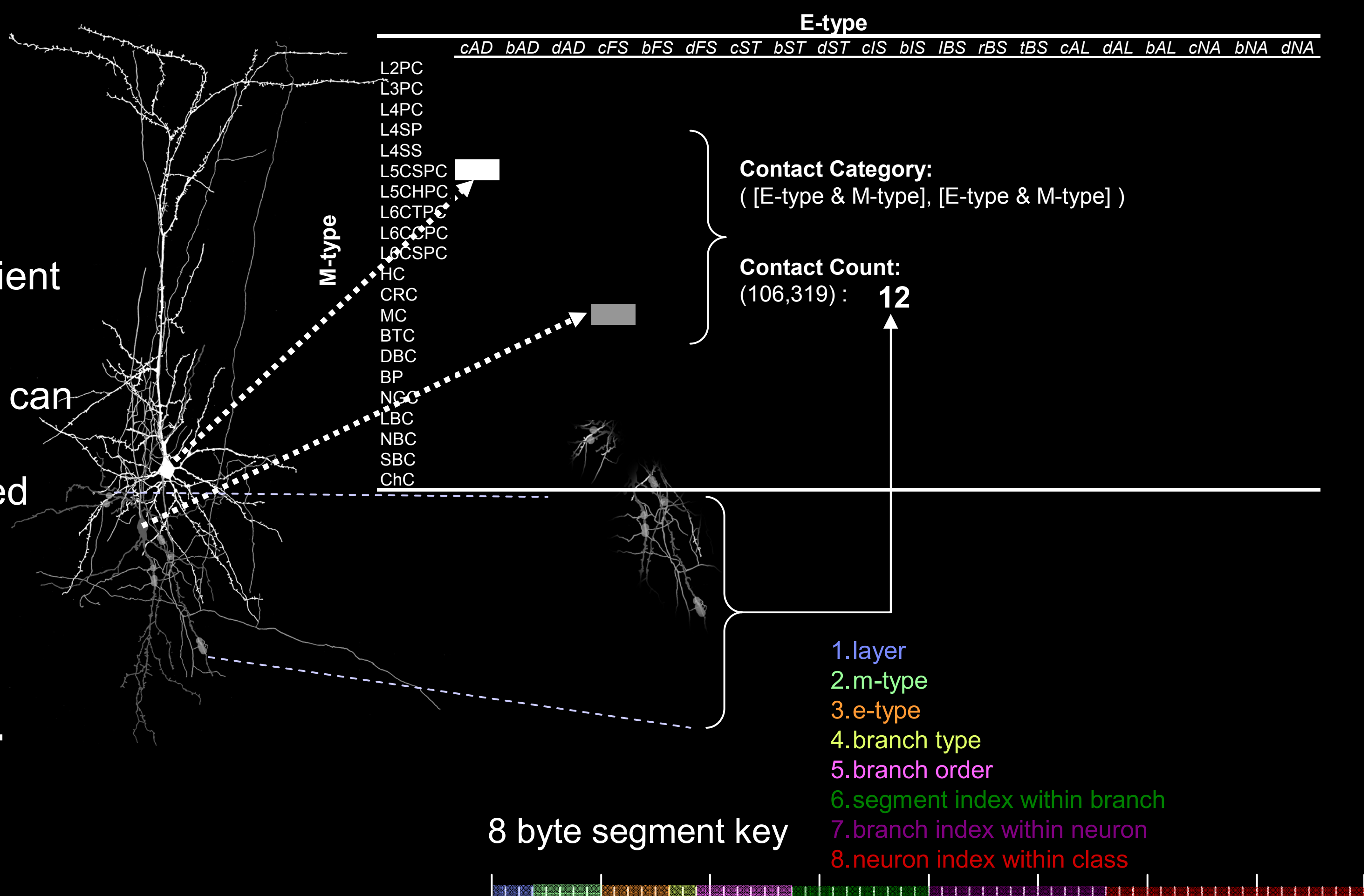


Graph Specification: Contact Detection

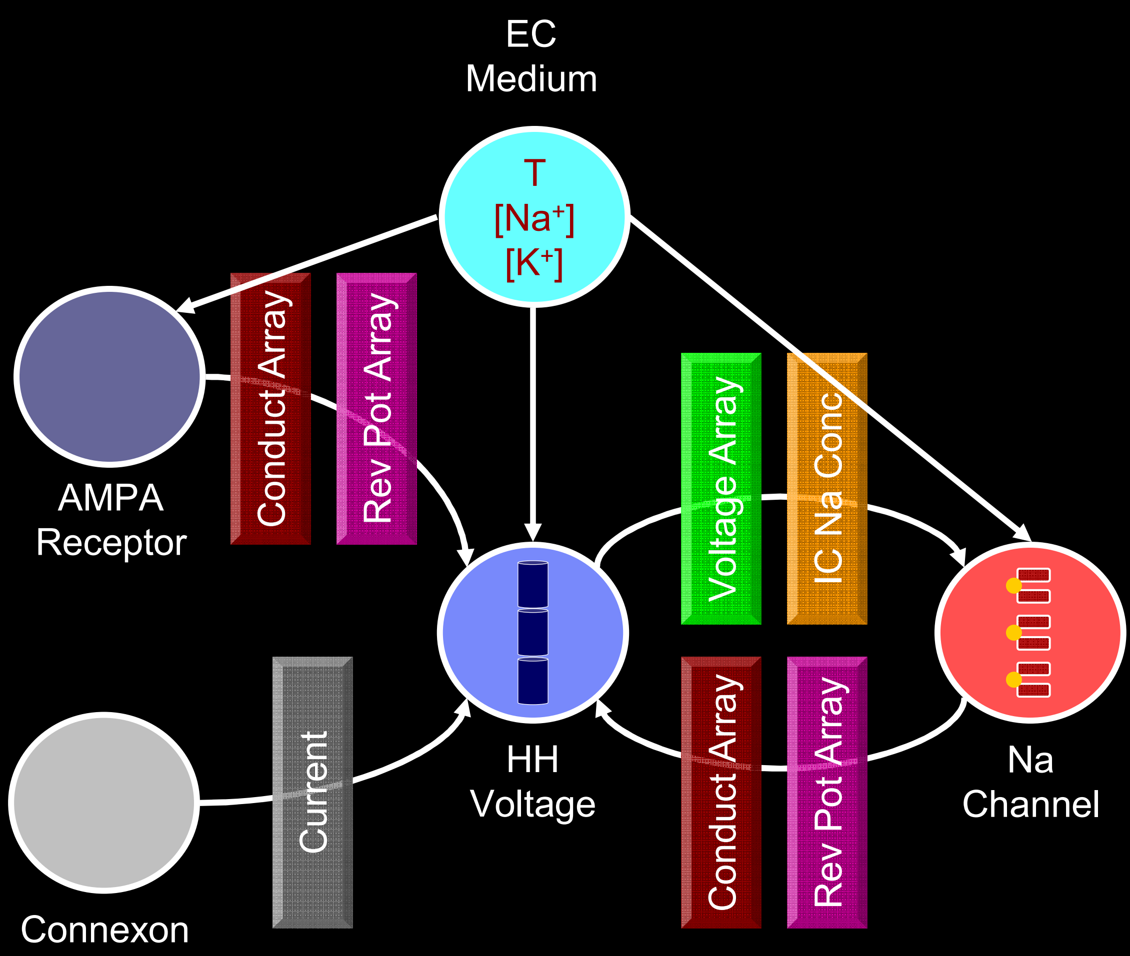
- New algorithm optimized to run multithreaded on Blue Gene/P’s 4-core compute nodes
- 25.5 billion contacts in 2.5 hours
- 4,096 nodes of Blue Gene/P

Graph Specification: Components Identities

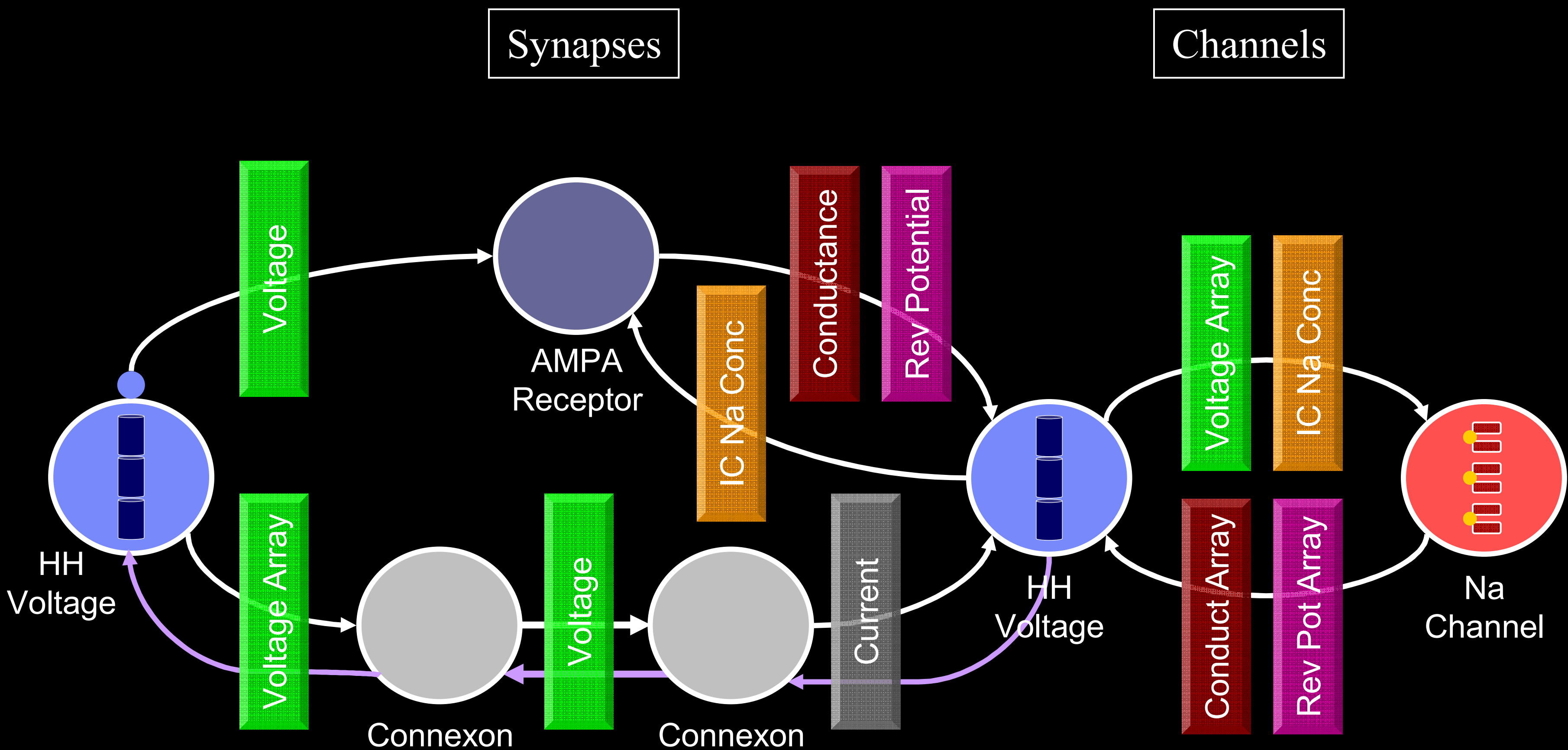
- Each segment describes itself
- Segment key is compressed for efficient communication
- With key, a segment can be identified, and its neuron data accessed



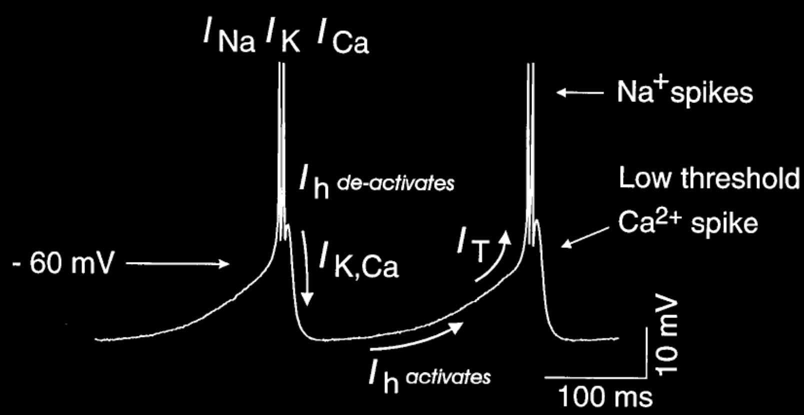
Graph View: Synapses and Channels

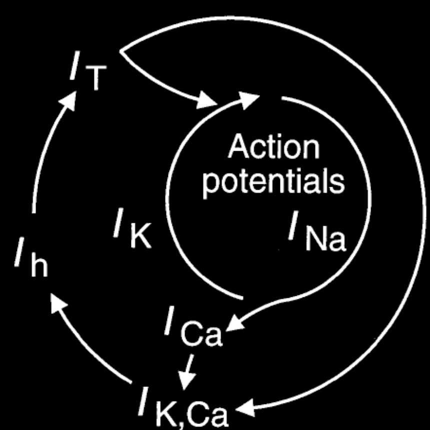


Graph View: Synapses and Channels

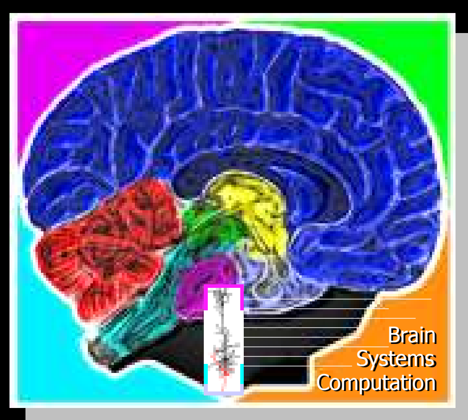


Next Steps: Inferior Olive





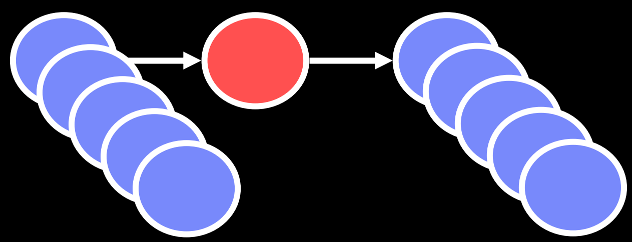
- Clear data constraints available for single cell electrophysiological model
- Oscillations generated by intrinsic interplay between membrane currents
- Subthreshold oscillations are not driven by spike *input*, but instead constrain and drive spike *output*



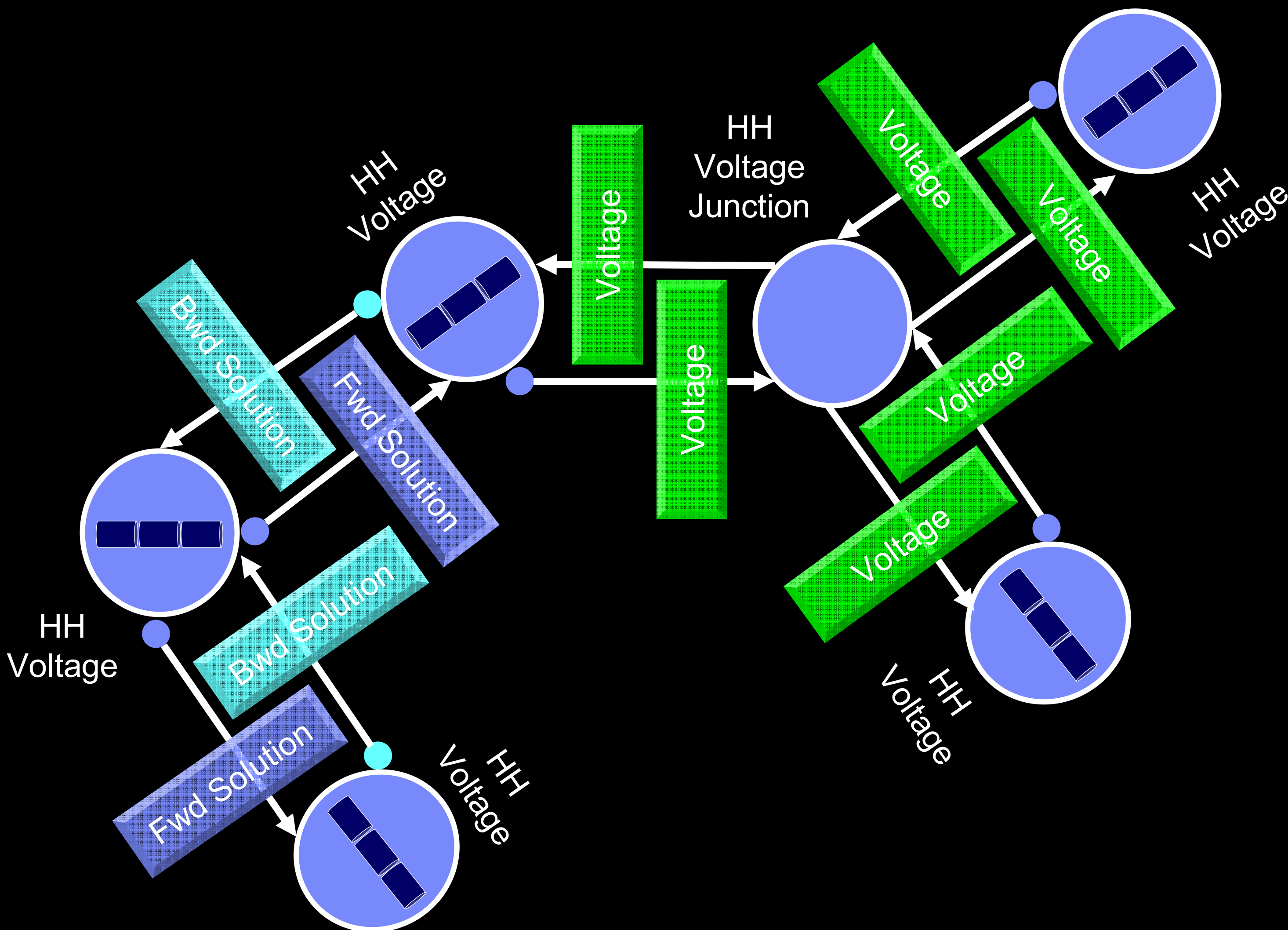
T. Bal and D. McCormick, "Synchronized oscillations in the Inferior Olive are controlled by the hyperpolarization-activated cation current I_h", J. Neurophysiol. 77:3145-3156, 1997.

Modeling Calcium Dynamics in IO neurons

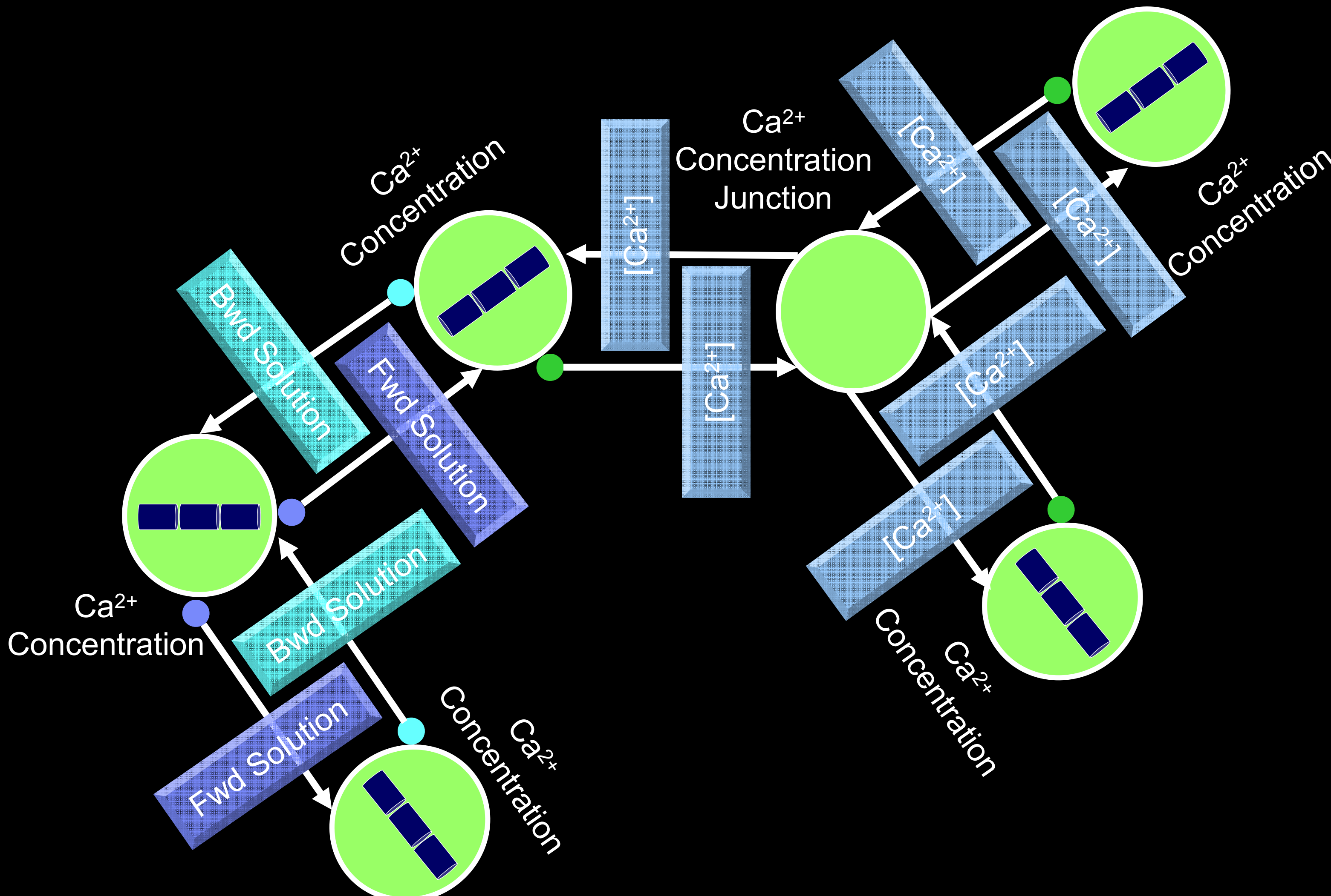
```
InitPhases = { initialize };
RuntimePhases = { run1, run2, run3, run4, run5, run6 };

NodeType HHVoltageJunction { predictState->run1,
                             correctState->run6 };

NodeType HHVoltage { forwardEliminateCO0->run2,
                     forwardEliminateCO1->run3,
                     backSubstituteCO1->run4,
                     backSubstituteCO0->run5 };

NodeType CaConcentration { forwardEliminateCO0->run2,
                           forwardEliminateCO1->run3,
                           backSubstituteCO1->run4,
                           backSubstituteCO0->run5 };
```

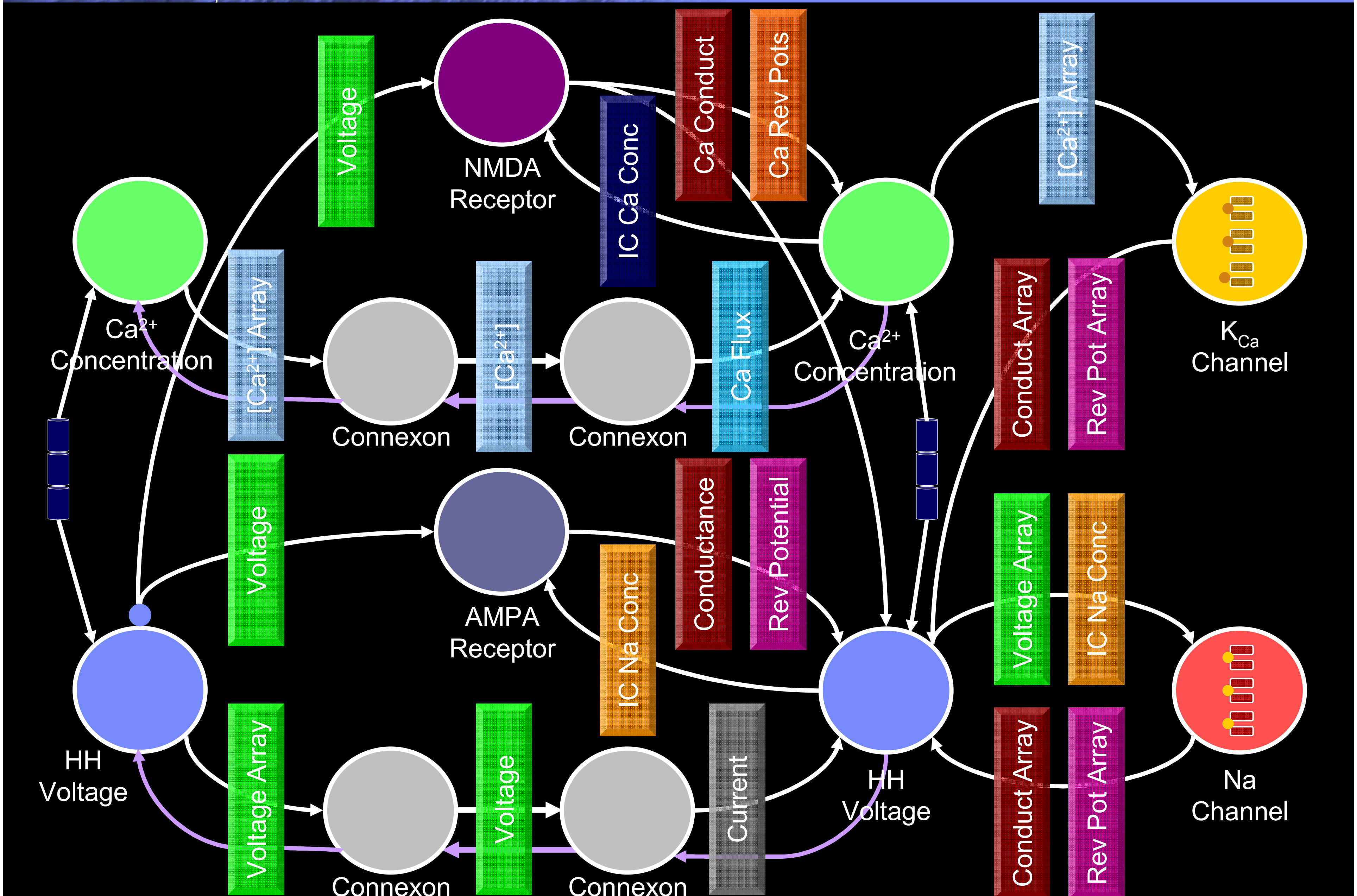


Graph View: Hybrid Voltage Solver



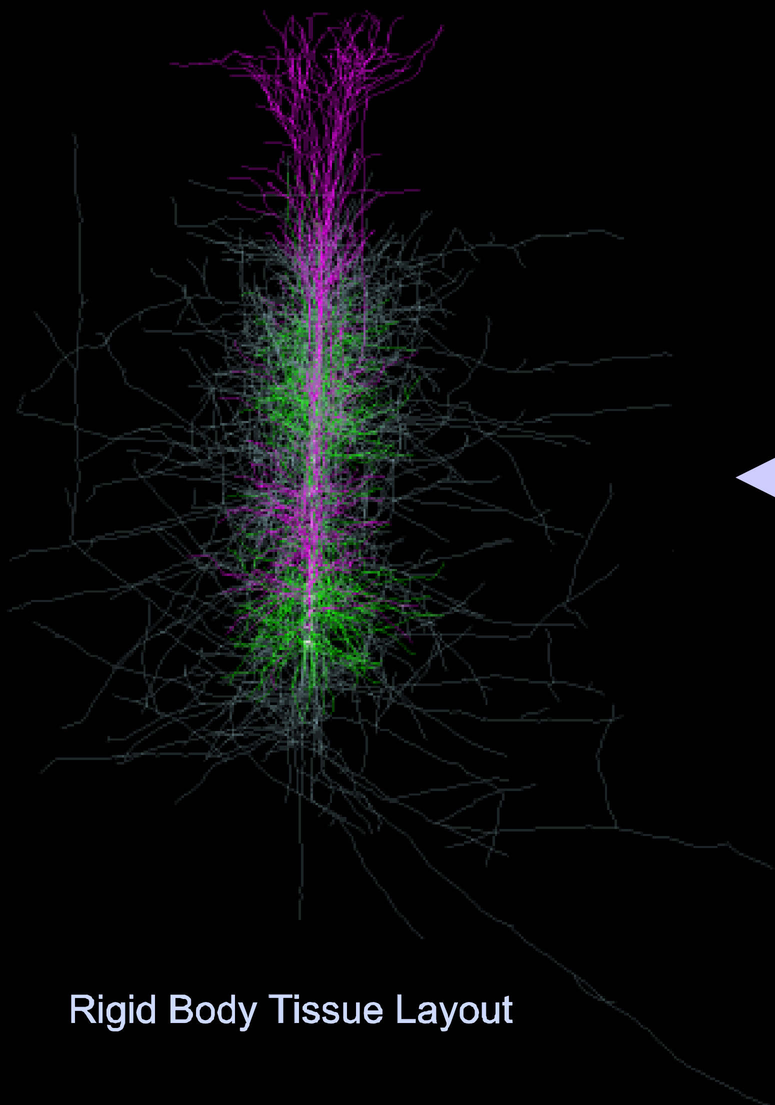
Graph View: Hybrid Calcium Solver



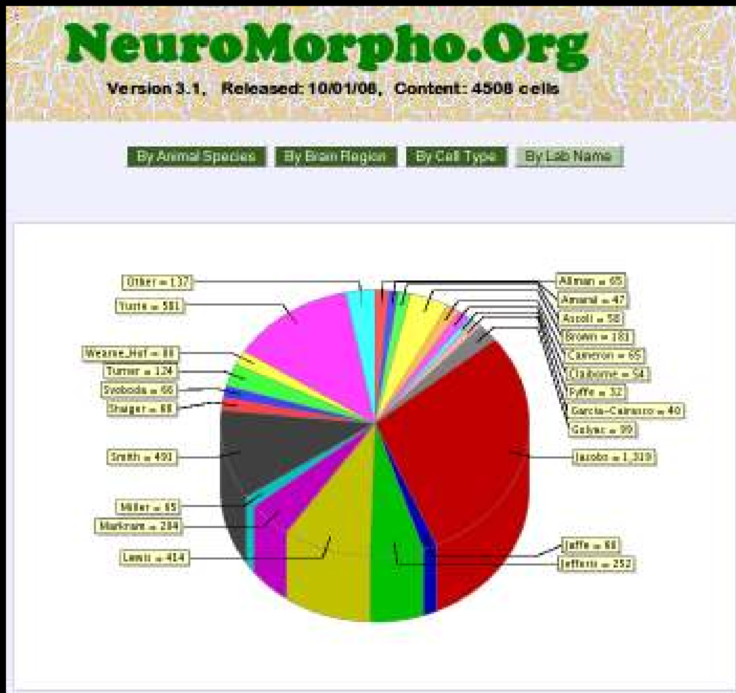


Voltage
NMDA Receptor
IC Ca Conc
Ca Conduct
Ca Rev Pots
[Ca2+] Array
Ca^{2+} Concentration
[Ca2+] Array
Connexon
[Ca2+]
Connexon
Ca Flux
Ca^{2+} Concentration
Conduct Array
Rev Pot Array
K_{Ca} Channel
Voltage
AMPA Receptor
IC Na Conc
Conductance
Rev Potential
Voltage Array
IC Na Conc
HH Voltage
Voltage Array
Connexon
Voltage
Connexon
Current
HH Voltage
Conduct Array
Rev Pot Array
Na Channel

SIMULATED “MINICOLUMN”

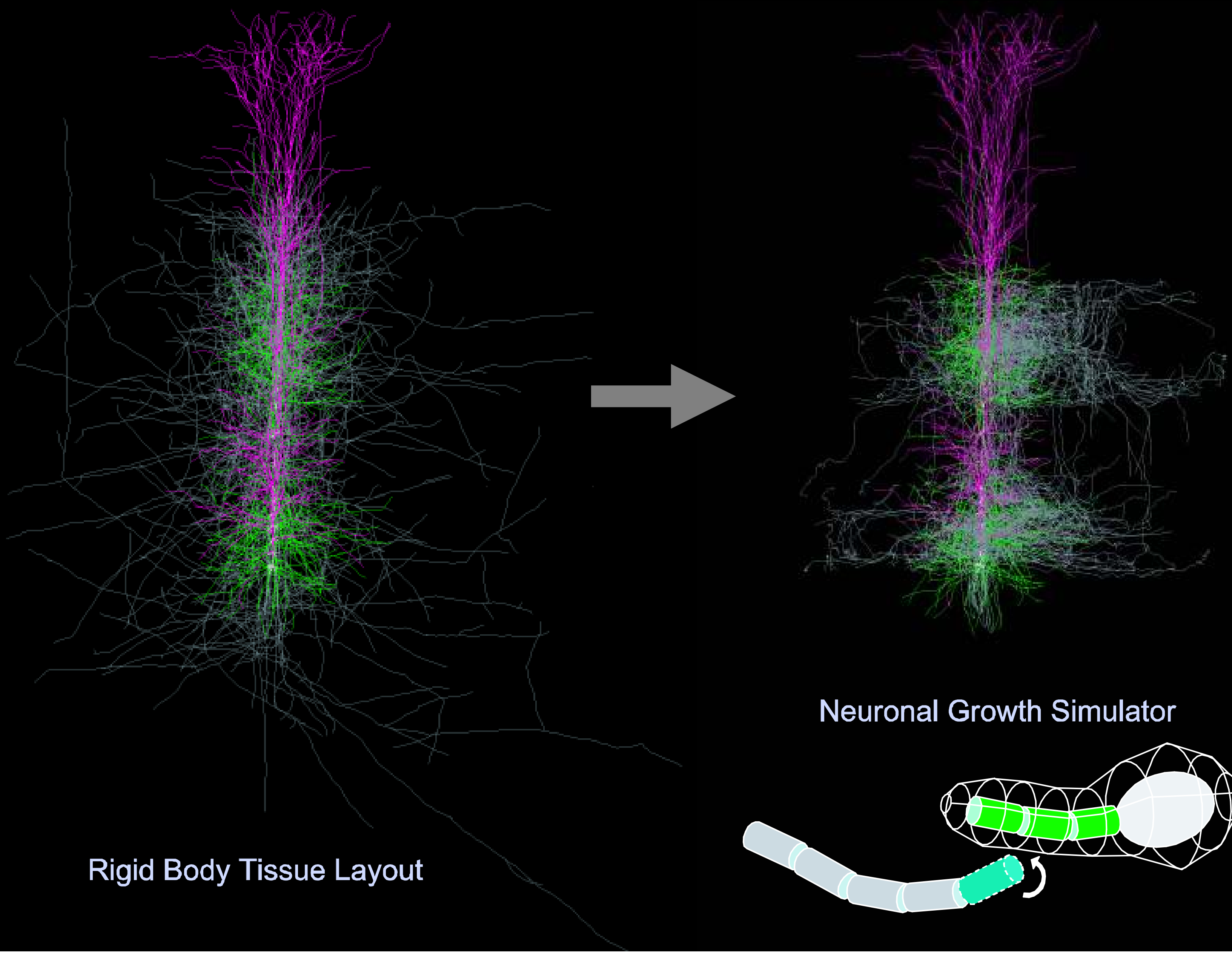


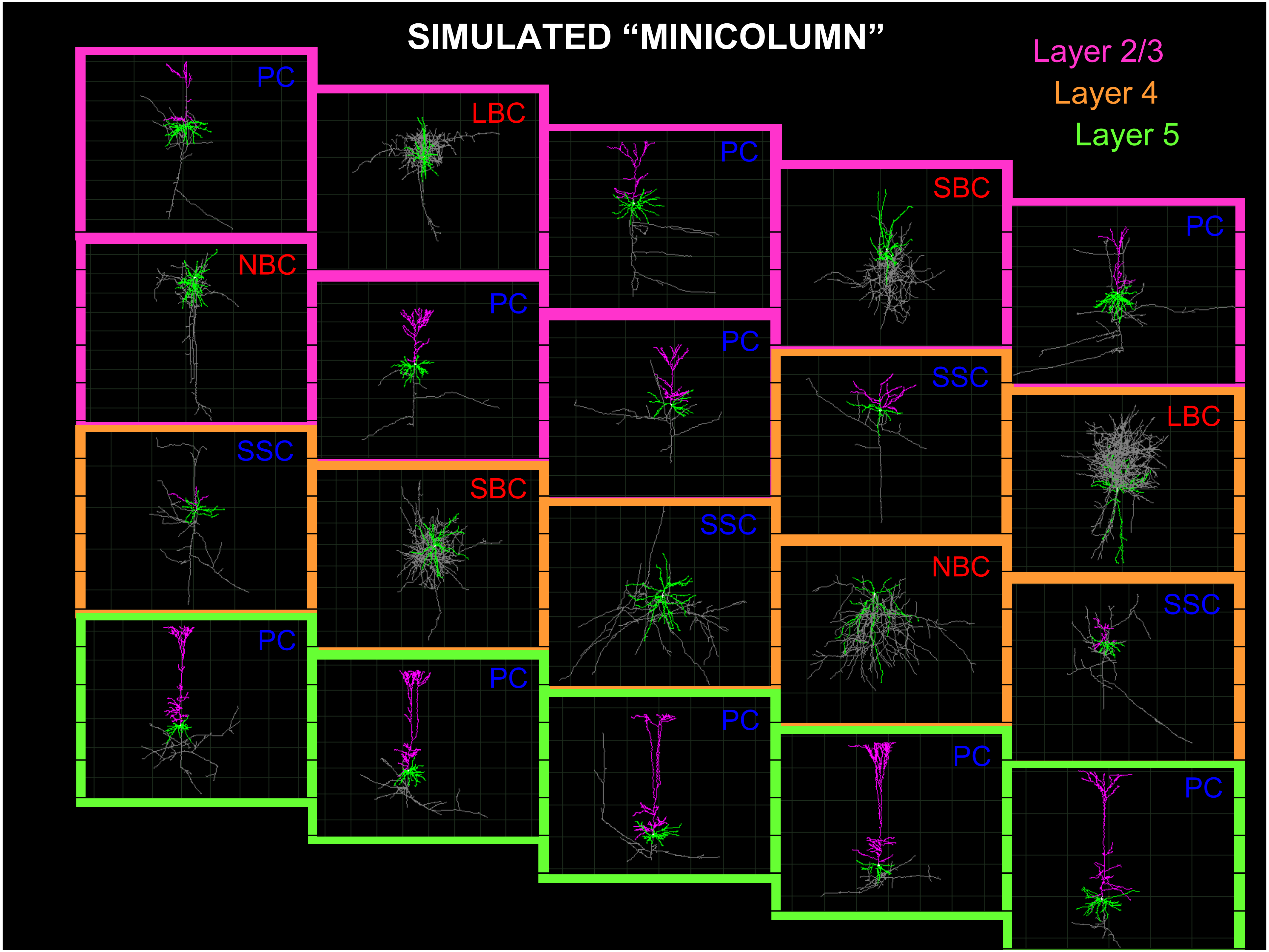
Rigid Body Tissue Layout

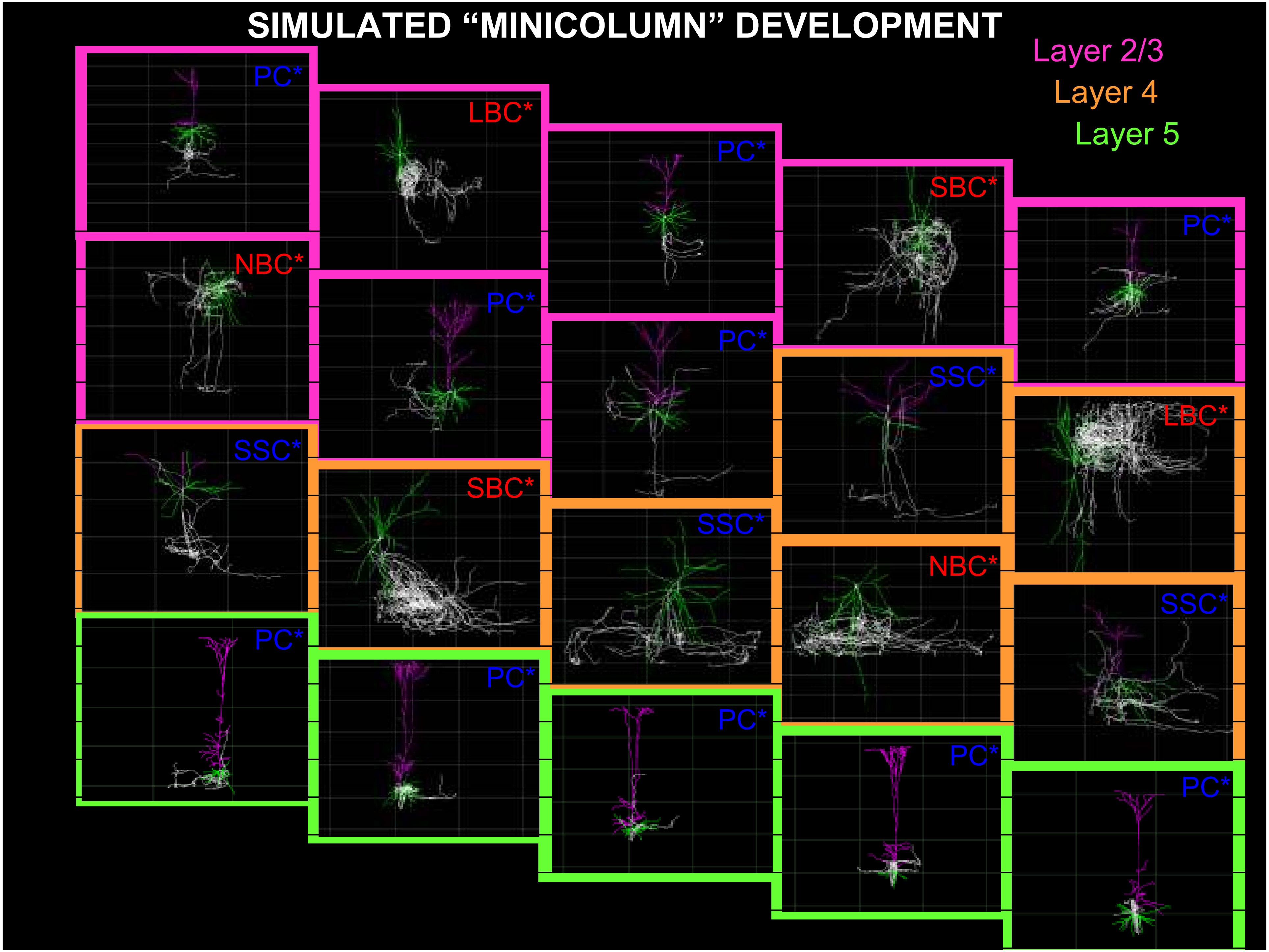


SIMULATED "MINICOLUMN" DEVELOPMENT

In collaboration with Mike Pitman, Protein Science & Molecular Dynamics




SIMULATED "MINICOLUMN"
Layer 2/3
Layer 4
Layer 5
PC
LBC
PC
SBC
PC
NBC
PC
PC
SSC
LBC
SSC
SBC
SSC
NBC
SSC
PC
PC
PC
PC
PC


SIMULATED "MINICOLUMN" DEVELOPMENT
Layer 2/3
Layer 4
Layer 5
PC*
LBC*
PC*
SBC*
PC*
NBC*
PC*
PC*
SSC*
LBC*
SSC*
SBC*
SSC*
NBC*
SSC*
PC*
PC*
PC*
PC*
PC*

Graph Specification: Compartment Variables

COMPARTMENT_VARIABLE_TARGETS 4
BRANCHTYPE
0 Voltage, Calcium
1 Voltage
2 Voltage, Calcium
3 Voltage, Calcium

COMPARTMENT_VARIABLE_COSTS 2
Voltage 1.0
Calcium 0.95

Graph Specification: Channels

CHANNEL_TARGETS 4
BRANCHTYPE
0 Na [Voltage] KDR [Voltage] Cah [Voltage, Calcium] KCa [Voltage, Calcium]
1 Na [Voltage] KDR [Voltage]
2 Cah [Voltage, Calcium] KCa [Voltage, Calcium]
3 Cah [Voltage, Calcium] KCa [Voltage, Calcium]

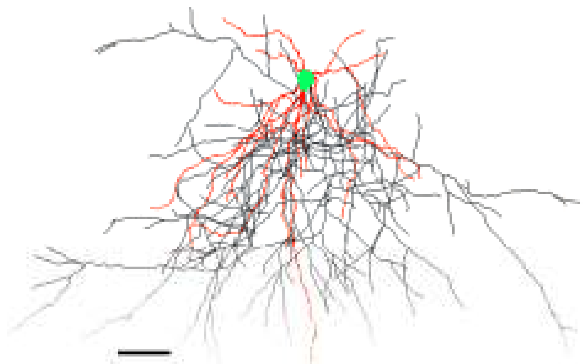
CHANNEL_COSTS 4
Na 0.414243
KDR 0.254051
Cah 0.414243
KCa 0.359252

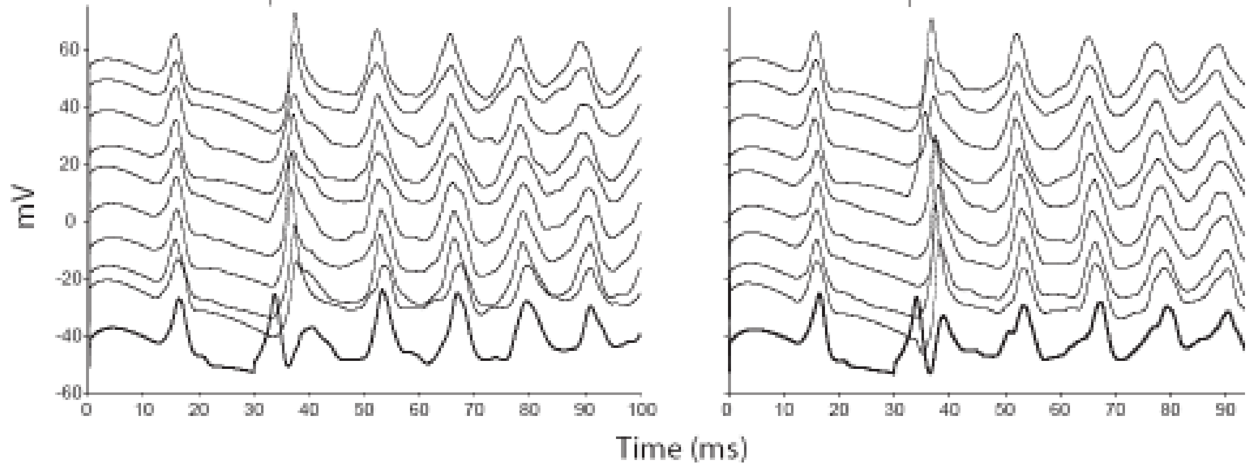
Graph Specification: Synapses

```
ELECTRICAL_SYNAPSE_TARGETS 2
BRANCHTYPE ETYPE
BRANCHTYPE ETYPE
1 0 1 0 AxAxGap [Voltage] 0.001
2 1 2 1 DenDenGap [Voltage] 0.001

ELECTRICAL_SYNAPSE_COSTS 2
AxAxGap 0.005309
DenDenGap 0.005309

CHEMICAL_SYNAPSE_TARGETS 6
BRANCHTYPE ETYPE
BRANCHTYPE ETYPE
1 1 2 0 GABAA [Voltage] [Voltage] 0.1667
1 1 2 1 GABAA [Voltage] [Voltage] 0.1667
1 1 3 0 GABAA [Voltage] [Voltage] 0.1667
1 0 2 0 AMPA [Voltage] [Voltage] 1.0
1 0 2 1 AMPA [Voltage] [Voltage] 1.0
1 0 3 0 AMPA [Voltage] [Voltage] 1.0 NMDA [Voltage] [Voltage, Calcium] 1.0

CHEMICAL_SYNAPSE_COSTS 2
AMPA 0.296407
GABAA 0.149978
```

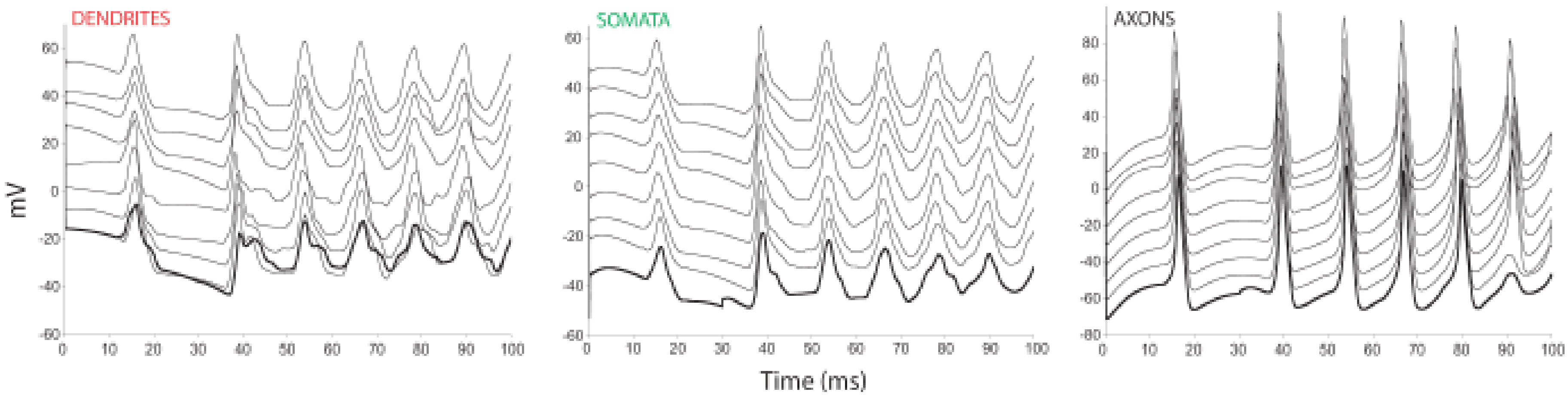
Neurons	1,024,000
AMPA	7,777,821,848
GABAA	2,142,318,617
Connexons	7,244,700
Synapses/Neuron	9,891

Neurons	1,024,000
AMPA	8,186,972,380
GABAA	2,295,068,848
Connexons	7,828,134
Synapses/Neuron	10,301

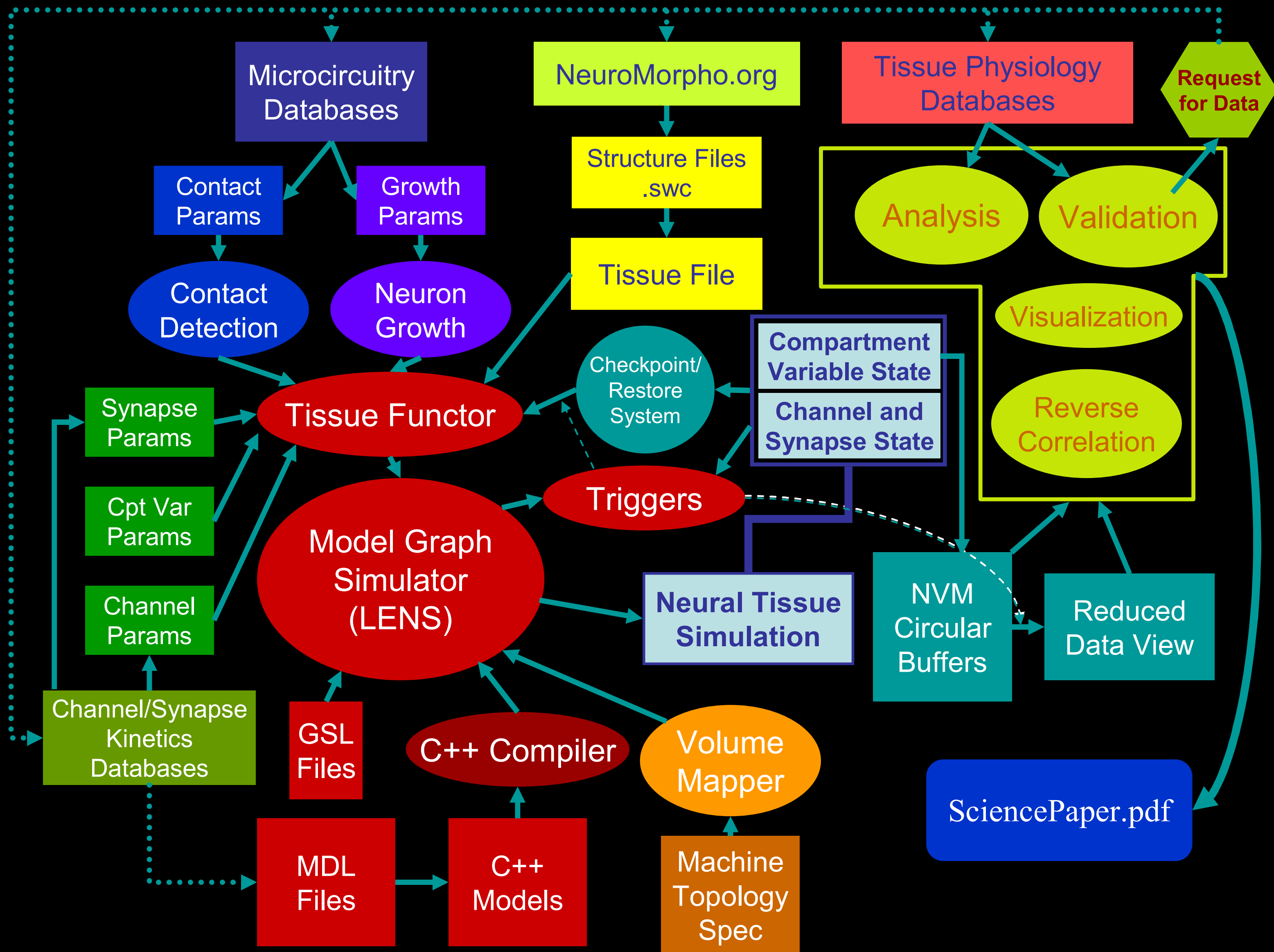




Neurons	16,000
AMPA	115,647,522
GABAA	44,352,919
Connexons	137,338
Synapses/Neuron	10,004

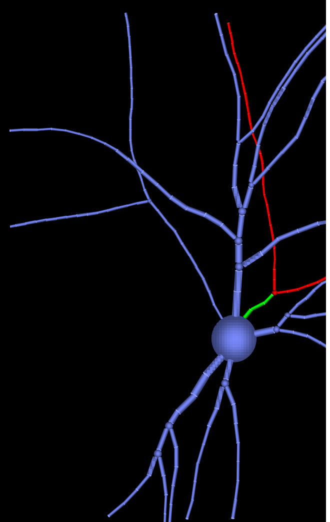


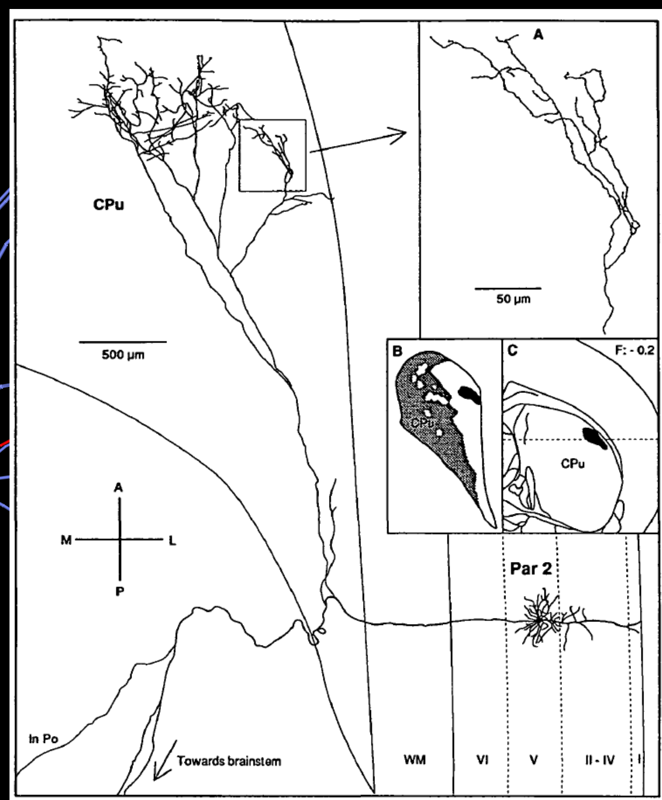
Neural Tissue Simulator Workflow

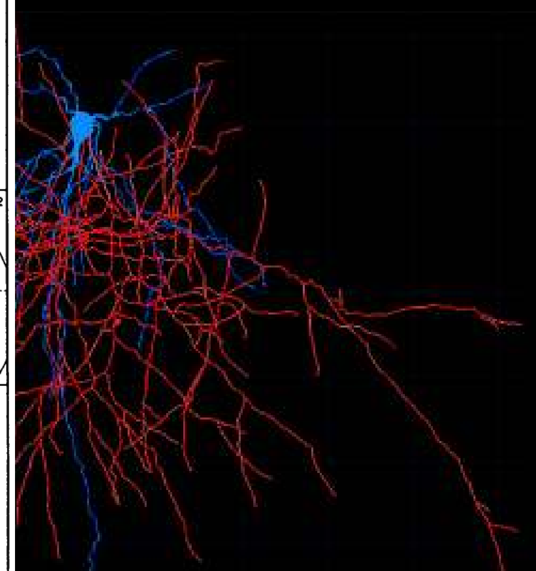


Neural Tissue Simulation: Scaling

The question of axons…



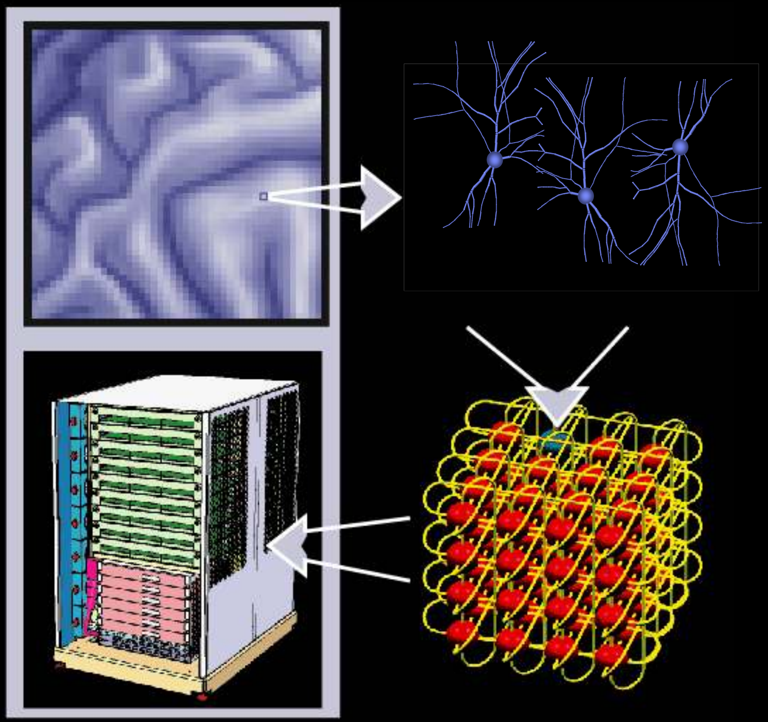




M. Lévesque, S. Gagnon, A. Parent, and M. Deschênes, "Axonal Arborizations of Corticostriatal and Corticothalamic Fibers Arising from the Second Somatosensory Area in the Rat

Neural Tissue Simulation: Scaling

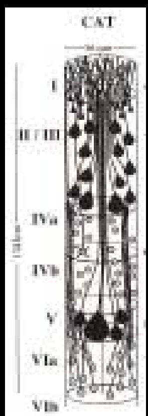
The question of axons…

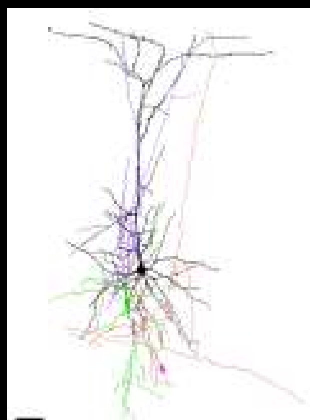


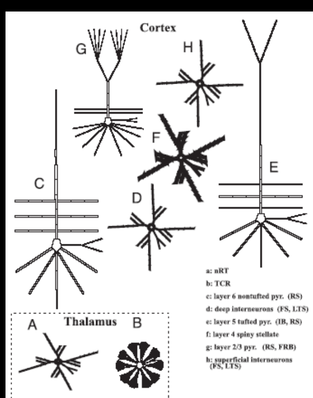
"…processors act like neurons and connections between processors act as axons…"

H. Markram. The Blue Brain Project. *Nat Rev Neurosci*, 7(2):153–160, Feb 2006.

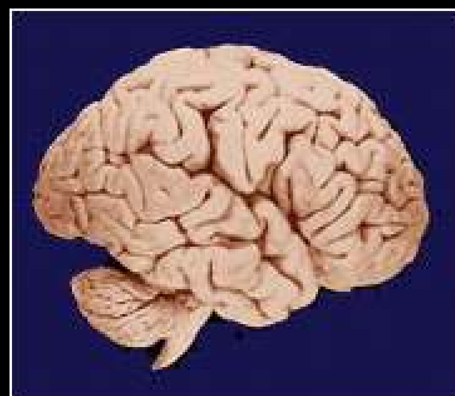
Scalability







Traub et al., *J Neurophysiol* 93: 2194 2232, 2005



≠

Neural Tissue Simulation: Scaling

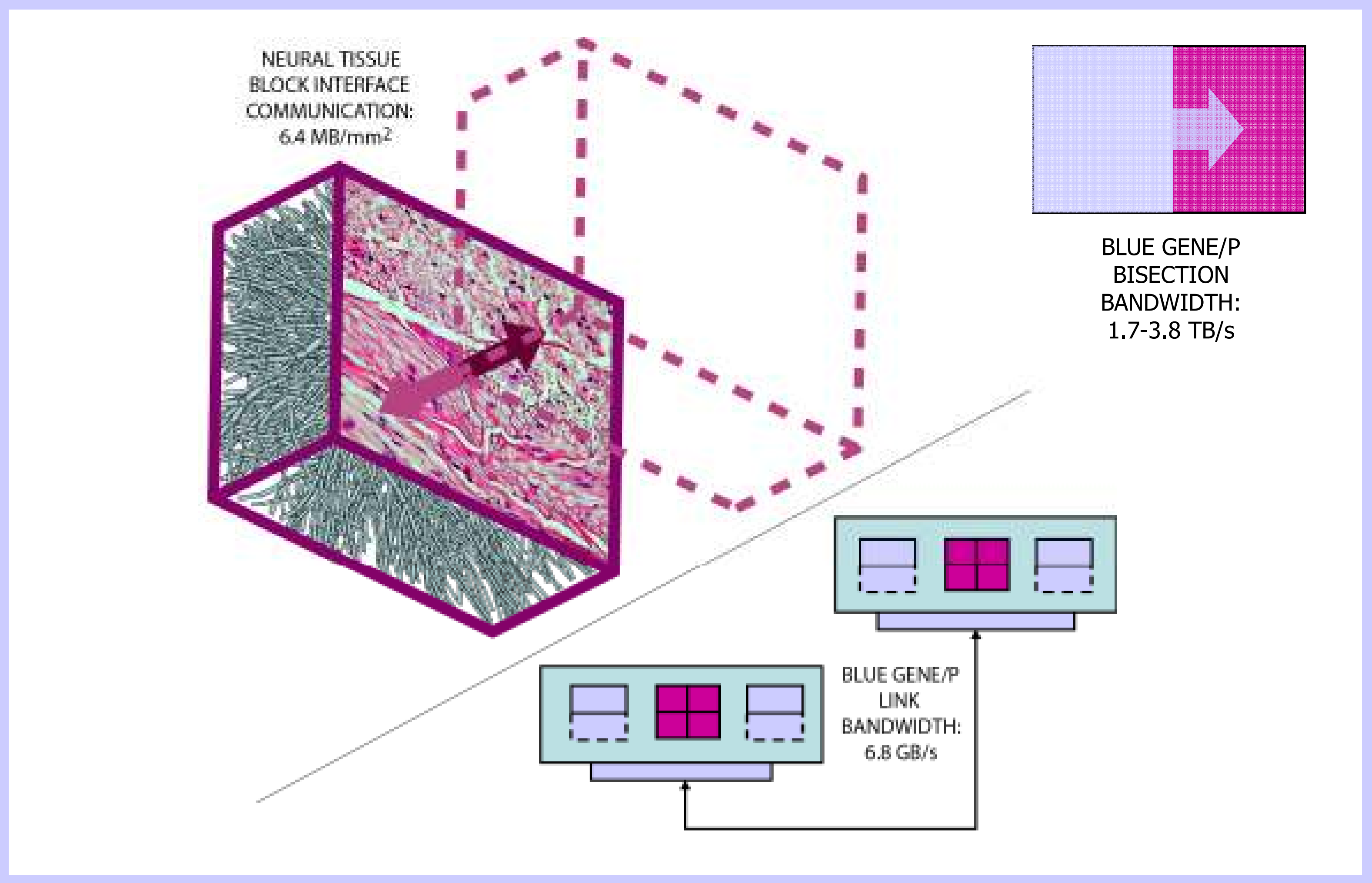
The question of axons…

- Failures of action potential propagation can occur at certain points along an axon, introducing uncertainty surrounding the signaling role of action potentials transmitted through otherwise reliable axons [1]
- Electrical synapses between axons can initiate action potentials without first depolarizing the axon initial segment [2]
- Action potentials may be generated by a mechanism that depends on the length of the axon (e.g., bursts of action potentials of a particular duration may be generated when a calcium spike from the cell body depolarizes an axon of a particular length [1])

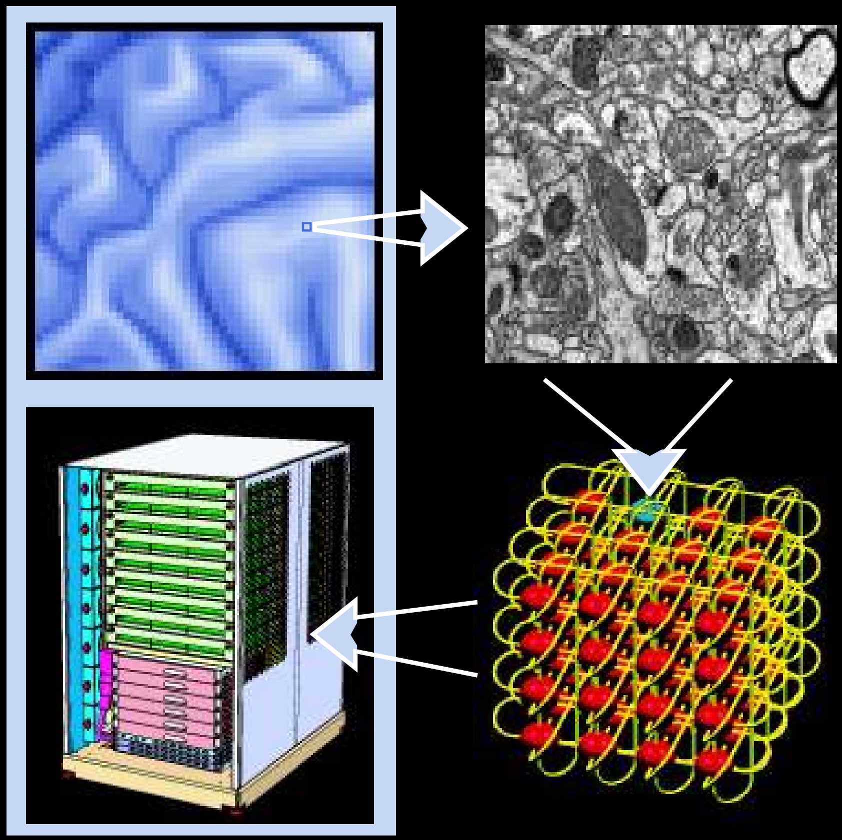
[1] A. Mathy, S. S. N. Ho, J. T. Davie, I. C. Duguid, B. A. Clark, and M. Husser. Encoding of oscillations by axonal bursts in inferior olive neurons. *Neuron*, 62(3):388–399, May 2009.

[2] D. Schmitz, S. Schuchmann, A. Fisahn, A. Draguhn, E. H. Buhl, E. Petrasch-Parwez, R. Dermietzel, U. Heinemann, and R. D. Traub. Axo-axonal coupling. a novel mechanism for ultrafast neuronal communication. *Neuron*, 31(5):831–840, Sep 2001.

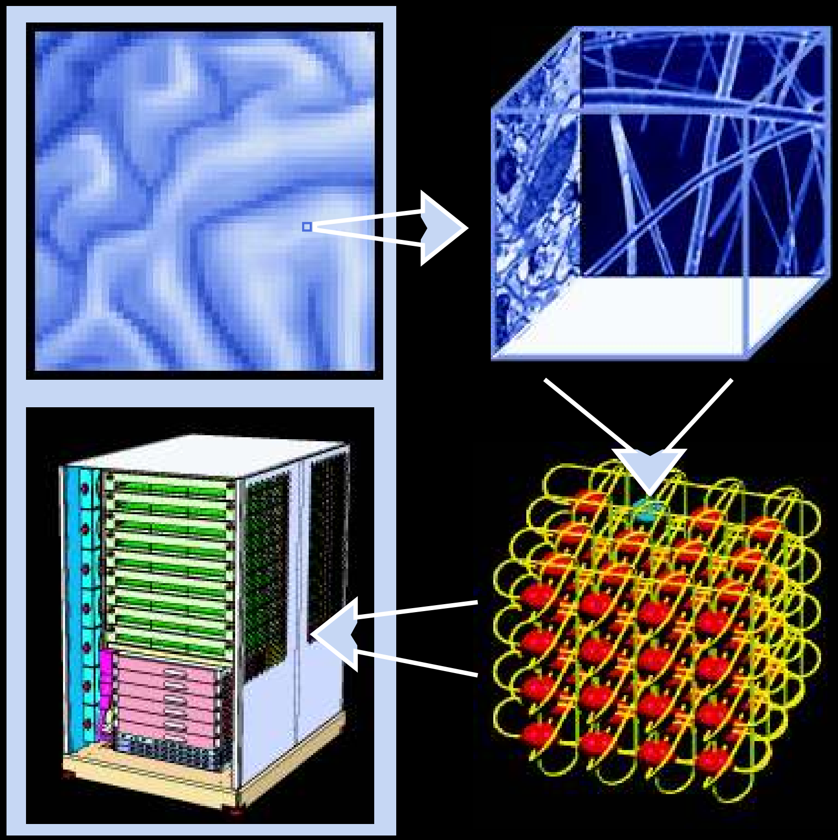
Neural Tissue Simulation: Network Bandwidth



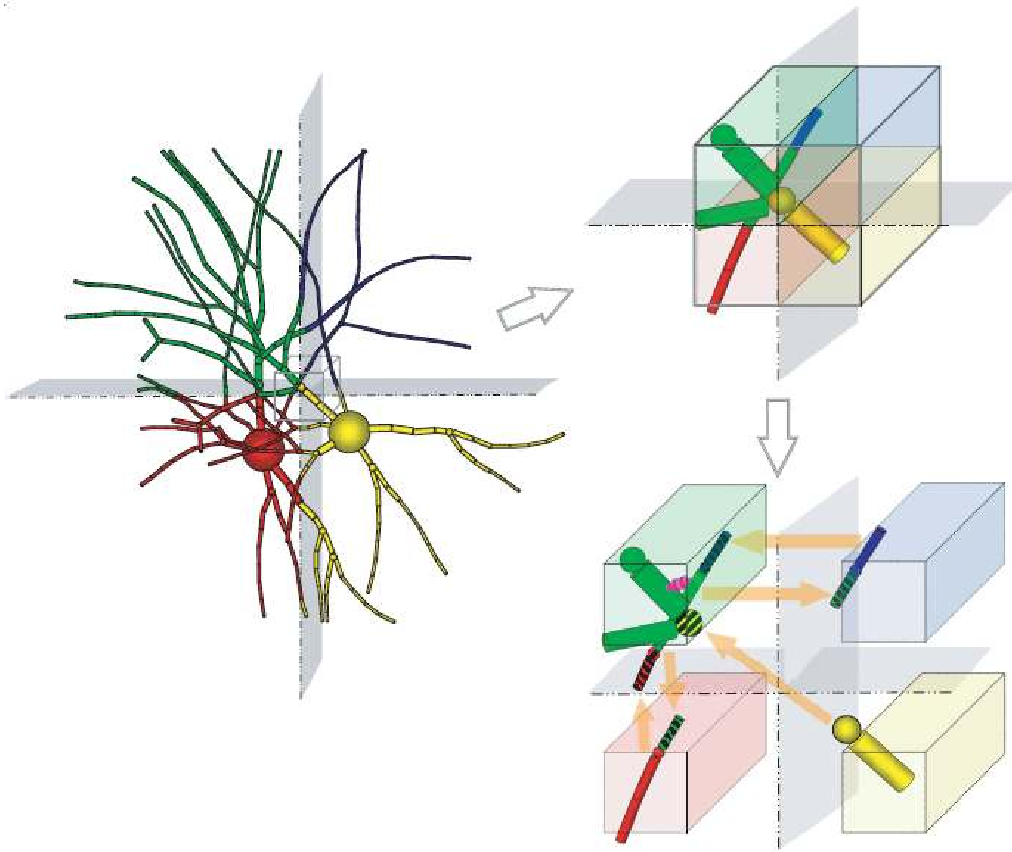
Neural Tissue Simulation



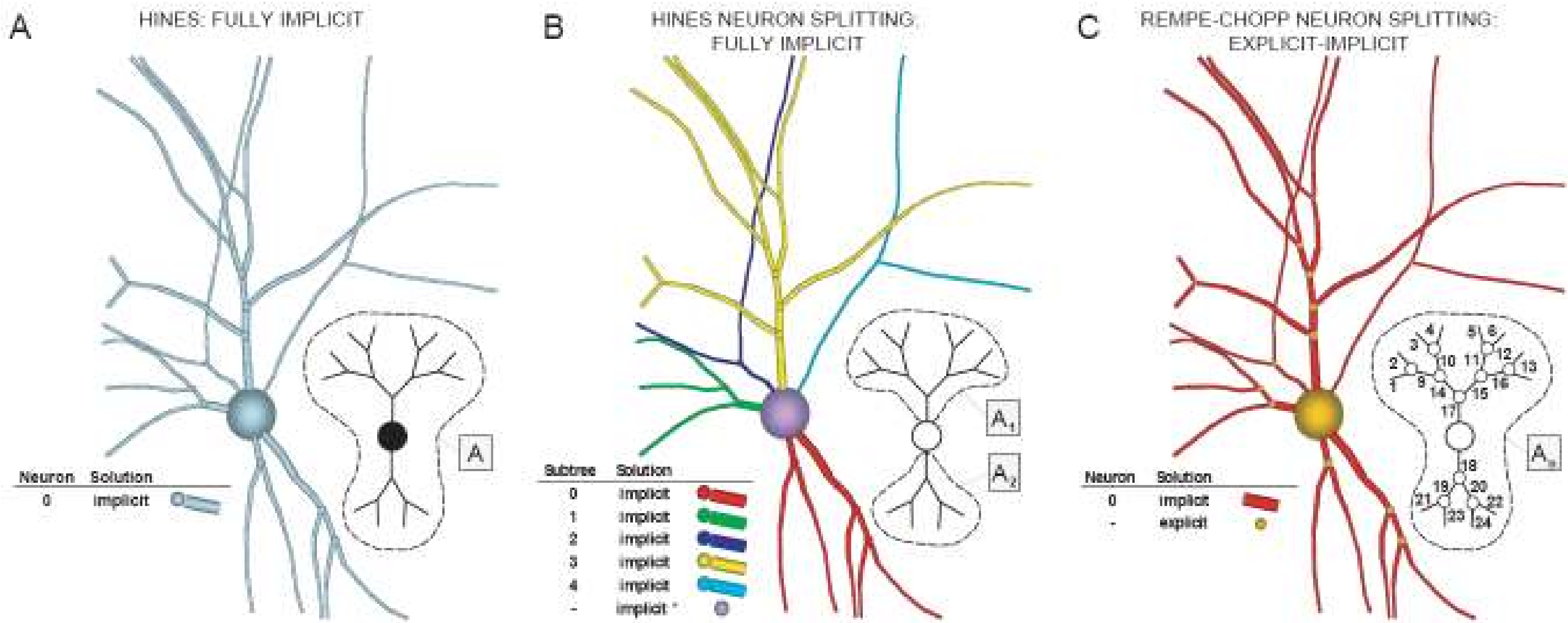
Neural Tissue Simulation



Model Graph Simulator: Tissue Volume Decomposition

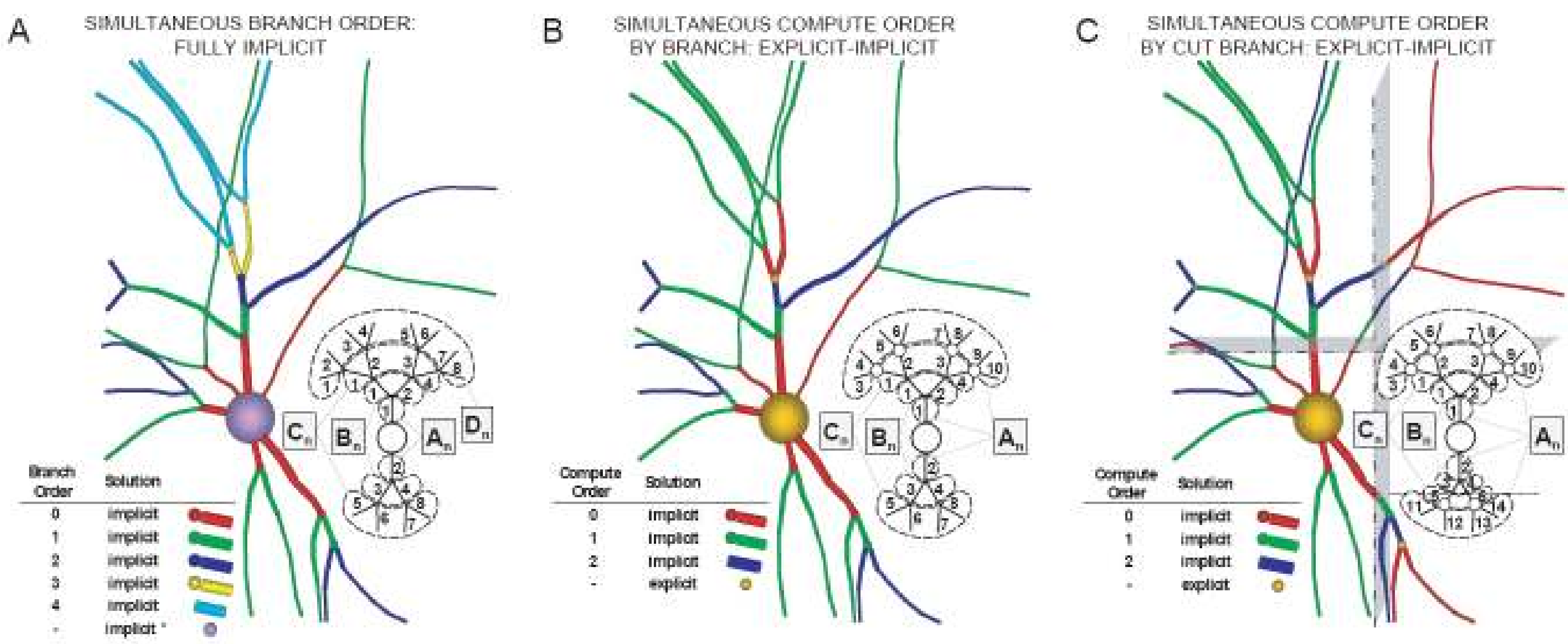


Previous Numerical Approaches

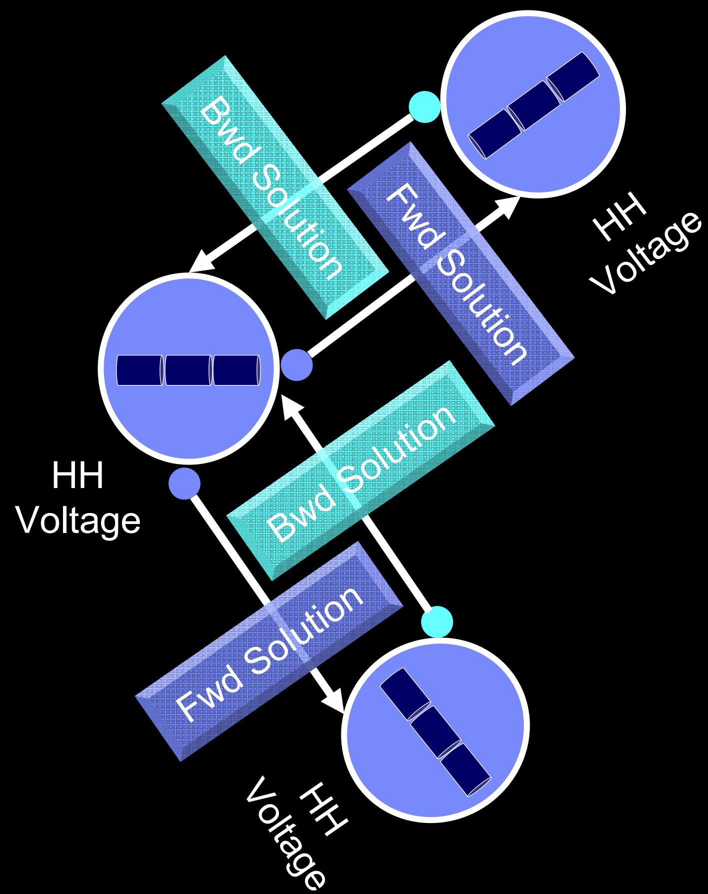


Our Numerical Approach

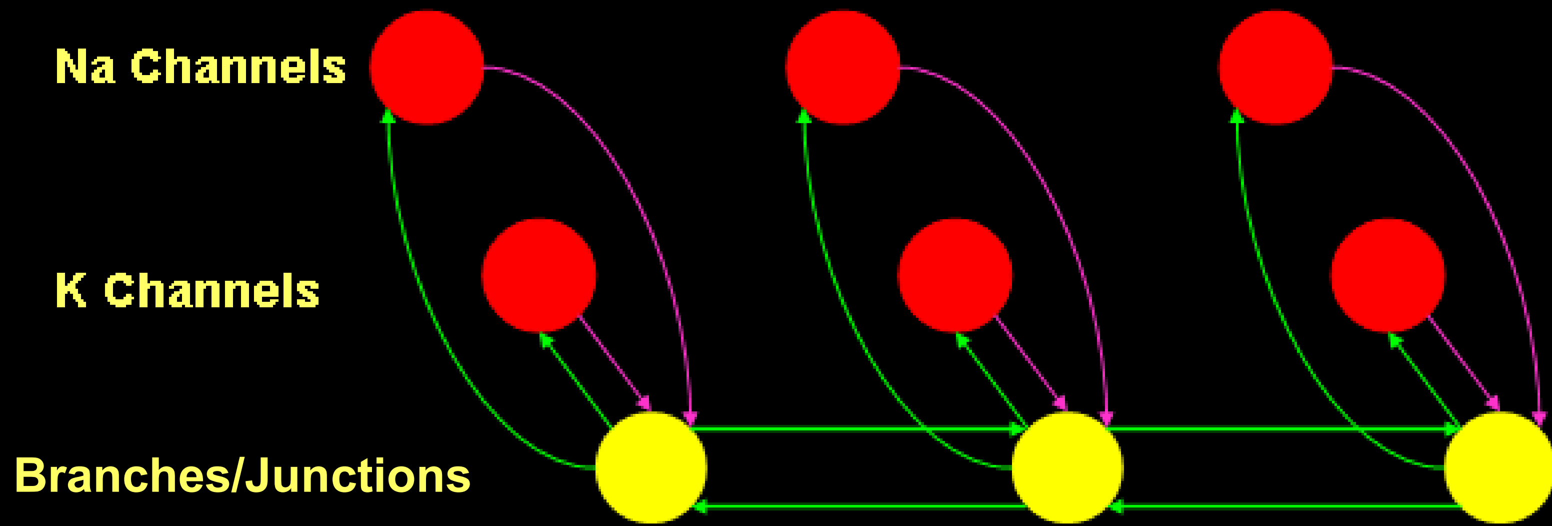
- John Wagner, Manager IBM Research Australia /Computational Biology Co-laboratory



Graph View: Implicit Junction Solver

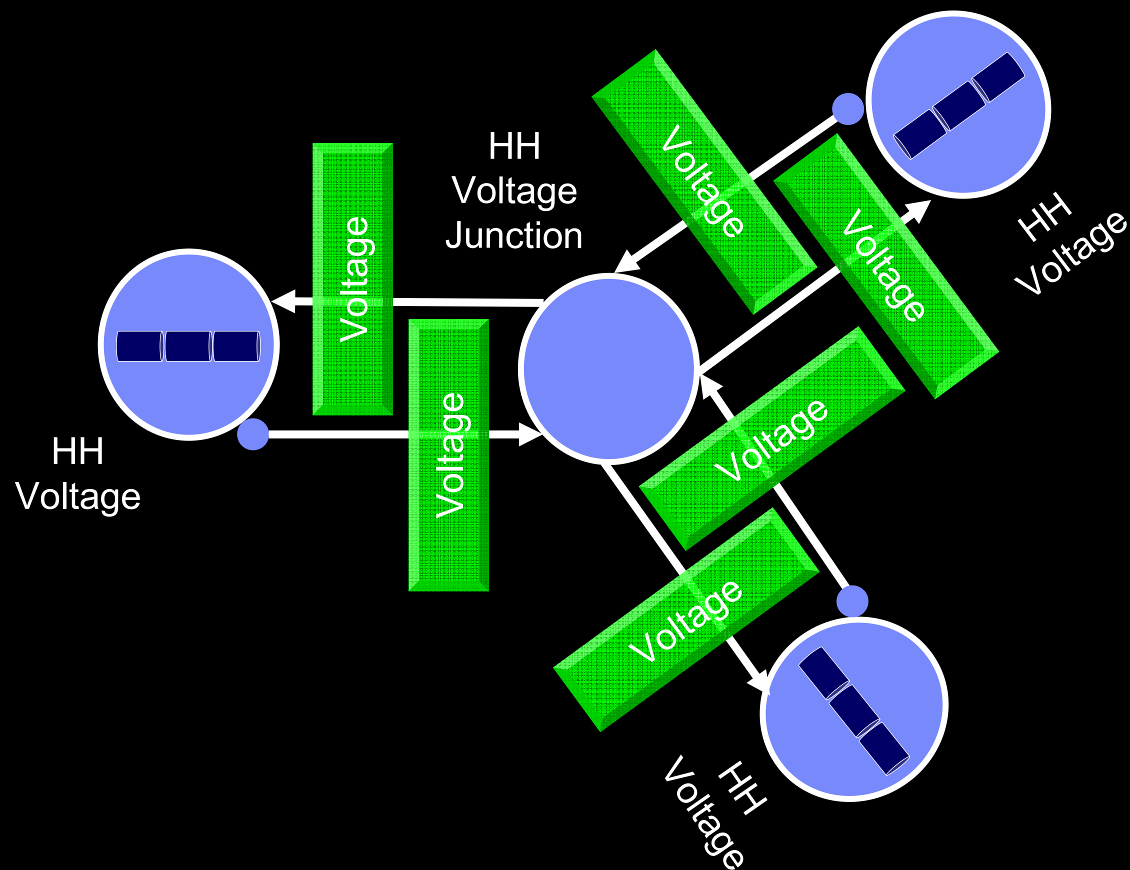


Branch-Based, Implicit Junction Algorithm: Phase Decomposition



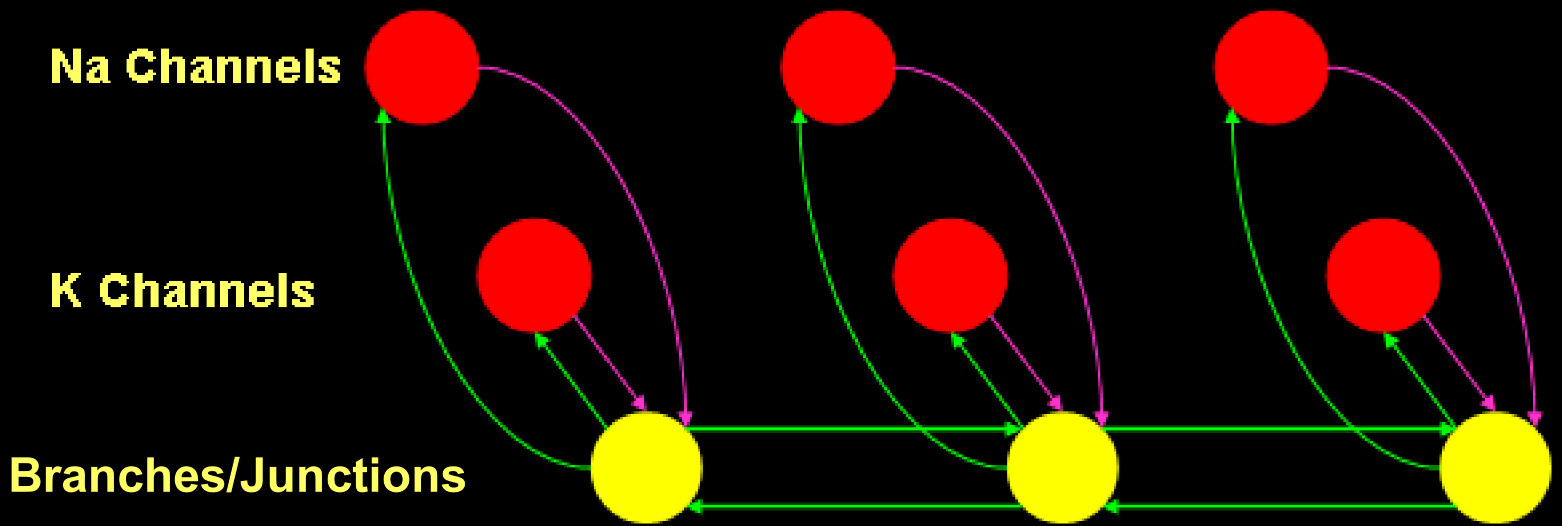
Update I_{flux}	Forward Solve: Branch Order N	Forward Solve: Branch Order 1	Backward Solve: Branch Order 1	Backward Solve: Branch Order N	Update Branch V_m

Graph View: Explicit Junction Solver



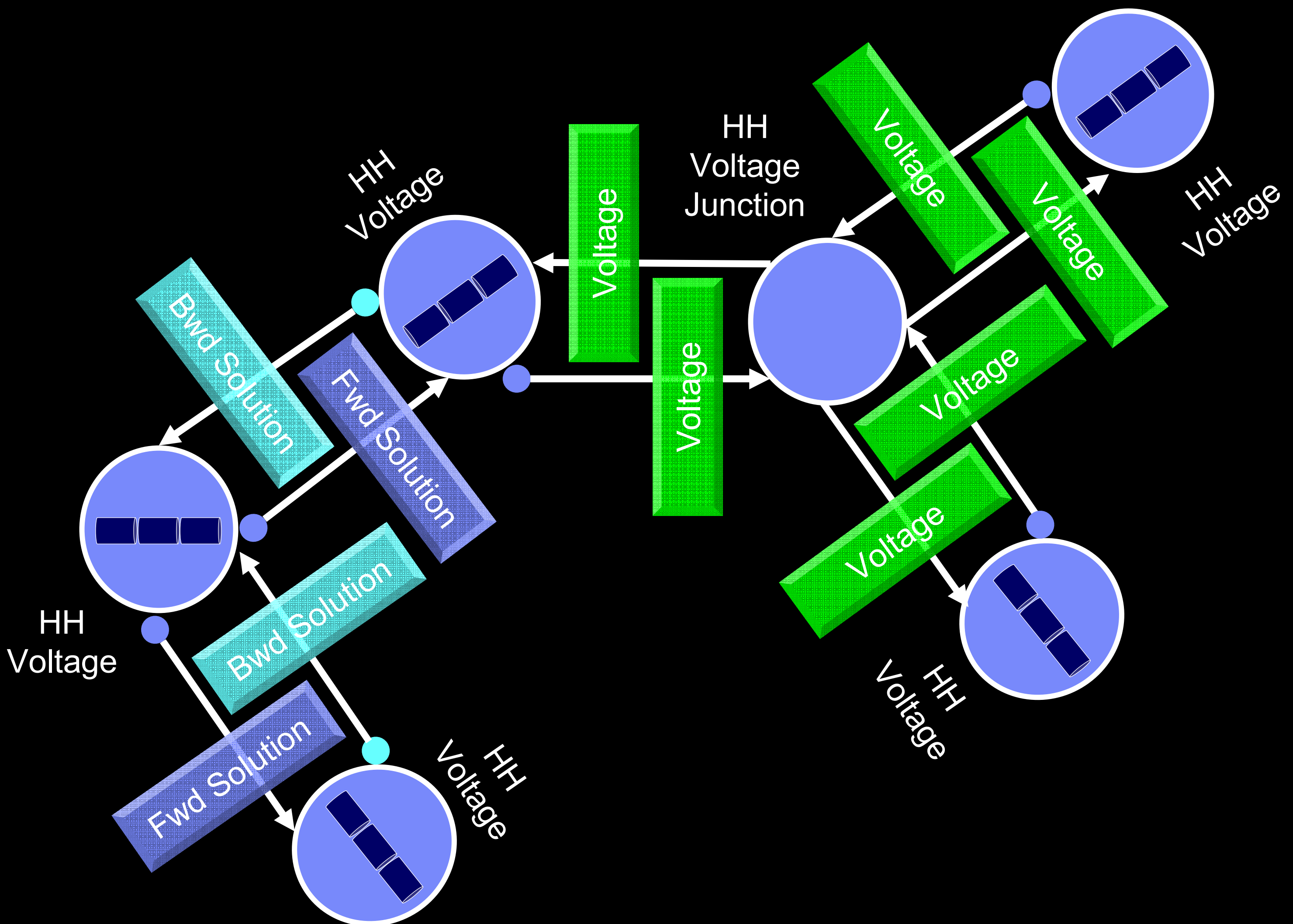
Branch-Based, Explicit Junction Algorithm: Phase Decomposition

Rempe and Chopp, 2006

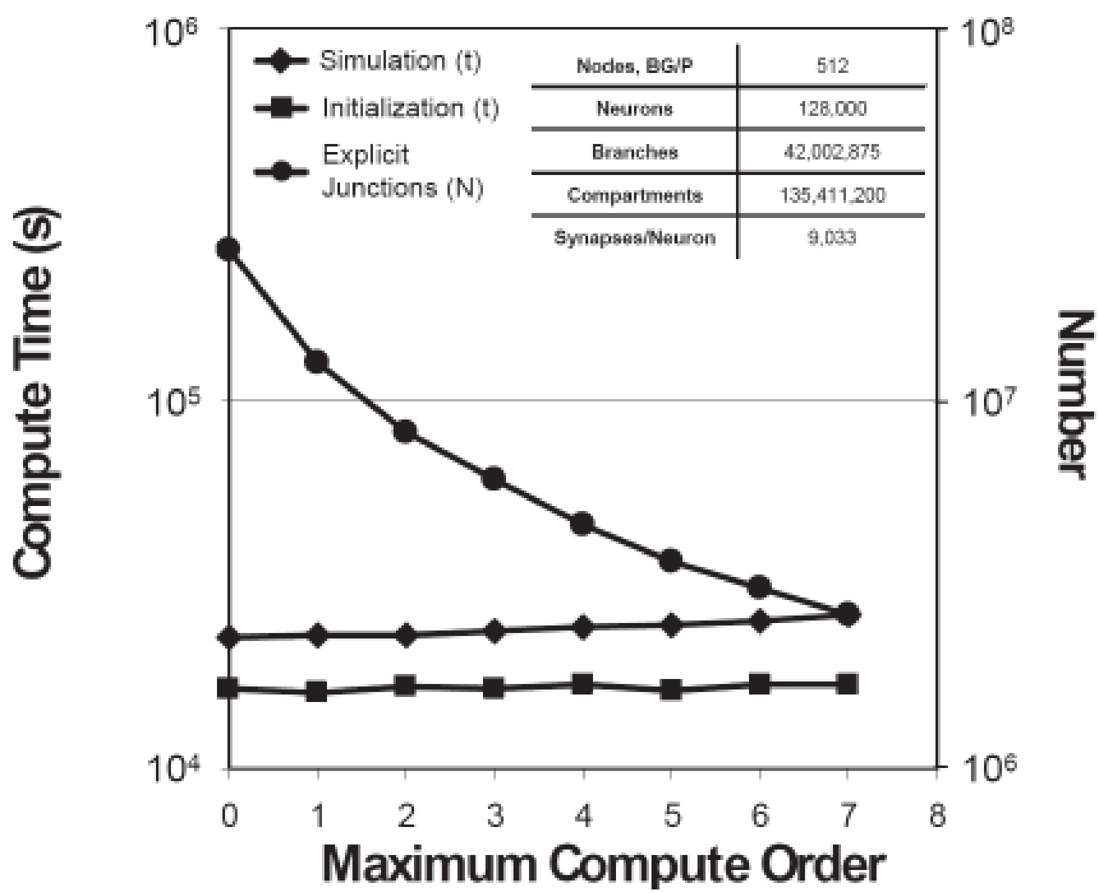


Update I_{flux}	Predict/Correct Junction V_m	Update Branch V_m

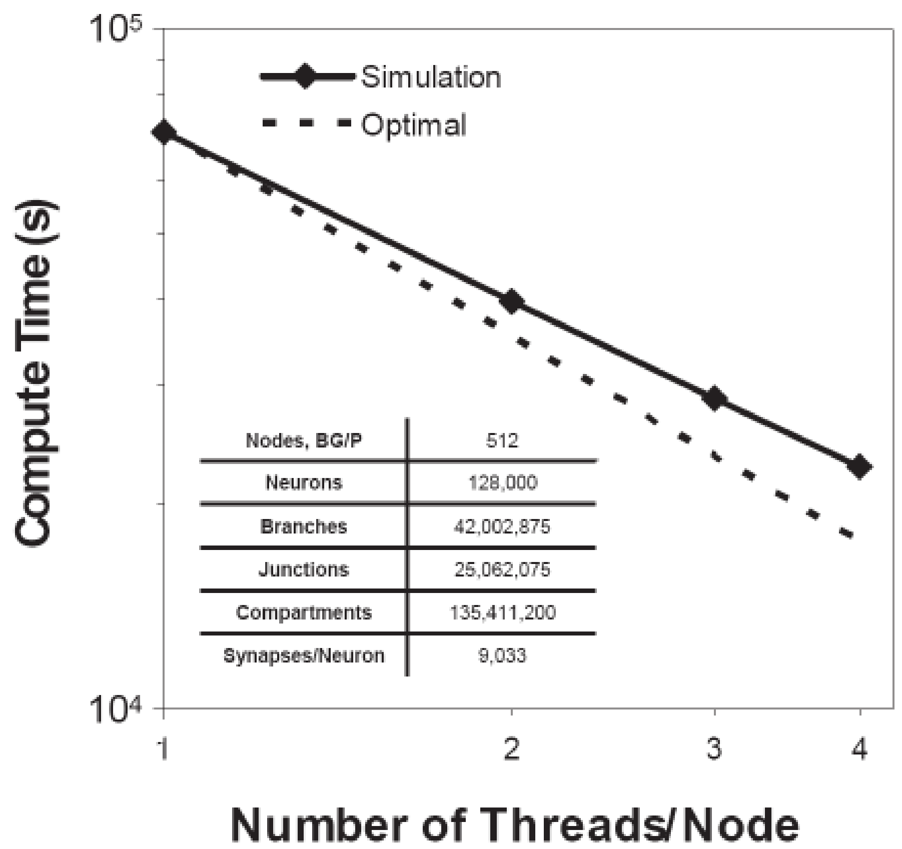
Graph View: Hybrid Solver



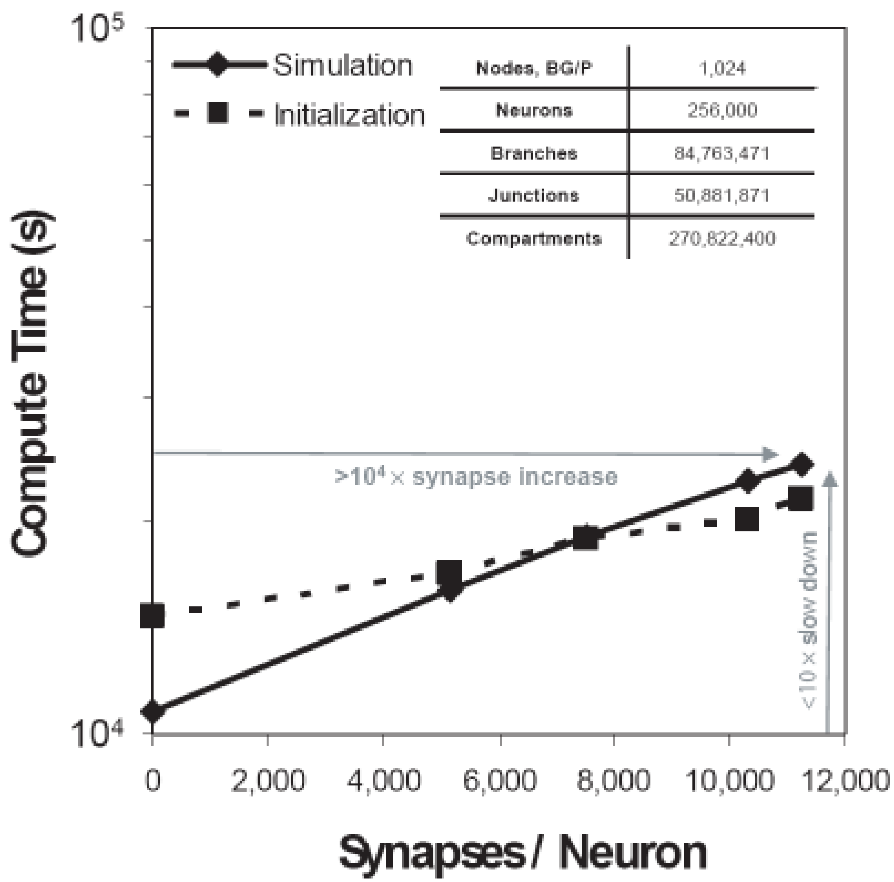
Numerics Scaling



Thread Scaling

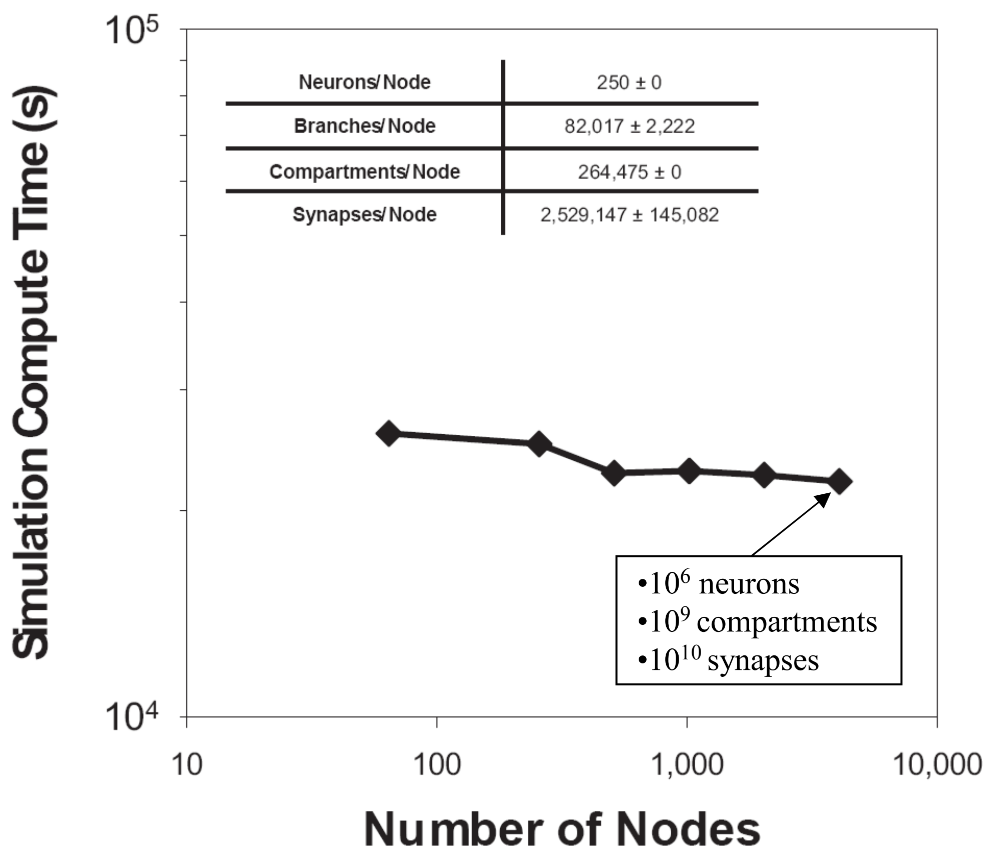


Synapse Scaling



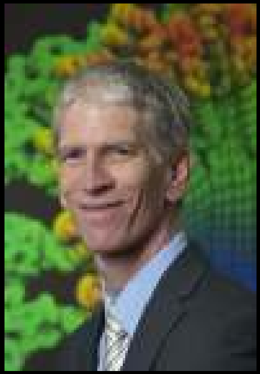
Nodes, BG/P	1,024
Neurons	256,000
Branches	84,763,471
Junctions	50,881,871
Compartments	270,822,400

Weak Scaling



Acknowledgments

- John Wagner
  IBM Research Collaboratory for Life Sciences-Melbourne
  Victorian Life Sciences Computation Initiative
  The University of Melbourne



- Charles Peck
  Manager, Biometaphorical Computing Research
  Research Staff Member, Computational Biology