# BMC Neuroscience

Poster presentation

# Multiscale modeling and interoperability in MOOSE

## Niraj Dudani, Subhasis Ray, Siji George and Upinder S Bhalla*

Address: National Centre for Biological Sciences, Tata Institute of Fundamental Research, Bangalore, India

Email: Upinder S Bhalla* - bhalla@ncbs.res.in

* Corresponding author

This abstract is available from: http://www.biomedcentral.com/1471-2202/10/S1/P54

## Introduction

MOOSE (Multiscale Object Oriented Simulation Environment; http://moose.ncbs.res.in) is a general biological simulator. It allows the simulation of models that span many levels of detail, from single molecules to neuronal networks. MOOSE is backward compatible with GENESIS [1] and has a similar set of objects of various types (channels, molecules, compartments) to represent biological concepts. We report current developments that combine to give MOOSE distinctive capabilities in multiscale modeling and interoperability.

## Multiscale modeling and scalability
### Specialized Numerical Engines

We have implemented Hines' solver for biophysical models of neurons and a GSL- (GNU Scientific Library-) based solver for kinetics of biochemical signaling networks. In terms of speed of execution, the Hines' solver matches the performance of GENESIS, whereas the kinetics' solver is 10×–50× faster than GENESIS. A Gillespie solver for stochastic chemical kinetics has been implemented.

### SigNeur

SigNeur is a set of tools to specify multiscale neuronal models comprising of biochemical signaling and cellular biophysics. The user embeds reaction networks in different parts of a cell, and specifies an appropriate solver for each part of the model (e.g., a stochastic chemical solver for small volumes). Communication is managed seamlessly between solvers. A specific example is upregulation of AMPA receptors in a spine, which is conveyed from a chemical solver to a neuronal solver, and interpreted there as an increase in maximal synaptic conductance.

### Parallelization

A developer- and user-friendly framework has been implemented using the MPI (Message Passing Interface) standard to allow MOOSE to run in parallel on a cluster. Both developers and users can think of their problem in a serial fashion, with the underlying framework managing cross-node communication and load-balancing. MOOSE has successfully cleared a large set of unit tests on a 64-node Linux based Opteron cluster.

## Interoperability
### PyMOOSE

PyMOOSE is a Python-based scripting interface to MOOSE implemented using SWIG (Scalable Wrapper Interface Generator). This lets us use Python to mediate communication among Python-aware simulators. For example [2], we have simulated a single neuron in NEURON [3], and an intracellular reaction network in MOOSE. Simulation state variables were communicated at runtime between the two simulators using Python.

### MUSIC

MUlti SImulator Coordinator (MUSIC) [4] is a standard to allow multiple simulators to exchange data during a simulation. Each simulator is allowed to be a parallelized application. MUSIC compatibility was added to MOOSE and the NEST [5] simulators by their respective developers. A simulation was carried out with a cortical network

model running in NEST sending spike-times via MUSIC to a striatal network model running in MOOSE.

### SBML

The capability to read and write models of chemical signaling in the simulator-independent Systems Biology Markup Language (SBML) is currently being implemented and tested in MOOSE.

## Acknowledgements

## References

1. Wilson MA, Bhalla US, Bower JM: **GENESIS: A system for simulating neural networks.** *Adv Neural Inf Process Syst* 1989, **1**:485-492.
2. Ray S, Bhalla US: **PyMOOSE: interoperable scripting in Python for MOOSE.** *Front Neuroinform* 2008, **2**:6.
3. Hines ML, Carnevale NT: **The NEURON simulation environment.** *Neural Comput* 1997, **9**:1179-1209.
4. Ekeberg Ö, Djurfeldt M: **MUSIC – multisimulation coordinator: Request for comments.** *Available from Nature Precedings* 2008 [http://dx.doi.org/10.1038/npre.2008.1830.1].
5. Gewaltig M-O, Diesmann M: **NEST (Neural Simulation Tool).** *Scholarpedia* 2007, **2**:1430.