841.11 / ZZ88 Performance of Parallel Neuronal Models on Triton Cluster

Anita Bandrowski, Prithvi Sundararaman, Subhashini Sivagnanam, Kenneth Yoshimoto, Amitava Majumdar and Maryann Martone

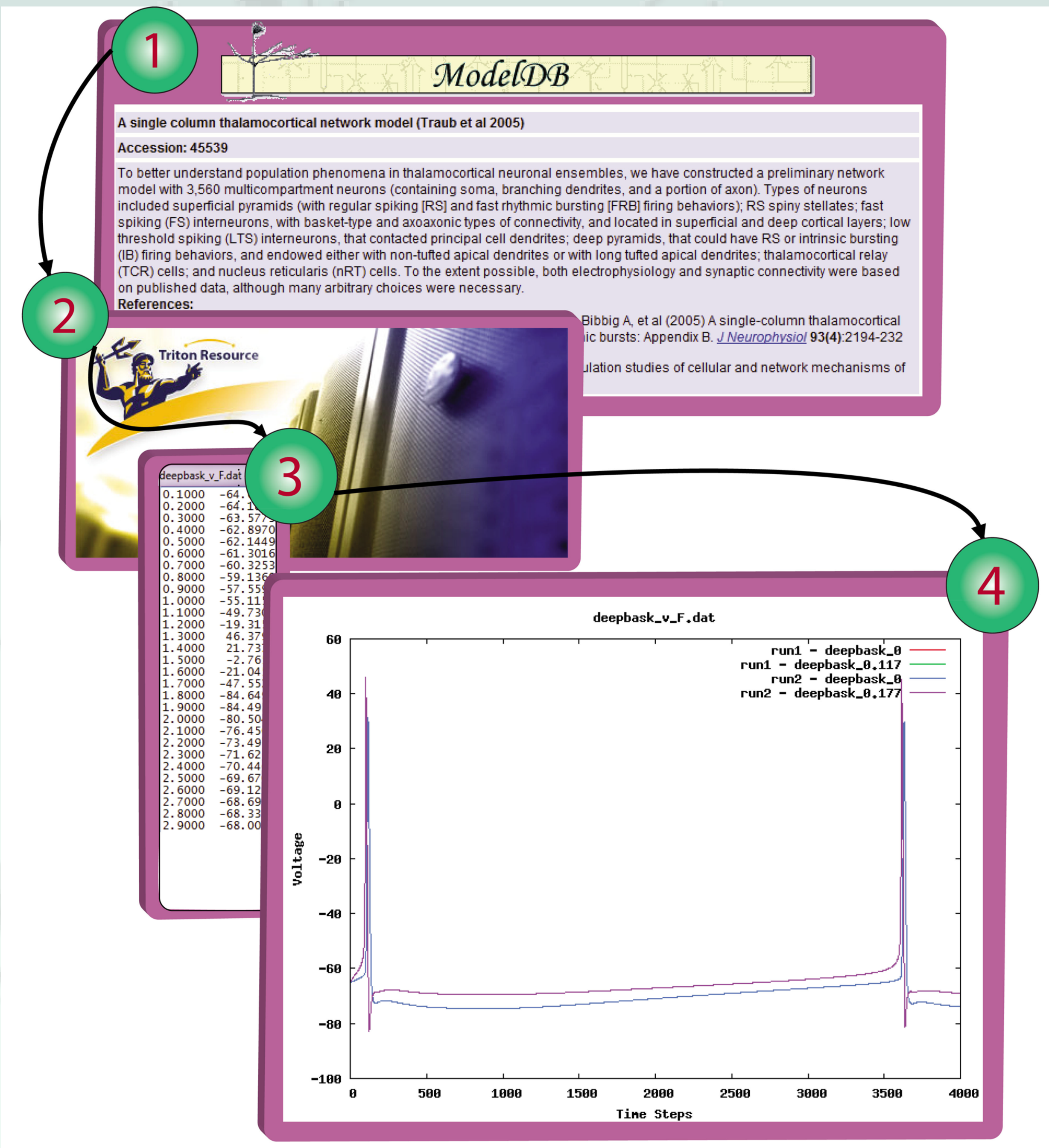
University of California , San Diego, CA

Introduction and Methodology

Large computational clusters can be very useful in quickly simulating large neuronal networks, however building a cluster can be a cost prohibitive and daunting task for an individual laboratory. Research staff at UCSD's Neuroscience department along with members of UCSD's San Diego Super Computer Center (SDSC) are exploring how models built for clusters can run on large machines and what scales are appropriate, with two main goals: to check the validity of data coming from large machines compared to single core machines and to determine performance metrics.

Neuronal models obtained from ModelDB (http://senselab.med.yale.edu/senselab/modeldb/) or neuroConstruct (http://www.neuroconstruct.org/) were run on the Triton compute cluster (256 nodes, 24 gigabytes of memory, Xeon E5530 processors with 8 megabyte cache) at SDSC. The Triton compute cluster is composed of Appro gB222X Blade Server nodes with dual quad-core Intel architecture Nehalem, running at 2.40 GHz.

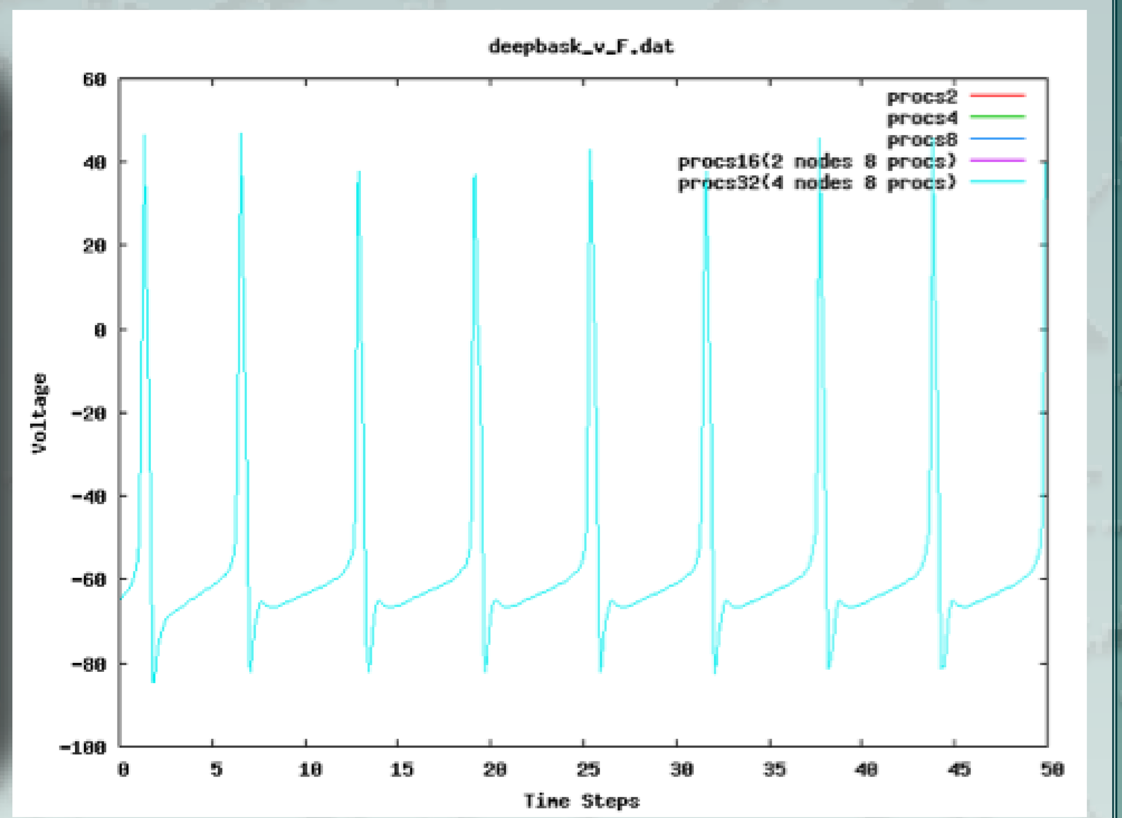
Models were run on varying numbers of processors on the Triton cluster to simulate smaller clusters, including desktop machines.

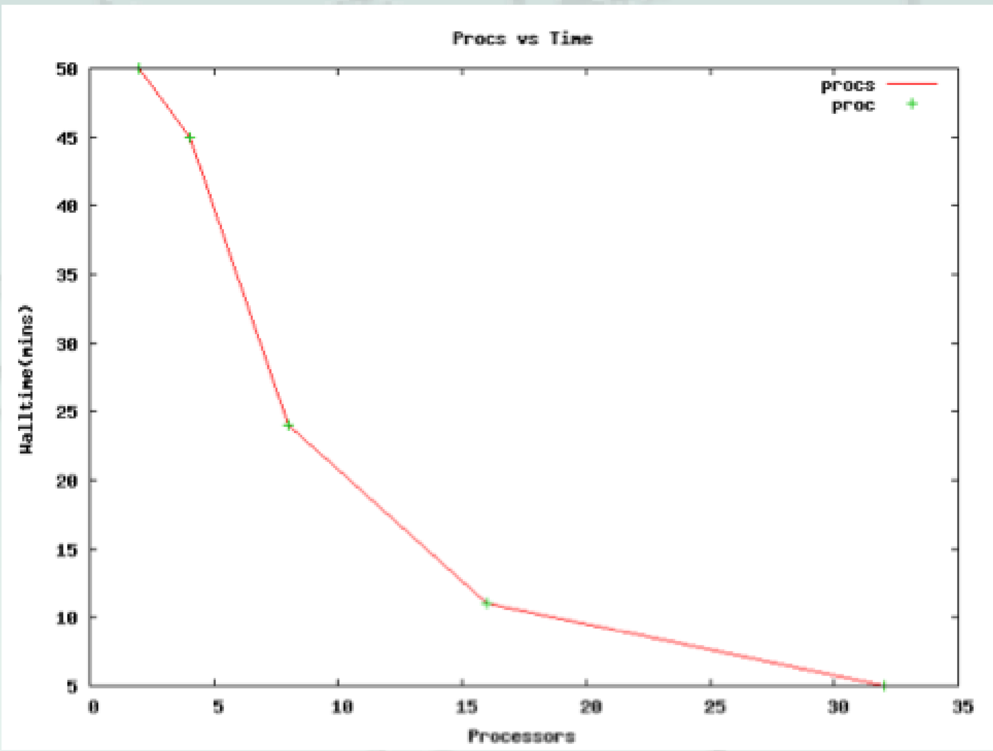


1. Acquired our model from the ModelDB database.
2. Ran simulation using Triton Supercomputer.
3. Produced data points corresponding to time steps and voltage.
4. Generated graphs with data

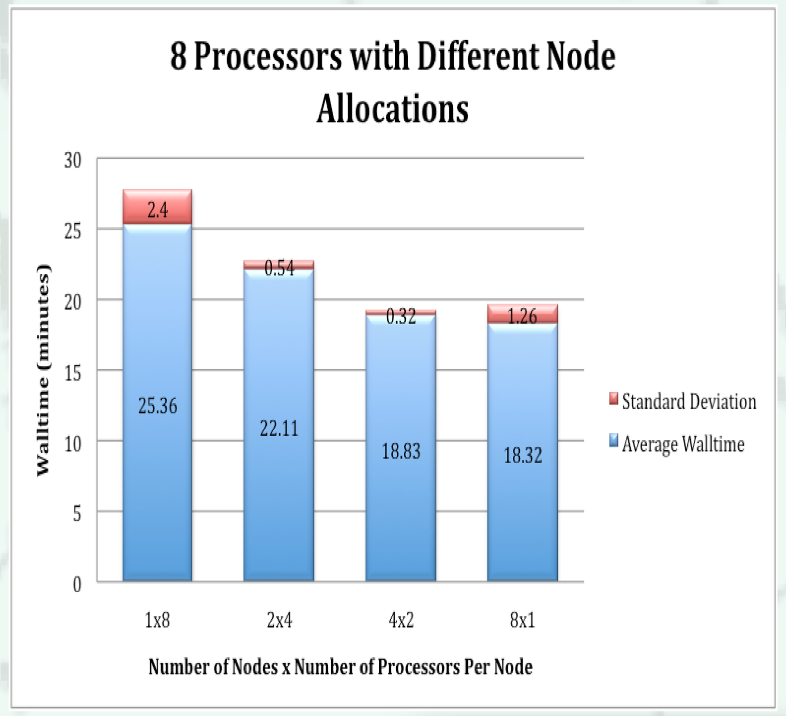
Results

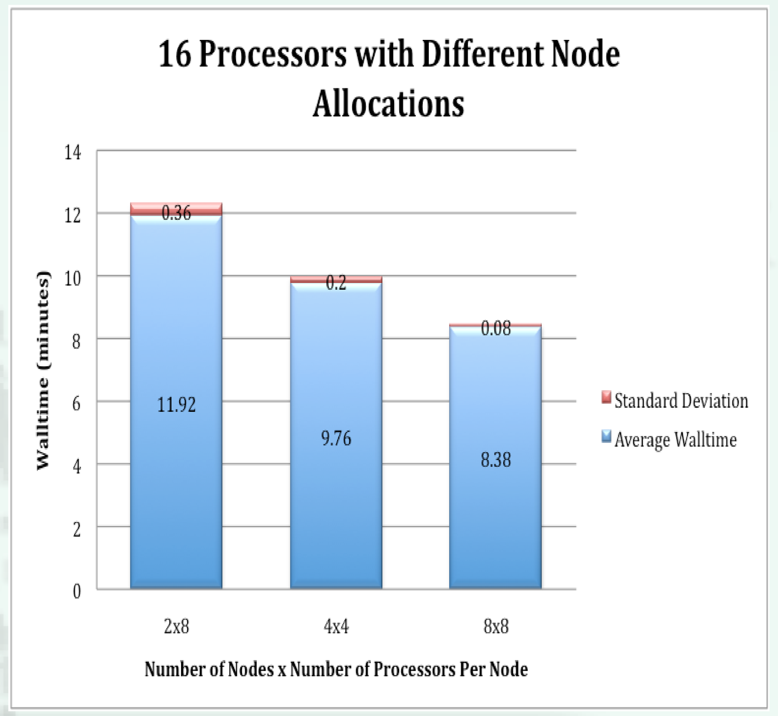
Consistency Results:
Output results are consistent when the model was run with different number of processors. The model used in this set of runs is the Traub (2005) model entitled "A single column thalamocortical network model" and it was obtained from modelDB (#45539). This model has 3,560 multi-compartment neurons, four output files are overlapped showing perfect alignment from one neuron.



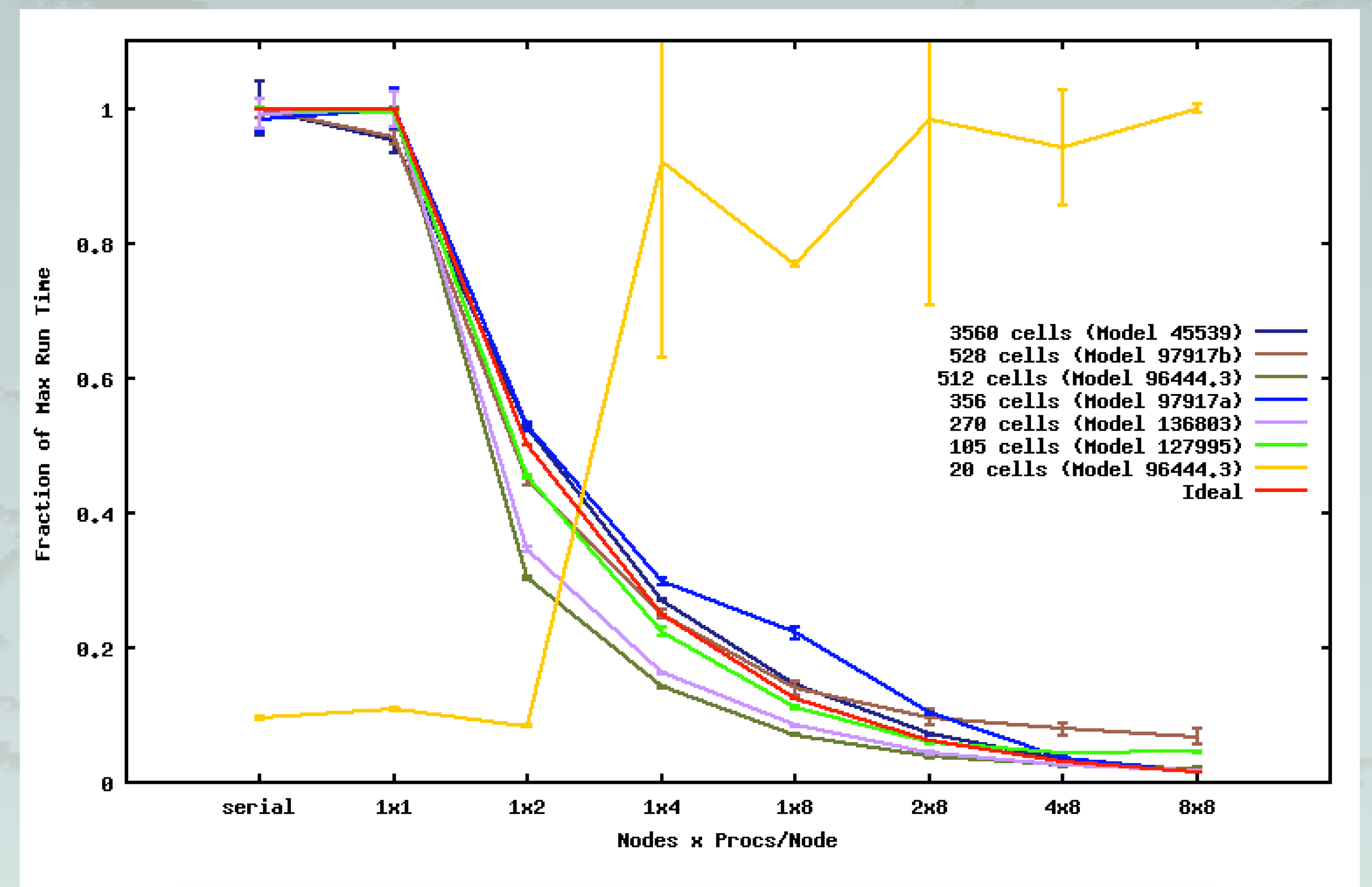


Scalability Results:
We observe a decrease in the wall clock time (execution time) as the number of processors increase.





Comparing Walltime for Different Nodes - Processors/node combination:
For model 45539 from modelDB, wallclock time was compared for different node-processors combination while keeping the overall processors per run a constant (8 or 16)



Scaling Results:
Models of different size (number of neurons) were run on varying number of processors. Large neuron-count models demonstrated reduced run time with higher processer counts. This was not true for low neuron count (20 neuron) models.

Conclusion:

• Parallel implementations of models with greater numbers of neurons performed well in terms of strong scaling where the simulation time decreased as the number of cores increased in a mode that provided sufficient memory per CPU.

•Models with less than 20 neurons didn't show a real need for using bigger computational clusters since serial runs performed better than parallel runs in terms of simulation time.

•The study identified various potential bottlenecks. Changing the number of processors per node, while maintaining the overall number of processors per run, can lead to different runtimes. Parallel performance was affected by caching. Interconnect speed and number of sockets per node could also affect the results on different clusters.

•The study proved that consistency is not affected by running a model with different processor counts. The study clearly showed that if parallelization of models was not done properly, the consistency of the results varied greatly.

We anticipate that dividing tasks of the neuronal model across the parallel cores may provide improved parallel performance which will be investigated further

NIH Blueprint
Funded in part by the NIH Neuroscience Blueprint HHSN271200800035C via NIDA
Funded in part by the Triton Research Grant