Parallelization of a Computational Model of a Biophysical Neuronal Circuitry of Rat Cerebellum

Afila Yoosef
Amrita School of Biotechnology
Amrita Vishwa Vidyapeetham (Amrita University)
Amritapuri, Kollam, 690525, Kerala, India
afilayoosef@am.amrita.edu

Harilal Parasuram
Amrita School of Biotechnology
Amrita Vishwa Vidyapeetham (Amrita University)
Amritapuri, Kollam, 690525, Kerala, India
harilalp@am.amrita.edu

Chaitanya Medini
Amrita School of Biotechnology
Amrita Vishwa Vidyapeetham (Amrita University)
Amritapuri, Kollam, 690525, Kerala, India
krishnachaitanya@am.amrita.edu

Sergio Solinas
Dept of Brain and Behavioral Sciences, University of Pavia, via Forlanini 6, 27100, Pavia, Italy
solinas@unipv.it

Egidio D'Angelo
Dept of Brain and Behavioral Sciences, University of Pavia, via Forlanini 6, 27100, Pavia, Italy
dangelo@unipv.it

Bipin Nair
Amrita School of Biotechnology
Amrita Vishwa Vidyapeetham (Amrita University)
Amritapuri, Kollam, 690525, Kerala,India
Ph:+914762896777
bipin@amrita.edu

Shyam Diwakar
Amrita School of Biotechnology
Amrita Vishwa Vidyapeetham (Amrita University), Amritapuri,
Kollam, 690525, Kerala, India
Ph:+914762803116
shyam@amrita.edu

ABSTRACT

Rapid progress in biophysical neural network modeling has been observed in the last years as a focus within computational neuroscience. Detailed multi-compartmental neuron models that were built to simulate physiological aspects of cerebellar neurons and microcircuits involve hundreds of equations. Simulating several hundreds of neurons is computationally expensive. Storage of data and run-time evaluations also prove to be major challenges in this kind of scenario, which limits the researchers.

In this paper, we use detailed models of neurons reconstructing the biophysics of cable properties and action ion channel models to generate a neural microcircuitry of cerebellum input layer. We report the process of adapting and profiling a parallel, MPI-based version of the network model on NEURON for large-scale simulations. Using multi-split and distributed approaches, our model was parallelized on multi-core, multi-processor systems. A spatio-temporal activation pattern, called the center-surround was elicited in the model validating the biophysical role of synaptic inhibition modulating excitatory activation, usually observed during sensory or tactile stimulation. Performance tests were carried out on two heterogeneous computing clusters. We see a significant reduction of computational cost in terms of power and time while simulating parallelized code although the most apt method depended on network size and nature of synaptic connections. We find 'embarrassingly parallel' method augmented efficiency in terms of processor core usage and also decreased simulation time.

ICONIAAC '14, October 10 - 11 2014, Amritapuri, India

http://dx.doi.org/10.1145/2660859.2660962

Categories and Subject Descriptors

I.6 [**Simulation and Modeling**]: Types of Simulation – *Discrete event, parallel*

General Terms

Performance, Algorithms, Theory, Measurement.

Keywords

Computational Neuroscience, Parallel Computing, Cerebellum Granule Cell, MPI

1. INTRODUCTION

Cerebellum has been known to play a critical role in movement coordination and balance. Several studies attributed to cerebellar function suggest spatio-temporal processing of signals [18][6]. Understanding spatio-temporal processing in such circuits through large scale mathematical modeling allows implications of millisecond-scale operations [3, 11] through intrinsic synaptic connections and the role of inhibitory neurons. Modeling the input layer cerebellar circuits require intensive computational power considering compartmental cable equations and modeling various ion channel dynamics [13]. Granule neurons in the input layer of the cerebellum account for 50% of total population of neurons although occupying 10% of the volume [12]. With the advent of high-performance computing clusters, we were able to take advantage of concurrent execution on multiple processors allowing us to simulate large-scale cerebellar circuits.

To face the challenges posed by constructing large neuronal models, computationally expensive models need to be moved to parallel simulations, for employing advantages of hardware with multiple processors and larger amount of memory. Previous studies indicate that large and complex network models can be allocated over several processors to obtain effective speedups[9]. Parallelization of circuit models with load-balancing have been studied on the NEURON simulation environment[8]. Embarrassingly parallel or pleasantly parallel models of simulations have also been employed as appropriate schemes for large-scale neuronal circuit simulations[14].

In the context of biophysically detailed models, NEURON simulation environment [8] allows distributed parallelization, embarrassingly parallel mode, multi-split method and split-cell method[1]. NEURON simulations may have Linda bulletin-board like system between the master and workers[2] or modes wherein each processor handles a part of the cell (load distributed over multiple processors)[1]. The current paper explains the process of parallelizing a central cerebellar microcircuit model for large scale simulations. The biophysically detailed network model of granular layer in rat cerebellum consisted of 720 granule cell models, 1 Golgi cell model and 55 glomeruli. Model details are already reported in a previous study [13]. We used the biophysical neuron models of a detailed multi-compartmental granule cell [4] and cerebellar Golgi cell [16][17] to generate a large scale detailed network of granular layer that reconstructed center-surround excitation patterns. Our aim was to understand the best parallelization approach for large-scale biophysically detailed neuron circuit models. In this paper, we compared the efficiency of the techniques using the distributed parallelization mode and multi-split method.

2. METHODS

As part of this study, we have used cerebellar granular layer network (720 granule cells) to analyze parallelization methods for event-related neural circuit models on a GNU/Linux-based computing cluster. All simulations were tested on NEURON platform compiled with MPICH2, which distributed the cells based on round-robin assignment [9]. Each node simulated an assigned number of cells and returned back the result to the master node. Scatter-Gather type of method [9] was used in these simulations. Master node split the large network model to multiple nodes thereby distributing jobs and then recollected the results at the end of simulation. Both serial version and parallel version of the network model were used to validate the increase in performance efficiency. In this paper, multi-split and embarrassingly parallel schemes of simulations were run on two different blade servers, one with a 132-core, 200 GB RAM – MPI-based cluster (built on Intel Xeon E5649 @2.53GHz processors) and a 24-core, 64GB RAM MPI-based cluster (built on Intel Xeon E5520 @ 2.27 GHz processors) to simulate parallel cerebellar granular layer network model and a workstation with 8GB RAM and Intel Xeon E5-2665 @ 2.40Ghz was used for comparing single neurons.

2.1 Cerebellar Granular layer network model

Detailed multi-compartmental models: granule cell model [4] and Golgi cell model [16] were used to build the network. Granule cell model consists of 52 active compartments [15]. The voltage difference (V_m) across the cell membrane for each compartment is calculated by:

$$\frac{dV_m}{dt} = \frac{1}{\tau_m}\left(V - \frac{\Sigma_i g_i(V - V_i) + \Sigma_{syn} g_{syn}(V - V_{syn}) + \Sigma_{br} g_{br}(V - V_{br})}{g_{tot}}\right)$$

Where $\tau_m$is the time constant ($\tau_m = C_m R_m$), R_m is the membrane resistance, C_m is the membrane capacitance, g is the conductance,syn (synaptic dynamics), br (neighbouring attached branch) and tot (total). Model also includes ion channels based on Hodgkin-Huxley like dynamics [4].

Table 1. Convergence/divergence ratio used in the granular layer model [5]

Connection	Convergence ratio	Divergence ratio
MF ⟶ GrC	4:1	1:53
MF ⟶ GoC	50:1	1:3.6
GrC ⟶ GoC	100:1	1:1.9
GoC ⟶ GrC	4:1	1:600

The network model included 720 multi-compartmental granule cells [4], 1 Golgi cell [16], 55 Mossy fibers (MF) and 8640 synapses to pack 30µm cubic slice of cerebellar cortex [18]. Convergence and divergence ratios (Table 1) were used to establish the synaptic connectivity (see Table 2) in the network model were based on earlier studies by[5]. Synaptic dynamics were reconstructed as reported in our previous study [4].

Network was simulated with center-surround excitation pattern. *In vitro* like behavior was simulated by giving single spike as input

through mossy fibers (MF)[13] while *in vivo* like behavior [13] was given to simulate T and C wave like input. The synaptic pattern (5%, 45%, 35%, 15%) was set based on [3] (see Fig. 1A).

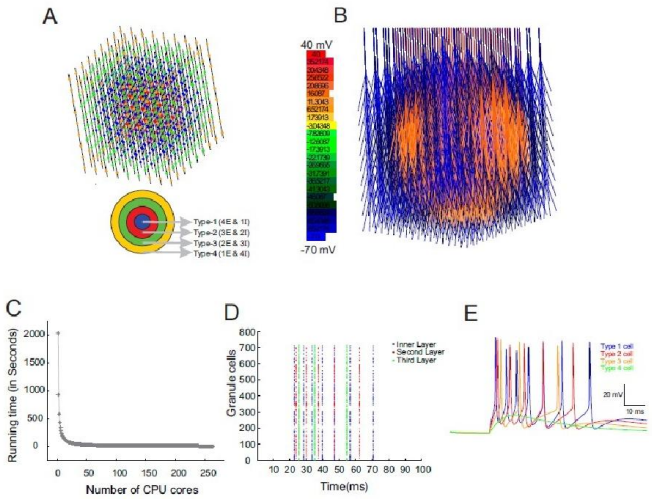

Figure 1. A. Center-surround excitation. Network model was simulated with center-surround "spot" activation. Dots (red, blue, green, orange) represent soma of cells in the network. Type-1 cells (5 % of total cells) located in the center of the network received 4 MF excitation (shown in red), Type-2 cells (45 % of total cells) in the network received 3 MF excitation (shown in blue), Type-3 cells (35 % of total cells) in the network received 2 MF excitation (shown in green) and Type-4 cells (15 % of total cells) in the network received 1 MF excitation (shown in orange). B. Visualizing network activity using voltage-sensitive dye (VSD) like color-map plots. Color-maps from blue to red were used to represent the voltage changes of soma when the network model was simulated with burst as input (in vivo like behavior). C. Parallel simulation of network model showed decrease in simulation time while increasing the number of cores used for simulation. D. Spike raster plots of granule cells in the network. Neurons in the network model could reproduce the spike raster for in vivo firing dynamics [2]. E. Granule cell firing behavior (in vivo). Voltage plots of the different cells in the network with varying excitatory inputs.

2.2 Parallel implementation of network model

NEURON environment implements an event delivery system for spike-triggered synaptic diffusion between cells [7]. To avoid connections in the same assignment [9], we used global identifier (gid) for each cell to be assigned to every existing host [9]. As suggested (in [9]), the parallel version allowed an information including the 'gid' and the spike time to be passed to all other hosts when a spike was generated by a presynaptic cell on one host [9]. Parallel simulation was achieved by using the NetCon and 'gid's on pre-cell and target post cell [9].

2.3 Multi-split algorithm - Parallelizing single neuron models

The multi-compartmental granule cell model [4] was used to implement the multi-split technique on single cell model [10].

In the multi-split method, the multi-compartment granule neuron was divided into a number of sub-trees. No sub-tree had more than two connection points to other sub-trees. Computationally, the divided sub-trees may be distributed on different processors. For the multi-split method, the multi-compartmental model of granule neuron was 'split' using NEURON's ParallelContext since it used a tractable Gaussian elimination scheme involving allocated multi-split components in order to increase efficiency compared to the serial version [10].

Table 2. Number of excitatory and inhibitory inputs to the network

Number of cells	Number of Excitatory inputs	Number of inhibitory inputs	Number of spikes	Spike timings
110	4	1	6	22.95, 29.075, 34.8, 41.45, 49.9, 62.125
110	3	2	3	23.825, 32.6, 42.5
211	2	3	EPSP	Nil
289	1	4	EPSP	Nil

2.4 Comparison of Multi-split and Embarrassingly Parallel Implementations

To understand the effective implementation of our granular layer network [13] on clusters, we compared two parallelization schemes namely, multi-split and 'embarrassingly' parallel methods. Embarrassingly parallel method used round-robin distribution to equally distribute the cells of the network over different processors. Multi-split method [10] numerically divided the multi-compartmental granule cell [4] mathematical model into two or more pieces in order to simulate codes with high-availability load balance. We adapted the multi-split technique [19] to validate the run-time efficiency against embarrassingly parallel simulations of the network. CPU complexity was estimated for the single neuron simulations. The number of cells were increased to calibrate the best model between multi-split and embarrassingly parallel techniques. In the multi-split technique, the cells were split into processes based on the load balance information generated. Run times for various configurations are shown in Table 3. While single neuron comparisons were performed on an Intel Xeon E5-2665 based workstation, comparison of multi-split vs embarrassingly parallel techniques on the network model was run on a 132-cores, 200 GB RAM-MPI-based cluster (running on an IBM blade server with Intel Xeon 2.53 GHz E5649 processors).

2.5 Number of processes and simulations

We ran two sets of simulations: by manually specifying the number of processes provided to mpirun. We used np=1 and

np=16 as two values to test the parallelization schemes of both single neuron model and a 720-cell granular layer network. All tests were carried out on a 24-core, 64 GB RAM MPI-based cluster (HP blade server with Intel Xeon E5520 @2.27 GHz processors).

2.6 CPU Complexity vs Number of Processes

We measured the complexity [19] for simulations run with varying number of processes (np=1,2..16) for the detailed multi-compartmental model.

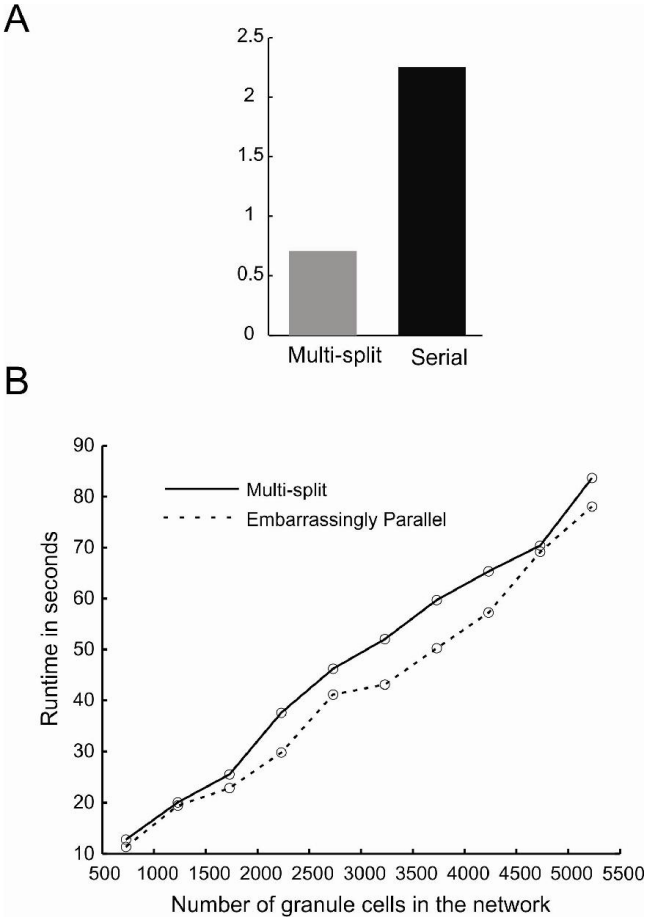

Figure 2. Comparison of parallel simulations. A. Comparison of multi-split single granule cell simulations runtime for 100ms simulation on workstation against serial implementation (black). B. Shows comparison of simulation runtime for multi-split (dark line) and embarrassingly parallel (dotted line) methods for varying network size scale-up. X-axis shows number of granule cells and while Y-axis shows simulation runtime.

3. RESULTS

We were able to reconstruct the model of granular layer network microcircuit using biophysically detailed multi compartmental neurons and could effectively parallelize the same using distributed parallelization. We compared the performance of both versions on serial and parallel hardware. Also, we were able to parallelize the single neuron models using multi-split scheme and were able to measure its performance on two different multiprocessor blade servers.

3.1 Granular layer activity and center-surround excitation

The network model nicely reproduced the biophysics of underlying cerebellum granular layer activity comparable to experimental validations [18][13][4]. The 3-D structure showed increased excitation in the center while lower excitation in the periphery (See Figure 1A). The ability of Golgi inhibition to differentially inhibit the same layer was shown in the reconstruction (See Figure 1B) and matched experimental results [4]. The model also reproduced spike timing (See Figure 1D) that matched with underlying cellular events (See Figure 1E). The network model closely reconstructed the spatio-temporal properties of sensory center-surround activation governed by cellular dynamics [13].

3.2 Parallel execution with increasing number of processes

Parallelization showed a significant decrease in runtime compared to serial implementations (See Figure 2A, Table 3). We used different number of processes explicitly to determine the effect on runtime efficiency of the network model. Parallel execution was effective when the total number of processes did not exceed total number of cores available on the system. We studied the effect of increasing number of processes until the maximum cores on the system on runtime efficiency of the network (Figure. 1C, 3, 5). There was a significant decrease in runtime efficiency as we increased the number of processes.

3.3 Multi-splitting a single neuron model

The implementation of multi-split scheme was done with the multi-compartmental granule cell model, which consisted of 52 active compartments including the soma with four dendrites and an axon [4]. We compared the multi-split implementation (gray) against the serial code (black) on a workstation (Figure 2A).

Another important observation was on the CPU complexity [19] of neuron model on the CPU. The granule cell model has maximum complexity of 1296 on a single processor. As the number of processes increased, complexity decreased and was reduced when extended to 16 processes (Figure 3). When the load imbalance exceeds > 10%, multi-split method need to be used.

3.4 Multi-split vs Embarrassingly Parallel – Comparison

The granular layer network was simulated in both parallelization schemes and outputs indicated embarrassingly parallel method provided significant performance than multi-split technique (Figure 2B). Simulations were performed with increased network size to test scale up and result suggest multi-split method was less effective than embarrassingly parallel technique with larger number of cells (See Table 3).

3.5 Runtime Efficiency Improved with Increased Number of Processes

With multi-split technique, runtimes decreased with more number of processes in both single granule cell and granular layer network model. 60% improvement was observed at the single neuron level simulations. Granular layer network with multi-split run with 16 processes (see Figure 4) showed significant decrease during runtimes. This may be favorable to implement large neuron models with few active compartments (with ion channel dynamics) and more passive compartments.

Table 3. Simulation run times for different number of granule cells.

#granule cells in network	Simulation run time for multisplit (s)	Simulation run time for embarrassingly parallel (s)
730	12.8	11.36
1230	20.06	19.45
1730	25.53	22.83
2230	37.58	29.79
2730	46.2	41.15
3230	52.03	43.1
3730	59.68	50.23
4230	65.32	57.23
4730	70.42	69.16
5230	83.69	78.06

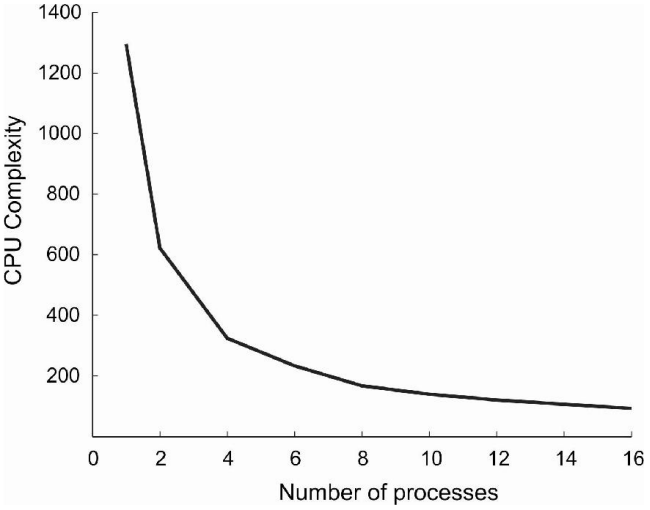

Figure 3. CPU complexity measured with increase in number of processes. Exponential decrease of complexity can be observed as number of process increases.

3.6 Network Models Showed Optimal Run Time Efficiency with Embarrassingly Parallel Technique

On the 132-node server, embarrassingly parallel technique showed optimal runtime decrease (see Figure 5). Simulation in embarrassingly parallel mode consisting of the granular layer network comprising of 720 neurons, with np=132, runtime was 15.62s, for np=4, it took 441.82s and for np=300, it was 15.85s.

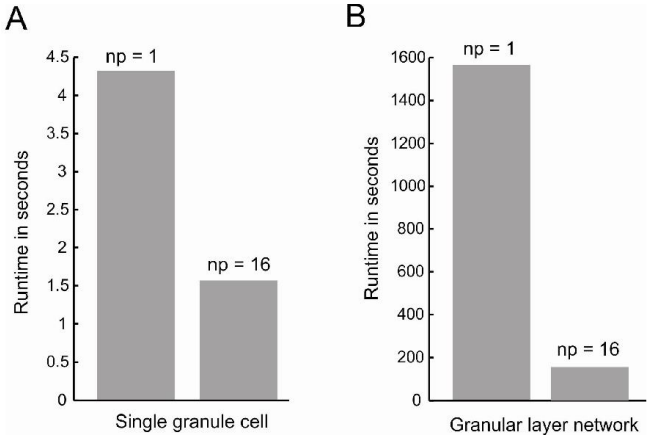

Figure 4. Comparing multi-split simulations with np=1 or 16. On x-axis plotted number of processor and on y-axis the total real time the machine took to complete the simulation.

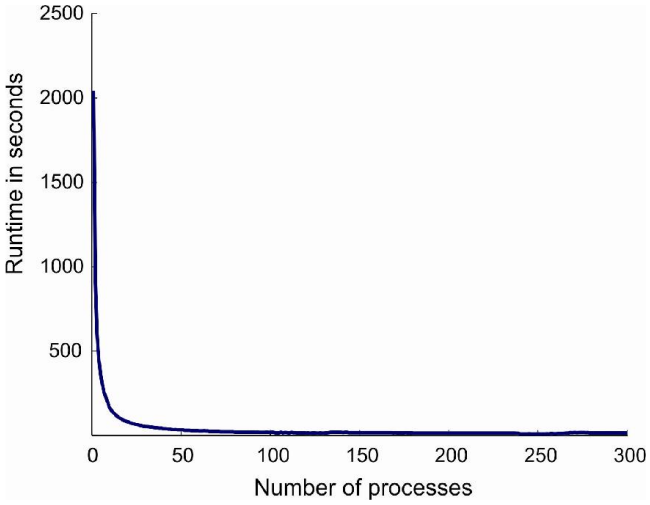

Figure 5. Embarrassingly parallel simulation, evaluating performance on number of processor. On x-axis plotted number of processor and on y-axis the total real time the machine took to complete the simulation.

4. DISCUSSION

The decrease in computation time on parallel executions in embarrassingly parallel mode allowed significant load balancing and runtime efficiency on our cerebellar granular layer model. Our tests suggest embarrassingly parallel modes scale reasonably better than multi-split methods on event-based models, perhaps due to the inherent parallel architecture of granular layer and

modular organization unique to some microcircuits in the brain [5] such as the cerebellum granular layer. We are yet to explore dynamic and hybrid load balancing techniques in detail although it is indicative from the current study that modular network models may perform better with embarrassingly parallel modes with higher number of cells in lieu of multi-split technique. When the number of processors is greater than the number of cells, the effective method was multi-split. This difference could be attributed to some of the processes being locked up at nodes during a multi-split mode of simulation. On cerebellum input layer models, embarrassingly parallel simulations elicit reasonable efficiency and decreased run-time compared to multi-split scheme.

Although computational circuit models could be parallelized using different approaches, a detailed study on load-balancing and assignment may require further validations using larger models. With this work as a precursor, we are extending the parallelization to our large-scale cerebellar microcircuits (~$2.5*10^5$ neurons).

5. ACKNOWLEDGMENTS

This work derives direction and ideas from the chancellor of Amrita University, Sri Mata Amritanandamayi Devi. The authors would like to thank Mithun Haridas and Nishitha Kalathil for their contribution to the present work. This work is supported by Grants SR/CSI/49/2010, SR/CSI/60/2011 and Indo-Italy POC 2012-2013 from DST and BT/PR5142/MED/30/764/2012 from DBT, Government of India.

6. REFERENCES

[1] Brette, R. et al. 2007. Simulation of networks of spiking neurons: a review of tools and strategies. *Journal of computational neuroscience*. 23, 3 (Dec. 2007), 349–98.

[2] Carriero, N. and Gelernter, D. 1989. Linda in context. *Communications of the ACM*. 32, 4 (Apr. 1989), 444–458.

[3] Diwakar, S., Lombardo, P., Solinas, S., Naldi, G. and D'Angelo, E. 2011. Local field potential modeling predicts dense activation in cerebellar granule cells clusters under LTP and LTD control. *PloS one*. 6, 7 (Jan. 2011), e21928.

[4] Diwakar, S., Magistretti, J., Goldfarb, M., Naldi, G. and D'Angelo, E. 2009. Axonal Na+ channels ensure fast spike activation and back-propagation in cerebellar granule cells. *Journal of neurophysiology*. 101, 2 (Feb. 2009), 519–32.

[5] Eccles, J.C., Ito, M. and Szentagothai, J. 1967. *The Cerebellum as a Neuronal Machine*. Springer.

[6] Gandolfi, D., Lombardo, P., Mapelli, J., Solinas, S. and D'Angelo, E. 2013. θ-Frequency resonance at the cerebellum input stage improves spike timing on the millisecond time-scale. *Frontiers in neural circuits*. 7, April (Jan. 2013), 64.

[7] Hines, M.L. and Carnevale, N.T. 2004. Discrete event simulation in the NEURON environment. *Neurocomputing*. 58-60, (Jun. 2004), 1117–1122.

[8] Hines, M.L. and Carnevale, N.T. 1997. The NEURON simulation environment. *Neural computation*. 9, 6 (Aug. 1997), 1179–209.

[9] Hines, M.L. and Carnevale, N.T. 2008. Translating network models to parallel hardware in NEURON. *Journal of neuroscience methods*. 169, 2 (Apr. 2008), 425–55.

[10] Hines, M.L., Markram, H. and Schürmann, F. 2008. Fully implicit parallel simulation of single neurons. *Journal of computational neuroscience*. 25, 3 (Dec. 2008), 439–48.

[11] Ito, M. 2000. Mechanisms of motor learning in the cerebellum. *Brain research*. 886, 1-2 (Dec. 2000), 237–245.

[12] Kandel, E.R., Schwartz, J.H., Jessell, T.M., Siegelbaum, S.A. and Hudspeth, A.J. *Principles of Neural Science*. McGraw-Hill Medical.

[13] Medini, C., Nair, B., D'Angelo, E., Naldi, G. and Diwakar, S. 2012. Modeling spike-train processing in the cerebellum granular layer and changes in plasticity reveal single neuron effects in neural ensembles. *Computational intelligence and neuroscience*. 2012, (Jan. 2012), 359529.

[14] Pacheco, P., Camperi, M. and Uchino, T. 2000. PARALLEL NEUROSYS: A system for the simulation of very large networks of biologically accurate neurons on parallel computers. *Neurocomputing*. 32-33, (Jun. 2000), 1095–1102.

[15] RALL, W. 1959. Branching dendritic trees and motoneuron membrane resistivity. *Experimental neurology*. 1, (Nov. 1959), 491–527.

[16] Solinas, S., Forti, L., Cesana, E., Mapelli, J., De Schutter, E. and D'Angelo, E. 2007. Computational reconstruction of pacemaking and intrinsic electroresponsiveness in cerebellar Golgi cells. *Frontiers in cellular neuroscience*. 1, (Jan. 2007), 2.

[17] Solinas, S., Forti, L., Cesana, E., Mapelli, J., De Schutter, E. and D'Angelo, E. 2007. Fast-reset of pacemaking and theta-frequency resonance patterns in cerebellar golgi cells: simulations of their impact in vivo. *Frontiers in cellular neuroscience*. 1, (Jan. 2007), 4.

[18] Solinas, S., Nieus, T. and D'Angelo, E. 2010. A realistic large-scale model of the cerebellum granular layer predicts circuit spatio-temporal filtering properties. *Frontiers in cellular neuroscience*. 4, (Jan. 2010), 12.

[19] Traub, R.D., Contreras, D., Cunningham, M.O., Murray, H., LeBeau, F.E.N., Roopun, A., Bibbig, A., Wilent, W.B., Higley, M.J. and Whittington, M.A. 2005. Single-column thalamocortical network model exhibiting gamma oscillations, sleep spindles, and epileptogenic bursts. *Journal of neurophysiology*. 93, 4 (Apr. 2005), 2194–232.