GPGPU Implementation of a Spiking Neuronal Circuit Performing Sparse Recoding

Manjusha Nair[1,2], Bipin Nair[1], and Shyam Diwakar[1]

[1] Amrita School of Biotechnology, Amrita Vishwa Vidyapeetham, Amrita University, Amritapuri, Kollam, Kerala, India
[2] Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amrita University, Amritapuri, Kollam, Kerala, India
manjushanair@am.amrita.edu, shyam@amrita.edu

Abstract. Modeling and simulation techniques have been used extensively to study the complexities of brain circuits. Simulations of bio-realistic networks consisting of large number of neurons require massive computational power when they are designed to provide real-time responses in millisecond scale. A network model of cerebellar granular layer was developed and simulated here on Graphic Processing Units (GPU) which delivered a high compute capacity at low cost. We used a mathematical model namely, Adaptive Exponential leaky integrate-and-fire (AdEx) equations to model the different types of neurons in the cerebellum. The hypothesis relating spatiotemporal information processing in the input layer of the cerebellum and its relations to sparse activation of cell clusters was evaluated. The main goal of this paper was to understand the computational efficiency and scalability issues while implementing a large-scale microcircuit consisting of millions of neurons and synapses. The results suggest efficient scale-up based on pleasantly parallel modes of operations allows simulations of large-scale spiking network models for cerebellum-like network circuits.

Keywords: Graphic Processing Units, cerebellum, computational Neuroscience,·neuron, synapse, adaptive Exponential Leaky Integrate and Fire Model.

1 Introduction

Computational modeling allows us to investigate behavior of the neurons and to frame or test hypothesis about their operations. Neural modeling at the level of ion-channel kinetics using Hodgkin-Huxley models had been useful in characterizing single neuron behavior. It is, however, computationally intensive to model large networks of neurons due to the number of simultaneous differential equations that must be evaluated and due to the abundant system parameters that need to be specified for the neuron being modeled. In order to perform information theoretic analysis of the spike responses, massive amount of data from simulations involving thousands of neurons are required. Well-known simulators such as NEURON[1]and GENESIS[2] are used widely for detailed biophysical simulations of single neurons or for the simulations of small

C. di Serio et al. (Eds.): CIBB 2014, LNCS 8623, pp. 285–297, 2015.
DOI: 10.1007/978-3-319-24462-4_24

network of neurons. Due to the computational overhead of the complex neuronal dynamics addressed by them, they fail to perform large-scale simulations in the timescale of the real network of brain. Hence they have been extended to support distributed simulations of biologically realistic network models[3,4]and are run in multi-CPU environments like multi-core processors and Beowulf clusters. Simulation with spiking neurons gained prominent importance in the computational neuroscience community to study the neuronal dynamics of large-scale microcircuits. Spiking Neural Network simulators like NEST[5] and SpikeNET[6]have also followed the same trajectory but used different computational models. Most of these simulators support networks of realistic connectivity employing multi-threading and message-passing interfaces on clusters of computers. With the newer multi-core processors designed as numeric computing engines and with their general purpose programming interfaces, recent computer hardware have shown significant efficiency improvements. Highly parallel programmable processors like GPUs deliver a high compute capacity at low cost. GPUs are enhanced with a greater arithmetic capability, streaming memory bandwidth and with a richer set of APIs. GPUs provide computing power that is easily and cheaply accessible to individuals who cannot afford clusters and supercomputers. Simple spiking neural network models such as Integrate and fire models without bio-realistic features were simulated in the older generation GPUs [7]. Another study focused on simulating a large scale Izhikevich-based realistic spiking network models having 10^5 neurons and 10^7 synaptic connections on GPUs[8]. More recent works proved the potential power of GPGPU techniques in real-time simulation of the different regions of the brain such as basal ganglia circuitry[9]and cerebellum[10]. The studies have demonstrated the use of GPU for neural network simulations.

We constructed a bio-realistic spiking network of neurons of the cerebellar granular layer. Since cerebellum contains more than half of the total population of the neurons in the entire brain, the implications of large-scale simulation become pertinent. Due to the 'embarrassingly' parallel architecture of different layers of neurons in the cerebellum and due to the modular connection geometry between them, we chose this model as a candidate for parallel simulation. Cerebellum is known to be involved in timing and in controlling the ordered and precise execution of motor sequences [11]. Input layer of the cerebellum has been studied for combinatorial operations. Due to its role in motor articulation control, cerebellar modeling is a main area of focus for many real-time robotic applications. Cerebellar granular layer consists of a large number of neurons that receive information from mossy fibers and spatially encode information which then converges onto Purkinje neurons via parallel fibers. The abundance, fast response time and modular architecture of the granular layer neurons of the cerebellum offered an opportunity as well as a challenge to this modeling process. A simple spiking neuron model in NEURON was tuned previously to predict how spikes were processed in the cerebellar granular layer network [12]. Biophysically realistic models of granule cells [11],[13,14] and Golgi cells [15] are available to map and test the known behaviors of granular layer neurons. Our goal was to understand and implement feasible fast models on GPUs. In this paper, individual granule neurons, Golgi neurons and their excitatory-inhibitory synapses were modeled.

A realistic large-scale model of the cerebellum granular layer [16] was reconstructed. The network was simulated on a Tesla K20C GPU with 2496 cores and SM 3.5 support running at 0.71 GHz. This study aimed to model and analyze a cerebellar network of granular layer neurons on GPUs in order to study the computational relevance of such implementations and to understand the role of parallelism in spatio-temporal encoding in the input layer of the cerebellum and.

2 Materials and Methods

2.1 Single Neuron and Synapse Modeling

Membrane and synaptic properties of two types of neurons in the granular layer were modeled using phenomenological models. Single neurons were modeled using adaptive exponential leaky integrate and fire model, a two-dimensional integrate and-fire model that combines an exponential spike mechanism with an adaptation equation, which was able to correctly predict timing of 96% of the spikes (±2 ms) and closely reconstructed the behavior as seen in a detailed conductance-based model [17]. The equations (1) and (2) of the model were able to generate different firing patterns and were used to simulate firing dynamics for single neurons in the network simulations [18].

$$\frac{dV}{dt} = \frac{-gl*(V-El)+gl*delT*\exp\left(\frac{V-Vt}{delT}\right)+Isyn-w}{C} \tag{1}$$

$$\tau w \frac{dw}{dt} = a*(V-El)-w \tag{2}$$

$$If\ V > 0\ mv,\ V = Vr\ \&\ w = w + b$$

The AdEx model is an extended integrate-and-fire model where the passive properties of the neuron and the action potential mechanisms are combined with the adaptation variable w. In the model, V represents the membrane voltage, C is the membrane capacitance, gl is the leak conductance, El is the resting potential, $delT$ is the slope factor, Vt is the threshold potential, Vr is the reset potential, $Isyn$ is the synaptic current, τw is the time constant, a is the level of sub-threshold adaptation and b represents the spike triggered adaptation. The first equation describes the dynamics of the action potential generation while the second equation describes adaptation in the firing rate of the neuron. The equation followed the dynamics of an RC circuit until V reaches Vt. The neuron fires on crossing this threshold voltage and the downswing of the action potential was replaced by a reset of membrane potential V to a lower value, Vr.

Granule cells in the cerebellum receive on an average 1 to 4 excitatory connections via mossy fiber (MF) inputs and 0 to 4 inhibitory inputs through Golgi cell synapses [14]. Variation in number of synaptic inputs affects spike responses in neurons. Bringing these synaptic behaviors [19] to artificial spiking neurons was essential in

understanding the various network dynamics. Excitatory synapses were modeled using AMPA receptor dynamics and inhibitory synapses were modeled using GABA receptor dynamics [20] as indicated in equations (3) and (4).

$$g_{AMPA} = \frac{g_{AMPAMax} \times e^{\frac{-t}{18}} \times \left(1 - e^{\frac{-t}{2.2}}\right)}{0.68} \tag{3}$$

$$I_{AMPA} = (V_m - E_{AMPA}) \times g_{AMPA}$$

$$g_{GABA} = \frac{g_{GABAMax} \times e^{\frac{-t}{25}} \times \left(1 - e^{\frac{-t}{1.0}}\right)}{0.84} \tag{4}$$

$$I_{GABA} = (V_m - E_{GABA}) \times g_{GABA}$$

The synaptic currents I_{AMPA} and I_{GABA} were modeled via ohmic conductance g_{AMPA} and g_{GABA} multiplied by the difference between the membrane potential V_m and the reversal potential of the synapses, which is represented by E_{AMPA} for AMPA synapses and E_{GABA} for GABA synapses respectively. The maximal conductance $g_{AMPAMax}$ and $g_{GABAMax}$ were adjusted to match the number of spikes experimentally [14].

2.2 Glomerular Organisation of the Granular Layer Network

Mossy fibers provide excitatory inputs to the granule cells through glutamatergic synapses and Golgi cell axons inhibit granule cell firing through GABAergic synapses. These synapses are located inside a glomerular structure [21]. Specific connection geometry exist between mossy fibers, granule cells and Golgi cells in the granular layer [22]. In this model, a 3D volume of the granular layer with 100 μ m edge length contained granule cells with density 4 x 10^6 / mm^3 (Fig. 1). Each granule neuron model received one to four excitatory connections from mossy fibers synapses and one to four inhibitory connections through Golgi cells [23]. The number of glomeruli were estimated using the convergence-divergence ratio of the mossy fiber-granule cell connections [24]. Each glomerulus received a mean of 53 dendrites from different granule cells and each granule cell had an average of 4 dendrites, each dendrites not extending to glomeruli farther than 40 μm.

Approximately, 2000 granule cells were inhibited by a Golgi cell. The length of the different axes of the cube was varied to incorporate increased or decreased volume of the 3D space while maintaining the geometric properties and convergence-divergence ratio.

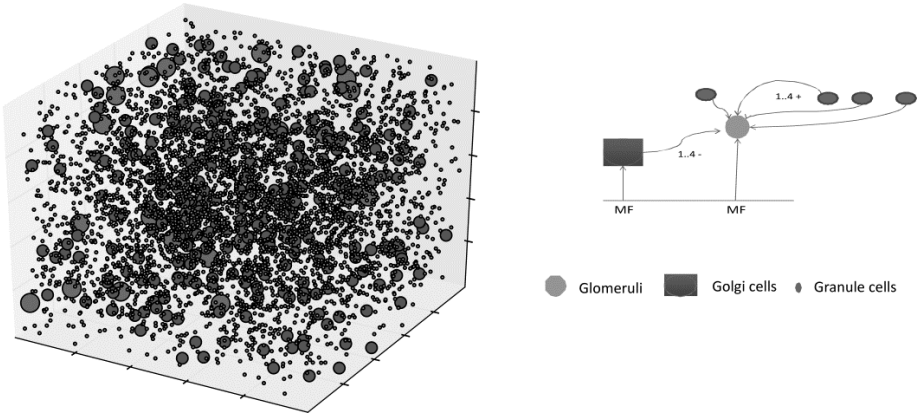

Fig. 1. Glomerular organization of the granular layer network model. Neurons were reconstructed within a spatial cube containing granule cell density 4 x 10^6 / mm ^{3} (LHS). Connectivity between mossy fibers, granule cell dendrites and Golgi cell axon are as indicated (RHS).

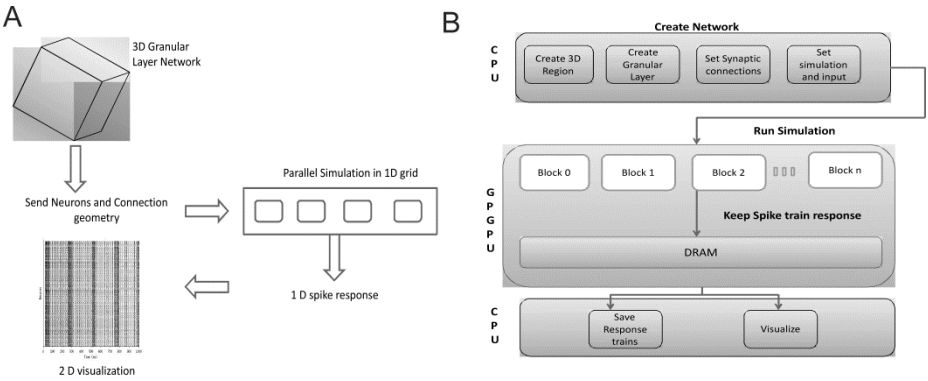

Fig. 2. GPU re-implementation of the granular layer network.(A) The Granular layer was modeled as a 3D cube with the volume occupied by the granule cells, Golgi cells and mossy fibers following precise connection rules. The connection geometry and neuron parameters were sent to the GPU for parallel simulation. The spike responses were collected for 2D visualization. (B) CPU and GPU task subunits for the simulator.

2.3 Simulation Background

The simulations were performed by activating a specific set of glomeruli or by activating entire glomeruli contained in the 3D space. Spike input of frequency 100 Hz were applied to the mossy fiber granule cell relay and the simulations were performed for different time windows ranging from 100 ms to 3 sec. Both *in vitro* like (1 spike per burst) and *in vivo* like (5 spike per burst) inputs were applied. The number of excitatory and inhibitory inputs to the granule cell was calculated at runtime depending on the connection rules and dynamics of the network.

2.4 Parallel Implementation

The cerebellar network reconstructed was both homogeneous and embarrassingly parallel. Standard fourth-order Runge-Kutta method was used for the numerical integration of the voltage equation of the neuron model. All neurons shared the same model equations and used the same integration steps for the computations. In order to achieve automatic scalability and increased efficiency, we adopted a single instruction multiple data paradigm for the simultaneous execution of different parts of the network on graphic processing units. Data-parallel processing mapped data elements to parallel processing threads. The essential serial components of the simulation such as initialization of the inputs and simulation parameters, network construction etc. were performed on an Intel Xeon CPU with 8 cores running on 2.6 GHz clock speed. The 3D network was constructed in the CPU and the neurons and connection information were sent as 1D arrays to the NVIDIA GPU: Tesla K20C with 2496 cores and 5 GB of DRAM. The parallelization can be summarized as follows.

```
BEGIN SIMULATION                                    }
   ReadSimulationParameters();                      }
   SetNeuronsAndSynapses();                         }   Serial Code
   CreateConnectionGeometry();                      }
   SimulateGolgiGranuleComplex()                    }
    BEGIN                                           }
     Forall  (neurons in 3D volume)                 }
     Until the end of simulation time               }
      EXCLUSIVE  BEGIN                              }
          Foreach synapse()                         }
            CalculateSynapticCurrent();             }
          SimulateGranuleCellsWithMF()              }   Parallel Code
          UpdateVoltage();                          }
          Foreach spike()                           }
            ModelReset();                           }
      EXCLUSIVE END                                 }
     Barrier;                                       }
     PackResponses();                               }
    END                                             }
    SaveResponse();                                 }
   VisualiseResponseTrains();                       }   Serial Code
END SIMULATION                                      }
```

The network of granule cells with 1 to 4 random MF connections with and without inhibition from Golgi cells were then parallel simulated in the GPU (Fig. 2: A & B). The spike responses of the cells were copied back to the CPU and were visualized as 2D raster plots. Each neuron was mapped to one thread of execution in the parallel GPU blocks and both thread level and block level parallelism were explored. The

memory requirement for CPU and GPU processes were calculated at runtime and the number of thread blocks were allocated in a scalable manner.

In a traditional CPU, if T time units were required to process each neuron, the time complexity of the entire simulation became directly proportional to M * N * T where M is the number of time steps and N is the number of neurons. In the GPU implementation, for P threads running in parallel where P==N, the total computations for N neurons takes only T time units and hence the total time required became directly proportional to M, the number of time steps, which was a constant. The time complexity of the algorithm was significantly reduced, when executed in parallel.

3 Results and Discussion

3.1 Single Neuron Simulations and Neuronal Firing Dynamics

The adaptive exponential leaky-integrate and fire model (AdEx) generated firing patterns depending on the parameters of the model equations [18]. The scaling and bifurcation parameters of the spiking neuron model were fine-tuned to match the electrophysiological recordings of the granule and Golgi neurons using current clamp protocol. Basic electro-responsiveness properties of both granule neurons and Golgi neurons (Fig. 3A & B) showed the increased firing rate and decreased first spike latency progressively with the injected current. Golgi cells showed spontaneous pace maker activity with a frequency between 1 and 8 Hz at room temperature [25] while granule cells showed no such spontaneous activity at rest but showed regular repetitive firing at current injection [26].

Dynamics of the synapses added significant computational overhead towards the overall time required for completing the simulation. The modeled granule cells contained 1 to 4 mossy fiber excitatory connections and 1 to 4 Golgi cell inhibitory connections. The synaptic dynamics with excitatory and inhibitory inputs were modeled using AMPA and GABA kinetics and the maximal conductance value was adjusted to suit the firing patterns [12] of granule cells during in-vitro (1 spike/burst) and in-vivo (5 spikes/burst) like inputs (Fig. 4: A &B). It was observed that two or more mossy fibers excitation was required to produce a spike output in granule cell. Also, increase in excitation increased the number of spikes while the increase in inhibition reduced the number of spikes. In order to apply tactile inputs which are seen essential for fine motor control, mossy fiber burst input was also applied to the cells (Fig. 4: C). Simulations allowed comparing the effect of these different types of inputs to the network.

For fidelity analysis, single neuron simulations were performed on CPUs and on GPUs and the firing patterns and the frequency of responses were compared. Even though both simulations produced similar numerical reconstructions, single neuron simulations took more time in GPU than in CPU. CPU simulations took 163.21 ms for a single granule cell and 172.35 ms for a single Golgi cell while the GPU simulations took 741.24ms and 1466.51ms respectively with inputs run for a total of 1000ms duration. The result indicated that CPU simulation was approximately 6x faster while GPU simulation was near real-time speeds compared to the biological neuron. The difference in performance time for a single neuron was consequential because CPU is faster on a per-core basis. We presume the delay to arithmetic pipelines and to the need of concurrent threads to sufficiently utilize the parallelism capabilities of the GPU.

3.2 Center-Surround Excitation in the Granular Layer

The distributed processing and plasticity capabilities of the neural network have been known to be dependent on spatial organization [27]. It has been observed that the activation of mossy fiber bundles to granular layer happens on an average in a center-surround manner with decreasing excitation from the center to the periphery for burst stimulation during the sensory inputs [16]. The same activation pattern was also observed when mossy fibers were stimulated with an electrode at specific locations[28]. Center surround structure of the granular layer determines the geometry of activation of the overlying Purkinje neurons of the molecular layer[11]. Centre-surround hypothesis was tested by giving strong excitation in the centre of the granular layer volume and progressively less excitation moving to the periphery (Fig. 5). 5% of glomeruli at the center received 4 MF inputs, 30 % of the surrounding glomeruli received 3 MF inputs, 55% received 2 MF inputs and the remaining 10% in the outer layer received 1 MF input. Granule cell responses to *in vivo* inputs (burst of 5 spikes at 500 Hz) in the centre showed bundle of spikes with shorter delay while the spike bursts in the surround showed reduced spike rate and longer delay[12]. Neurons in the centre responded to spike bursts over a broader frequency region while varying the frequency of the inputs.

3.3 Parallel Network Simulation

A network of granule cells with 1 to 4 random MF connections with and without inhibition from Golgi cells were simulated for 1000 ms time with *in vivo* burst-like input. We considered instantaneous post-synaptic current and hence white Gaussian noise was added to the network. Simulating a volume containing 4096 granule cells and 27 Golgi cells on GPU for 1000 ms took 3492.76 ms to complete the computations and memory transfers while the same in a single CPU took 2534445.25 ms. The GPU simulation was found to be 3.4 times slower than biological neural circuits while the single CPU simulation was 2534 times slower than biological networks. The results indicated the advantage of using GPUs for simulations of large network of neurons. A raster plot of spike responses during the simulation is shown (Fig.6: A). Scalability of the network implementation was tested by increasing the volume of the cube and measuring the computational time required to complete the simulation. Both CPU and GPU time taken for the network was calculated to justify the use of GPUs for large network simulations (Fig. 6: C). 550000 neurons were simulated in GPU and the running time linearly increased with the problem size (Fig. 6: B).

Since each neuron in the granular layer processes the input independent of other neurons in the same layer, our embarrassingly parallel approach of computations took little communication of results between tasks. Hence, no special algorithms were needed to get a working solution. A single large volume of the granular layer was divided into many smaller volumes which are handled by different simultaneously executing blocks of the GPU. Each neuron was mapped to one thread of execution and the simulations of a network of granular layer neurons were performed with

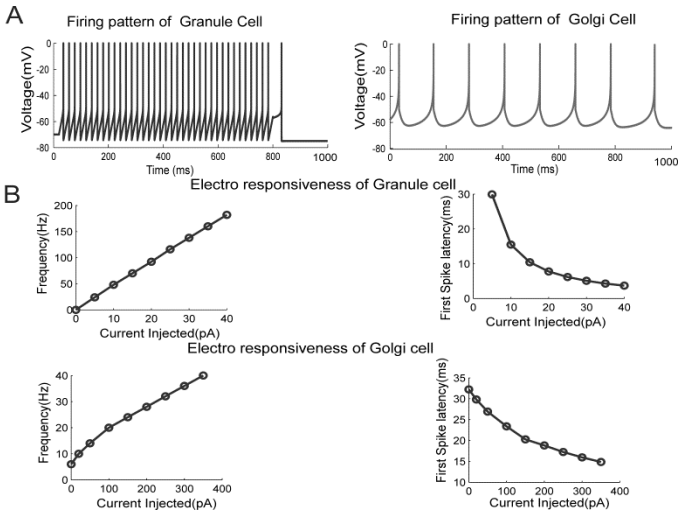

Fig. 3. Single cell electroresponsive properties of granule and Golgi cell. A) Firing patterns of granule and Golgi cell for 10 pA input current. B) The frequency of both granule and Golgi cell firing increased with the increase in current injection while the first spike latency is decreased.

N-parallel threads in a single block. Both block level and thread-level parallelisms were used for the simulations. For single neurons and network of size less than 1024, only thread-level parallelism was used. For networks of larger size, different concurrent blocks were launched. The thread assignment was a multiple of the warp size such that warp scheduling problem was avoided. GPU allowed automatic scalability with the increase in size of the granular layer network without modifications in the program.

Optimal GPU implementation not only depended on parallelization of the underlying algorithm or computations but also on memory optimizations and thread management[29].

Sufficient Parallelism: GPUs gave better performance improvements than CPUs only when sufficient parallelism was employed to hide the latency of the arithmetic pipelines. This was evident from the running time obtained while simulating single neurons on CPUs and GPUs. Since a single neuron was mapped to a single thread in the GPU simulation, the minimum network size was selected as 1024 to sufficiently exploit the parallelism in a single block of the GPU. This utilized the maximum number of threads allocated on a single block in a Tesla K20C GPU card.

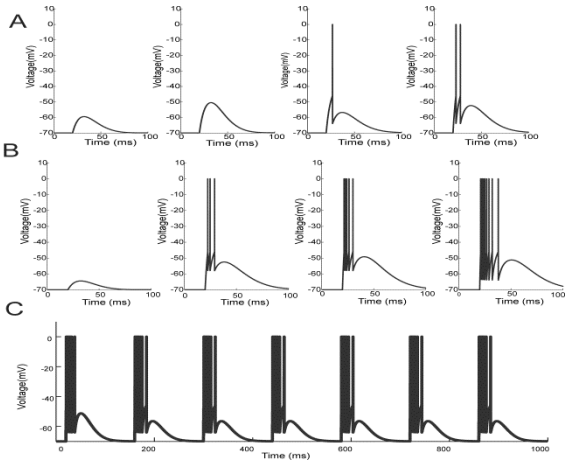

Fig. 4. Simulated response of granule neurons for various inputs. The inputs were provided starting at 20 ms. A) Granule cell responses for *in vitro* inputs (1 spike/burst). The MF inputs were increased from 1 to 4 in the figures from left to right. B) Granule cell responses for *in-vivo* inputs (5 spikes/burst, inter-spike interval 10 ms). The MF inputs were increased from 1 to 4 in the figures from left to right. C) Granule cell responses for tactile inputs (5 spikes/burst, inter-spike interval 10ms, inter-burst interval 100 ms).

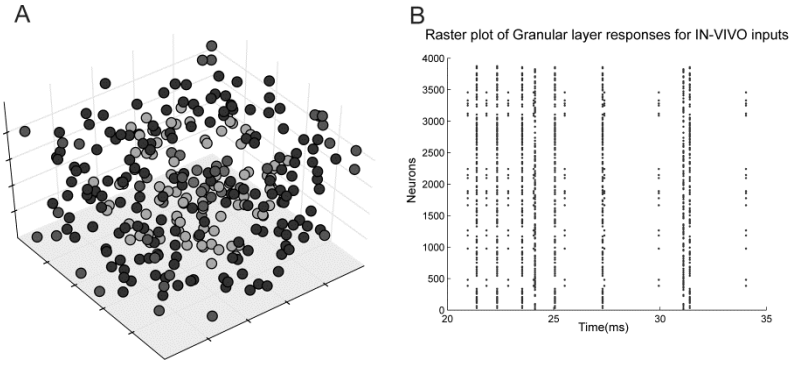

Fig. 5. Network simulation with center-surround excitation. Glomeruli were activated in a center-surround manner. A) The glomeruli in the center received four excitatory and the excitation is reduced progressively going to the periphery. B) The network was simulated with *in vivo* inputs (5 spikes /burst) and the spike responses were shown as the raster plot.

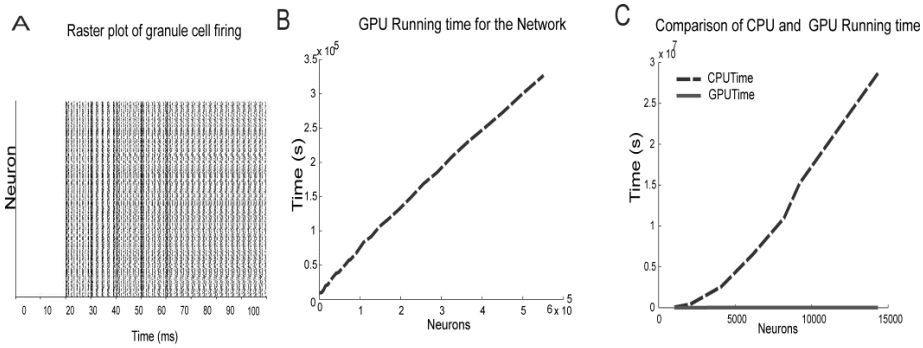

Fig. 6. A): Raster plot of granule cell firing patterns with white Gaussian noise in the network. *In vivo* like input (5 spikes per burst, inter spike interval 10 ms) was provided to the glomerulus through mossy fibers. B) The granular layer network was simulated with 550000 neurons with the granular layer volume increased from 125μm^3 to 125mm^3. The GPU runtime was found to be increasing linearly with the increase in network size. (C) A similar network was simulated entirely in a single CPU and the running time was compared. GPUs outperformed CPUs and took significantly lesser time to finish the simulation.

Minimizing Warp Divergence. Since all the neurons were simulated for almost same timescales and each neuron received on an average of 4 MF synapses, the wrap divergence and thread wait were not major issues in our current implementation.

Size of Thread Block and Occupancy: The kernel, *SimulateGranuleCellsWithMF()* which was ranked first for optimization based on execution time was chosen for performance improvement since the percentage of total GPU compute time spent executing instances of this kernel was found to be 70%. The register usage was limited by managing register spill-over using local memory (L1). LMEM is slower than registers. A 100% occupancy was not the aim of GPU optimizations since it slowed small network models although it improved runtime on large-scale network models.

4 Conclusion

We were able to reconstruct a cerebellum granular layer microcircuit in order to characterise the activity of a network of neurons sparsely activated by synaptic inputs in the rat cerebellum for the analysis of spatio-temporal geometry affecting signal activation in a central neural circuit. Even though we performed our simulations on a single high end GPU device, this study was a precursor for scaling up of the network model to include different layers of the cerebellum with greater number and more types of neurons to be simulated on clusters of GPUs. GPU based simulations may need to focus on lesser communications and pleasantly parallel or embarrassingly parallel schemes may be apt for large-scale neural simulations rather than fine-grained parallelization. Fixed time-step was more suited for such event-related simulations since variable time-step integration for several neurons caused performance decreases. Occupancy had to be pre-estimated

as 100% occupancy was not favorable for small networks since it increased runtime. Sufficient parallelism was essential to compensate latency delays in arithmetic pipelines, which was observed for single neuron and small-sized network simulations.

Extending the current cerebellar model including the molecular and Purkinje layer neurons together with the presently modeled granular layer neurons will make the network a right candidate to explore the other forms of parallelism with dependent computations. As a work in progress, we have started investigating a large-scaled network model on multi-GPU machines. Further studies may be necessary to understand the inherent parallelism and spatial recoding in cerebellar circuits from reconstructed network models.

Acknowledgements. This work derives direction and ideas from the chancellor of Amrita University, Sri Mata Amritanandamayi Devi. This work is supported by NVIDIA CTC grants 2012–13, 2013-14, 2015-16 and partially by DST SR/CSI/49/2010 and SR/CSI/60/2011 and Indo-Italy POC 2012-2013 from DST and BT/PR5142/MED/30/764/2012 from DBT, Government of India.

References

1. Hines, M.L., Carnevale, N.T.: The NEURON simulation environment. Neural Comput. 9(6), 1179–1209 (1997)
2. Bower, J.M.: GEneral NEural SImulation System (2003)
3. Hines, M.L., Carnevale, N.T.: Translating network models to parallel hardware in NEURON, 169(2) (2008)
4. Goddard, N.H., Hood, G.: Large Scale simulation using parallel GENESIS. In: The Book of Genesis, pp. 349–380 (1996)
5. Plesser, H.E., Eppler, J.M., Morrison, A., Diesmann, M., Gewaltig, M.-O.: Efficient parallel simulation of large-scale neuronal networks on clusters of multiprocessor computers. In: Kermarrec, A.-M., Bougé, L., Priol, T. (eds.) Euro-Par 2007. LNCS, vol. 4641, pp. 672–681. Springer, Heidelberg (2007)
6. Delorme, A., Thorpe, S.J.: SpikeNET: An Event-driven Simulation Package for Modeling Large Networks of Spiking Neurons. Netw. Comput. Neural Syst. 14, 613–627 (2003)
7. Bernhard, F.: Spiking Neurons on GPUs (2005)
8. Nageswaran, J.M., Dutt, N., Krichmar, J.L., Nicolau, A., Veidenbaum, A.: Efficient simulation of large-scale Spiking Neural Networks using CUDA graphics processors. In: 2009 Int. Jt. Conf. Neural Networks, pp. 2145–2152, June 2009
9. Igarashi, J., Shouno, O., Fukai, T., Tsujino, H.: Real-time simulation of a spiking neural network model of the basal ganglia circuitry using general purpose computing on graphics processing units. Neural Netw. 24(9), 950–960 (2011)
10. Yamazaki, T., Igarashi, J.: Realtime cerebellum: A large-scale spiking network model of the cerebellum that runs in realtime using a graphics processing unit. Neural Netw., February 2013
11. D'Angelo, E.: Neural circuits of the cerebellum: hypothesis for function. J. Integr. Neurosci. 10(3), 317–352 (2011)
12. Medini, C., Nair, B., D'Angelo, E., Naldi, G., Diwakar, S.: Modeling spike-train processing in the cerebellum granular layer and changes in plasticity reveal single neuron effects in neural ensembles. Comput. Intell. Neurosci. 2012, 359529 (2012)

13. Nieus, T., Sola, E., Mapelli, J., Saftenku, E., Rossi, P., D'Angelo, E.: LTP regulates burst initiation and frequency at mossy fiber-granule cell synapses of rat cerebellum: experimental observations and theoretical predictions. J. Neurophysiol. 95(2), 686–699 (2006)
14. Diwakar, S., Magistretti, J., Goldfarb, M., Naldi, G., D'Angelo, E.: Axonal Na+ channels ensure fast spike activation and back-propagation in cerebellar granule cells. J. Neurophysiol. 101(2), 519–532 (2009)
15. Solinas, S., Forti, L., Cesana, E., Mapelli, J., De Schutter, E., D'Angelo, E.: Computational reconstruction of pacemaking and intrinsic electroresponsiveness in cerebellar golgi cells, vol. 1, December 2007
16. Solinas, S., Nieus, T., D'Angelo, E.: A realistic large-scale model of the cerebellum granular layer predicts circuit spatio-temporal filtering properties. Front. Cell. Neurosci. 4, 12 (2010)
17. Brette, R., Gerstner, W.: Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. J. Neurophysiol. 94(5), 3637–3642 (2005)
18. Naud, R., Marcille, N., Clopath, C., Gerstner, W.: Firing patterns in the adaptive exponential integrate-and-fire model. Biol. Cybern. 99(4–5), 335–347 (2008)
19. Bengtsson, F., Jörntell, H.: Sensory transmission in cerebellar granule cells relies on similarly coded mossy fiber inputs. Proc. Natl. Acad. Sci. U.S.A. 106(7), 2389–2394 (2009)
20. David, J.H., McCormick, A., Wang, Z.: Neurotransmitter Control of Neocortical Neuronal Activity and Excitability. Cereb. Cortex 3(5), 387–398 (1993)
21. Rossi, D.J., Hamann, M.: Spillover-Mediated Transmission at Inhibitory Synapses Promoted by High Affinity α 6 Subunit GABA A Receptors and Glomerular Geometry. Neuron 20, 783–795 (1998)
22. Purve, D.: Neuroscience. Sinauer Associates, Inc., Sunderland (2004)
23. D'Angelo, E., Solinas, S., Mapelli, J., Gandolfi, D., Mapelli, L., Prestori, F.: The cerebellar Golgi cell and spatiotemporal organization of granular layer activity. Front. Neural Circuits 7, 93 (2013)
24. Solinas, S., Nieus, T., D'Angelo, E., Bower, J.M.: A realistic large-scale model of the cerebellum granular layer predicts circuit spatio-temporal filtering properties, 4, 1–17, May 2010
25. Forti, L., Cesana, E., Mapelli, J., D'Angelo, E.: Ionic mechanisms of autorhythmic firing in rat cerebellar Golgi cells. J. Physiol. 574(Pt 3), 711–729 (2006)
26. D'Angelo, E., De Filippi, G., Rossi, P., Taglietti, V., Liu, A., Regehr, W.G., Maejima, T., Wollenweber, P., Teusner, L.U.C., Noebels, J.L., Herlitze, S., Mark, M.D., Brackenbury, W.J., Calhoun, J.D., Chen, C., Miyazaki, H., Nukina, N., Oyama, F., Ranscht, B., Isom, L.L., Filippi, G.D.E.: Ionic Mechanism of Electroresponsiveness in Cerebellar Granule Cells Implicates the Action of a Persistent Sodium Current Ionic Mechanism of Electroresponsiveness in Cerebellar Granule Cells Implicates the Action of a Persistent Sodium Current. J. Neurophysiol., 493–503 (1998)
27. Mapelli, J., D'Angelo, E.: The spatial organization of long-term synaptic plasticity at the input stage of cerebellum. J. Neurosci. 27(6), 1285–1296 (2007)
28. Jonathan Mapelli, E.D., Gandolfi, D.: Combinatorial Responses Controlled by Synaptic Inhibition in the Cerebellum Granular Layer. J. Neurophysiol. 103(1), 250–261 (2010)
29. Hwu, W.W., Kirk, D.B.: Programming Massively Parallel Processors: A Hands-on Approach. Morgan Kaufmann (2009)