GPGPU Implementation of Spiking Neural Networks

Vivek K. Pallipuram, Mohammad A. Bhuiyan, and Melissa C. Smith
The Holcombe Department of Electrical and Computer Engineering, Clemson University

There has been a strong interest in the neuroscience community to model a mammalian brain in order to study its architecture and functional principles. Spiking Neural Network (SNN) models have been widely employed to simulate the mammalian brain, capturing its functionality and inference capabilities. The biologically accurate models from this class include the Hodgkin-Huxley (HH) model [1], Morris-Lecar (ML) model [2], Wilson model [3], and the Izhikevich model [4]. The HH model is the most compute intensive of the aforementioned models whereas the Izhikevich model only requires nominal amount of computation. The ML and Wilson models fall in between the two extremes. Since the SNN models are highly data parallel, they are an excellent fit for the General Purpose Graphical Processing Units (GPGPUs) that have massively parallel architectures. This allows for large scale SNN simulations in order to simulate neurons in near real-time.

We present our research on GPGPU acceleration of the SNN models using a two-level character recognition network capable of recognizing 48 alpha-numeric characters as shown in Figure 1. The compute intensive recognition phase (level-1) was implemented on the GPGPU whereas the output phase of the network (level-2) was implemented on the host processor. For our experiments, we have used a 2.66 GHz Intel Core 2 Quad host processor coupled with Nvidia's state-of-the-art *Fermi* architecture and Compute Unified Device Architecture (CUDA) as the programming model [5]. The networks were scaled from 9264 neurons (96 x 96 image size) upto 9.73 million neurons (3120 x 3120 image size). The focus of this work includes: 1) Analysis of thread-level data-level parallelism for the four SNN models used in the character recognition network. 2) Network scalability and speed-up performance of the single-GPGPU implementation of the SNN models. 3) Exploration of the GPGPU optimization techniques. To better study the GPGPU optimization techniques, we develop a hierarchy of three implementations progressively adding the CUDA optimization techniques [6]. The final implementation of the hierarchy exhausts all of the optimization techniques plausible with the CUDA programming paradigm.

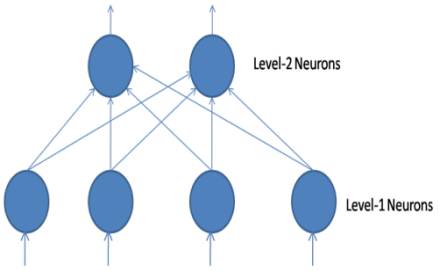

Figure 1. Two-level Character Recognition Network

The performance was found to depend on the communication-to-computation requirements of each of the SNN models, and the way they were mapped on the GPGPU. All of our parallel implementations were compared against the Intel Core 2 Quad host processor. The speed-up values were found to increase with the increase in the network size. For the largest network size 3120 x 3120, our final implementation of the hierarchy achieved a speed-up of 976x for the compute dense HH model, 190x for the ML model, 20x for the Wilson model, and 14x for highly compute efficient Izhikevich model. Figures 2 and 3 provide the performance results for the HH and Izhikevich models respectively. This work not only achieves significant speed-up and high network scalability for the SNN models, but also highlights efficient use of device optimization techniques and several bottlenecks that must be avoided to achieve high application performance.

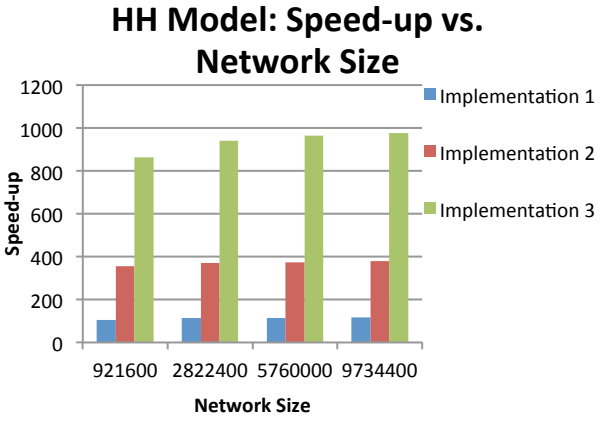

Figure 2. HH Model: Speed-up vs. Network Size

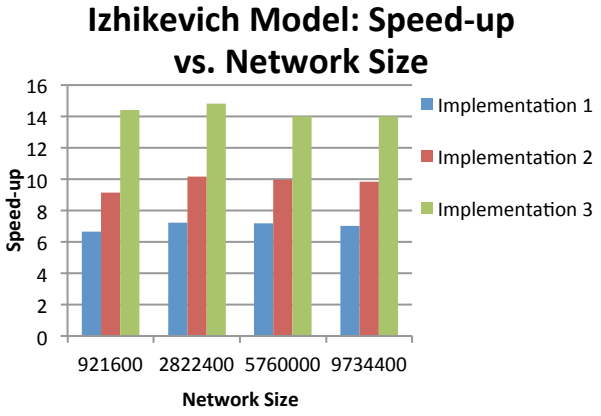

Figure 3. Izhikevich Model: Speed-up vs. Network Size

References

[1] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and application to conduction and excitation in nerve," *Journal of Physiology*, vol. 117, pp. 500-544, 1952

[2] C. Morris and H. Lecar, "Voltage oscillations in the barnacle giant muscle fiber," *Biophys. J.*, vol. 35, pp. 193-213, 1981

[3] H. R. Wilson, "Simplified dynamics of human and mammalian neocortical neurons," *J. Theor. Biol.,* vol. 200, pp. 375-388, 1999

[4] E. M. Izhikevich, "Simple Model to Use for Cortical Spiking Neurons," *IEEE transactions on Neural Networks*, vol. 14, no. 6, pp. 1569-1572, November 2003

[5] "NVIDIA CUDA Programming Guide", http://developer.download.nvidia.com/compute/cuda/3_0/toolkit/docs/NVIDIA_CUDA_ProgrammingGuide.pdf

[6] Vivek K. Pallipuram, Mohammad A. Bhuiyan, and Melissa C. Smith, "A comparative study of GPU programming models and architectures using neural networks", *The Journal of Supercomputing,* DOI: 10.1007/s11227-011-0631-3