A comparative evaluation of Xeon Phi platforms based on a Hodgkin-Huxley Neuron Simulator

George Chatzikonstantis*, Diego Jiménez†, Esteban Meneses†‡, Christos Strydis§, Harry Sidiropoulos*, Dimitrios Soudris*

* Microprocessors and Digital Systems Lab, National Technical University of Athens

†Advanced Computing Laboratory, Costa Rica National High Technology Center

‡School of Computing, Costa Rica Institute of Technology

§Neuroscience Department, Erasmus Medical Center Rotterdam

***Abstract*—The field of computational neuroscience, in its efforts to reveal details of neuron operation, has been developing a demand for biophysically meaningful neural models. The increasing complexity of said models and the need for large-scale or real-time experiments have presented significant challenges to the world of high-performance computing (HPC). We explore Intel's newest generation of Xeon Phi computing platforms, named Knights Landing (KNL), as a way to match the need for processing power and as an upgrade over the previous generation of Xeon Phi models, the Knights Corner (KNC). Our analysis is done using a simulator, which implements a state-of-the art physiologically plausible model of the inferior-olive nucleus (InfOli), that has been ported on both generations of Xeon Phi platforms. The application uses the OpenMP interface for parallelization and the available vectorization buffers present in Xeon Phi platforms for Single-Instruction Multiple Data (SIMD) processing. In this analysis we provide insight as to how efficiently the application takes advantage of both Xeon Phi architectures and how the KNL measures against its predecessor. An out-of-the-box porting of the application onto Knights Landing results in our case, on an average 2.4× speedup with a 48% less energy consumption than KNC.**

***Keywords*—Intel Xeon Phi, Knights Landing, Computational Neuroscience**

I. Introduction

In recent years neuroscientists have been gradually revealing details of neuron operation. Using this knowledge, there is a wide research interest in studying the behaviour of single-neuron, a network of neurons and eventually study brain-wide populations of neurons. Simulating these neuronal networks on various platforms is an active field of research [1], [2].

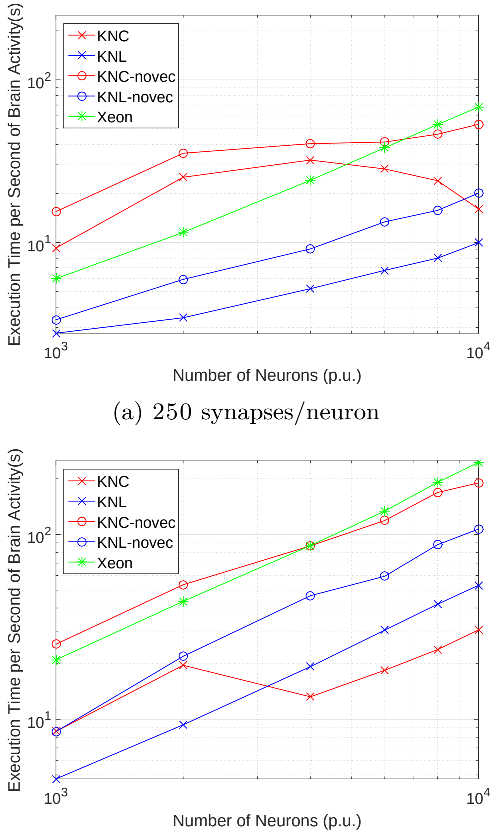
In our current comparative study we feature a simulator for biophysically plausible neuron models, targeting a part of the human brain named the Inferior Olivary Nucleus, which specializes in the coordination and learning of motor function [3]. The modeling accuracy is at the cell conductance level (Hodgkin and Huxley models [4]), belonging at an analytical and complicated class of models which allow us to expose fine details of the neuron's mechanisms. This workload is an excellent candidate for parallelization on HPC architectures, such as the Intel Xeon Phi system [5], due to the large inherent parallelism of the models. Additionally, it constitutes a realistic worst-case scenario in terms of model complexity, hence a benchmark for neuron modeling workloads.

In order to explore whether Intel's newest generation of the Xeon Phi computing platform, named Knights Landing (KNL), is a suitable platform for neuroscientific workloads, in the current paper we evaluate its performance and energy consumption compared to the previous version, Knights Corner (KNC). We utilize the aforementioned Inferior Olivary Nucleus simulator, named InfOli, which was developed for the KNC generation of Xeon Phi [6]. This comparison will highlight how the evolution of Intel's Xeon Phi architecture can improve the performance of a challenging application in the field of computational neuroscience. Since the application is fine-tuned to the previous version of Xeon Phi processors, we will, accordingly, explore the behaviour of an "out-of-the-box" application on the KNL.

II. Evaluation

The InfOli simulator is a transient simulator; brain activity is calculated in simulation steps, with each step set to represent 50us of activity in a fixed manner. The steps are calculated sequentially, until the entirety of the requested brain activity is computed. In order to boost simulation speed, OpenMP [7] has been employed to parallelize the application. The network is divided in equal parts and assigned to different OpenMP threads, ensuring a balanced distribution of workload.

The measurements presented in this section have been carried out using two different generations of Intel Xeon Phi. The Knights Corner co-processor's model is 3120P, featuring 57 cores at 1.1GHz, each supporting up to 4 threads running concurrently via multithreading technology. Cores run at 300W thermal design power (TDP). The Knights Landing processor's model is 7210, with 64 cores at 1.3GHz and similar multithreading capacities. Its TDP is noticeably lower at 215W. MCDRAM for the KNL was set to cache mode as this setting is completely transparent to software and allows for "out-of-the-box" codes like the neuron simulator being tested, to take advantage of the high-bandwidth-memory technology. As for the clustering mode, quadrant configuration was chosen based on recognition that the cache-quadrant combination offers performance gain to HPC applications [8], [9]. We include performance curves from an Intel Xeon E5-2609-v2, a 4-core server-grade processor utilizing 4 threads concurrently. The processor's simulation speed acts as a baseline.

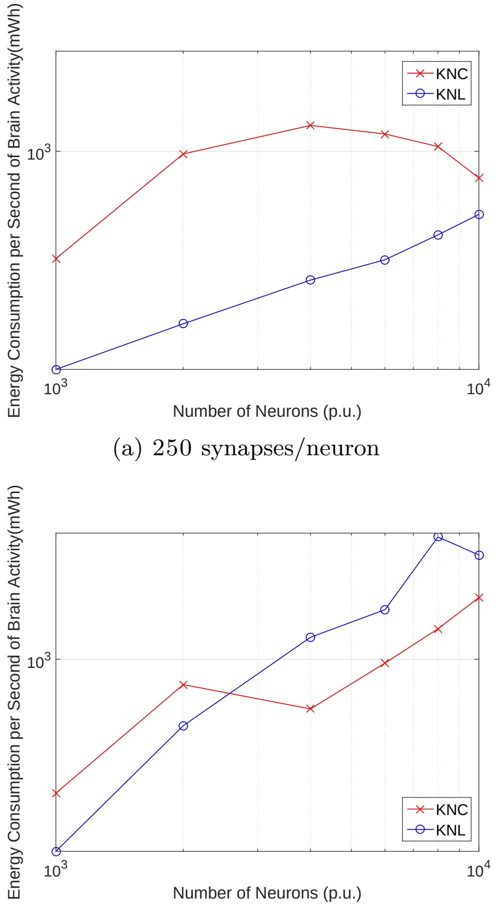


(a) 250 synapses/neuron

(b) 1000 synapses/neuron

Figure 1: Execution Time per second of simulated brain activity, comparison between Knights Corner (KNC) and Knights Landing (KNL) on different Simulator configurations. Performance on Xeon processor (4 threads) added as a baseline.

All experiments in Figure 1 have been carried out using approximately the maximum amount of threads available to each platform. For the KNC, we used 220 threads, whereas the KNL offered 256 threads. On average the KNL platform outperforms the KNC platform by 2.4× in terms of execution time. The maximum speed-up is 6×, while in some cases the KNC comes in front with up to 1.6× speed up over the KNL. More specifically, we can observe that, in the cases of low connectivity density (Figure 1a), which translates to a low amounts of workload per thread, the KNL shows a superior performance to the KNC. On the other hand, as the computational workload assigned to each thread increases for denser networks, the KNC performs significantly better (Figure 1b). The performance gap between the two platforms lessens as the KNC can use its assets with increasing efficiency, since the application has been optimized with the KNC architecture in mind.



(a) 250 synapses/neuron

(b) 1000 synapses/neuron

Figure 2: Execution Time per second of simulated brain activity, comparison between Knights Corner (KNC) and Knights Landing (KNL) on different Simulator configurations. Performance on Xeon processor (4 threads) added as a baseline.

In Figure 2, we present information regarding the energy required by each computing fabric in order to simulate a second of brain activity, measured in mWh. The Figure is directly linked to Figure 1, since energy consumption is dependent on execution time needed for simulation of each second of brain activity. As such, we can observe similar patterns between the two Figures. On average we have to note that the KNL consumes 48% less energy than the KNC. Because of the KNL's lower TDP and better performance for light workloads, there is a significant reduction in energy consumption when computing for small networks. To put this claim into perspective, whereas the simulation of one second of brain activity in a network of 4000 neurons, with a density of 250 synapses per neuron (Figure 2), requires over $1200mWh$ for the KNC, the KNL consumes under $300mWh$ for the same workload, improving on energy efficiency by a factor of 4×.

On the contrary, due to the KNC's smaller execution times for larger, denser networks, it is preferable from a power consumption standpoint to the KNL for such workloads. A network of 10,000 neurons, each forming 1,000 synapses with the rest of the network, requires 27% less energy on the KNC ($1600mWh$ per second of simulated time) than on the KNL ($2200mWh$ for the same amount of activity).

In HPC, efficiency metrics offer insight as to how well an application utilizes the underlying platform's resources. In our case, we calculated the efficiency metric by dividing execution speedup with the number of OpenMP threads spawned, with a range of OpenMP threads utilized from 1 to 200, on both platforms. For the KNL, we observed that the efficiency of utilizing up to approximately 50 threads remains at satisfactory levels. In these cases, each core spawns either one or two threads (due to the selected balanced thread affinity) and, in contrast to the KNC, the KNL's cores operate significantly better when operating with only one thread [8].

On larger networks, however, KNC offer better opportunities to utilize its computational assets efficiently, maintaining a speedup-to-threads ratio above 70% even for 200 threads. The KNL's threading efficiency sharply declines when employing massive degrees of parallelism, dropping below 40% when using more than 140 threads. The application's inability to utilize the entirety of KNL's assets efficiently to tackle demanding simulations explains the performance gap between the two platforms for larger workloads. This inability is mostly attributed to the fact that the simulator has been fine-tuned to the KNC environment and has been tested "out-of-the-box" on the KNL.

III. Conclusion

In this evaluation, a computationally demanding application from the field of computational neuroscience that had previously been extensively developed and optimized for the Intel KNC, has been tested "out-of-the-box" for the second generation of Xeon Phi, the KNL. The InfOli biophysically-accurate simulator's performance was tested using a range of workloads, from small, unconnected neuronal populations to larger, dense networks. The results were evaluated from both a simulation-speed and a power-efficiency standpoint. On average KNL offers a speed up of 2.4× while consuming 48% less energy. Smaller workloads, by taking advantage of the KNL's superior single-threaded performance, exhibit very significant gains in both speed and, even more so, energy consumption, with specific experiments demanding 75% less Wh of energy per second of simulated brain activity on the KNL. On the other hand, without further fine-tuning of the application to the architectural details of the KNL, OpenMP-thread efficiency suffers when running on the KNL, causing the simulator to handle more demanding networks poorly, relatively to the optimized KNC version. Furthermore, throughout the whole range of experiments, it has been shown that the KNL offers a more robust, dependable performance curve with little variability.

These findings are promising enough to warrant further optimization of the simulator for the new generation of the Xeon Phi. As future work, we would suggest using an optimized version of the simulator on a cluster of KNL processors, in order to simulate neuronal networks of much larger sizes and take advantage of Intel's OmniPath technology for inter-node communication [10].

Acknowledgment

This work is partially supported by European Commission project H2020–687628–VINEYARD.

References

[1] Bhuiyan, M. et al., "Acceleration of spiking neural networks in emerging multi-core and gpu architectures," in *IPDPSW*, 2010.

[2] Nguyen, H. A. Du et al., "Accelerating complex brain-model simulations on gpu platforms," in *DATE*, 2015, pp. 974–979.

[3] De Zeeuw, C. I. et al., "Microcircuitry and function of the inferior olive," *Trends in neurosciences*, vol. 21, no. 9, pp. 391–400, 1998.

[4] A. L. Hodgkin and A. F. Huxley, "Propagation of electrical signals along giant nerve fibres," *Proceedings of the Royal Society of London. Series B, Biological Sciences*, vol. 140, no. 899, pp. 177–183, 1952.

[5] Fang, J. et al., "Test-driving intel xeon phi," in *ICPE*, 2014.

[6] G. Chatzikonstantis, D. Rodopoulos, C. Strydis, C. I. De Zeeuw, and D. Soudris, "Optimizing extended hodgkin-huxley neuron model simulations for a xeon/xeon phi node," *IEEE Transactions on Parallel and Distributed Systems*, 2017.

[7] L. Dagum and R. Enon, "Openmp: an industry standard api for shared-memory programming," *IEEE CSE*, vol. 5, no. 1, pp. 46–55.

[8] J. Jeffers, J. Reinders, and A. Sodani, *Intel Xeon Phi Processor High Performance Programming: Knights Landing Edition*. Morgan Kaufmann, 2016.

[9] C. Rosales, D. James, A. Gómez-Iglesias, J. Cazes, L. Huang, H. Liu, S. Liu, and W. Barth, "TACC Technical Report TR-16-03 KNL Utilization Guidelines," University of Texas at Austin, Texas Advanced Computing Center, Tech. Rep., November 2016. [Online]. Available: https://portal.tacc.utexas.edu/documents/10157/1334612/KNL+Utilization+Guidelines/95cc0f23-1755-424d-8d29-64a91a09cf33

[10] M. S. Birrittella, M. Debbage, R. Huggahalli, J. Kunz, T. Lovett, T. Rimmer, K. D. Underwood, and R. C. Zak, "Intel® omni-path architecture: Enabling scalable, high performance fabrics," in *High-Performance Interconnects (HOTI), 2015 IEEE 23rd Annual Symposium on*. IEEE, 2015, pp. 1–9.