SAND2022-2297C

Benchmarking a Bio-inspired SNN on a Neuromorphic System

Luke Parker
lgparke@sandia.gov
Sandia National Laboratories
Albuquerque, New Mexico, USA

Frances S. Chance
fschanc@sandia.gov
Sandia National Laboratories
Albuquerque, New Mexico, USA

Suma G. Cardwell
sgcardw@sandia.gov
Sandia National Laboratories
Albuquerque, New Mexico, USA

ACM Reference Format:
Luke Parker, Frances S. Chance, and Suma G. Cardwell. 2022. Benchmarking a Bio-inspired SNN on a Neuromorphic System. In *Neuro-Inspired Computational Elements Conference (NICE 2022), March 28-April 1, 2022, Virtual Event, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3517343.3517365

INTRODUCTION

Neuromorphic computing aims to derive the benefit of computational dynamics observed in biological neural systems. This has led to novel non-Von Neumann device architectures and algorithms that operate in the spiking domain. Large, densely-connected neural networks on a traditional device consume more power than their biological counterparts, especially when accounting for both training and inference. Neuromorphic devices present an opportunity to place these large, densely-connected networks on architectures that more closely resemble biological neural systems. By placing these networks in silicon, researchers in both engineering and neuroscience can better understand the cost and constraints of using neural dynamics for computation in a synthetic system.

Performing computation with spikes is one of the major differences between artificial neural networks (ANNs) and spiking neural networks (SNNs). This difference is most pronounced when using a computing architecture designed to handle spiking computation such as a neuromorphic device. By implementing a SNN on both a traditional Von Neumann device, like a CPU, and a neuromorphic device, the benefits of using a native spiking architecture can be compared and analyzed. *To that end, this work introduces initial findings in comparing the computational efficiency between a traditional and neuromorphic platform when implementing a bio-inspired SNN. [3] These findings contribute to the growing body of benchmark literature that highlight the performance benefits of using neuromorphic devices for bio-inspired neural network designs.*

The bio-inspired SNN is a simplified, spiking version of the neural network found in [3], and consists of two densely-connected layers as shown in Figure 1. The network is driven by stimulating the input layer to cause network activity, and the weights of the network are fixed as no learning occurs in the model.

The neuromorphic platform chosen for this work is Intel's Loihi, a neuromorphic chip supporting 128,000 neurons and 128 million synapses on a single chip. [4] The computation is digital, but executes using spiking dynamics inspired by neural systems. The SNN is executed using this device, and the results are compared to the SNN implementation on an Intel Xeon W-2123 processor, the choice for the traditional computing platform.

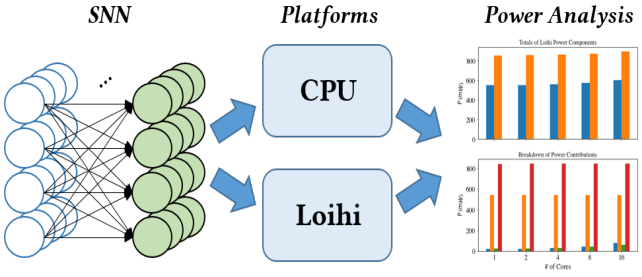

Figure 1: Comparing performance of a densely-connected SNN when implemented on both traditional and neuromorphic platforms.

The SNN is implemented on Loihi using single-compartment neuron primitives for each layer with all-to-all connectivity. The weights are converted to 8-bit precision due to memory constraints. The first layer is stimulated by using a bias parameter that causes its neurons to accumulate charge until a certain threshold. This approach simulates external input stimulating the first layer's neurons.

The custom CPU implementation consists of a matrix representing the spiking state of the first-layer neurons due to an external stimulus. As such, each element of the matrix is 0 or 1 at timestep t, where 1 indicates a spike to be transmitted. The weights defining the connection strengths between the first and second layers use floating-point precision. When a neuron in the first layer transmits a spike, the connected neurons of the second layer accumulate charge proportional to the associated connection weights.

EXPERIMENTAL SETUP

Different software tools were used to gather profiling numbers on power consumption, energy consumption, and execution time of the SNN on both Loihi and the CPU. For Loihi, the built-in energy and execution time probes of Intel's NxSDK 0.9 API were used, providing estimates for power consumption and execution time directly from the chip. For the CPU, the package pyJoules [6] was used, which acts as a Python wrapper for Intel's RAPL technology, providing estimates for CPU package and DRAM power consumption along with the execution time of decorated Python functions.

Experiments were set up to gather data for both Loihi and CPU by writing Python scripts that collected measurements of each device while executing the SNN under two different conditions: *Spikes* and *No Spikes*. In the *Spikes* condition, the neurons of the first layer of the SNN fire simultaneously at each timestep, whereas

NICE 2022, March 28-April 1, 2022, Virtual Event, USA

ACM ISBN 978-1-4503-9559-5/22/03.
https://doi.org/10.1145/3517343.3517365

in the *No Spikes* condition, no stimulus is present in the network. These two cases portray maximum network activity and minimum network activity, respectively.

Each experiment is characterized as a *workload*, which is defined as a sweep over network sizes consisting of N^2 neurons in both network layers for $N = 1, 2, 3, ..., 30$, so each network consists of a total of $2N^2$ neurons. Workloads for both Loihi and CPU were executed several times consecutively to collect enough samples for statistical integrity. The power, energy and execution time data is provided by averaging the sample measurements post-execution.

Loihi Experiment

The NxSDK energy probes used for recording Loihi data return measurements from the board's host CPU, which gathers power consumption data using an onboard sensor. This data is recorded as a function of the host CPU's clock and is reconciled with the timestamps associated with the asynchronous algorithmic timesteps of the SNN's execution. An algorithmic timestep consists of of different "phases" pertaining to spike transmision, learning rule calculations, state updates, and optional embedded code used to interact with the SNN.

The energy and execution time of these different phases can be captured by the probe to observe what processes or components most impact the total power consumption. An example is shown in the top plot of Figure 2, where the total Core power consumption (neurocores, embedded processor, mesh router, etc.) and SRAM power consumption (memories in the embedded processor and neurocores) are plotted for a single group of neurons that increases in size on a per core basis. The bottom plot shows a breakdown of each power measurement for two types of contributions: neurocore updates and background activity. The power consumed by neurocore updates represents the cost of processing and transmitting spikes in the network, whereas the background activity is the remaining amount of power consumed by the board.

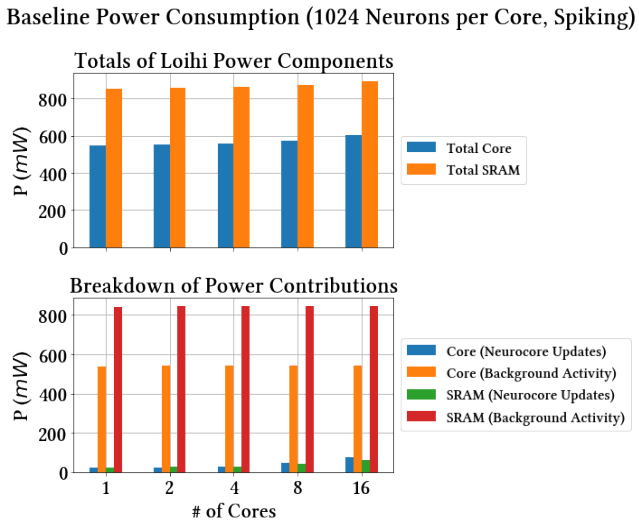

Figure 2: Breakdown of Power Components from an Example Group of Spiking Neurons on Loihi.

In Figure 2, the example group of neurons is internally stimulated using a bias parameter to spike on each timestep. These data are compared to the same group of neurons exhibiting no spiking activity in Figure 3, which indicates that spiking activity for a group of neurons does not significantly impact the chip's measured power consumption.

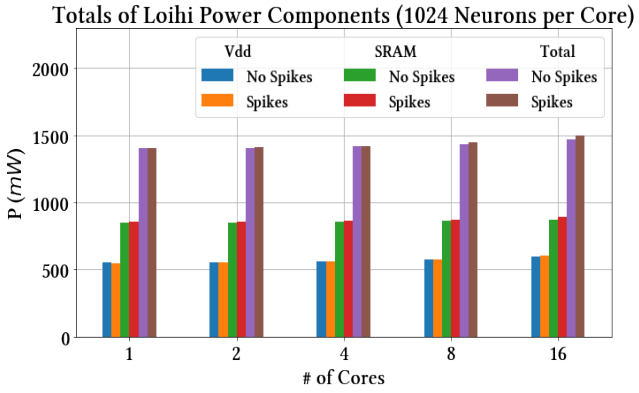

Figure 3: Stimulus of the Example Neuron Group does not significantly impact Power Consumption for smaller SNNs.

When executing the SNN, the Core and SRAM power contributions were recorded for each network size. Although the contribution of the background activity (the largest consumer of power per Figure 2) are considered in the totals used in comparing the CPU data, the contribution of the neurocores' activity was of particular interest.

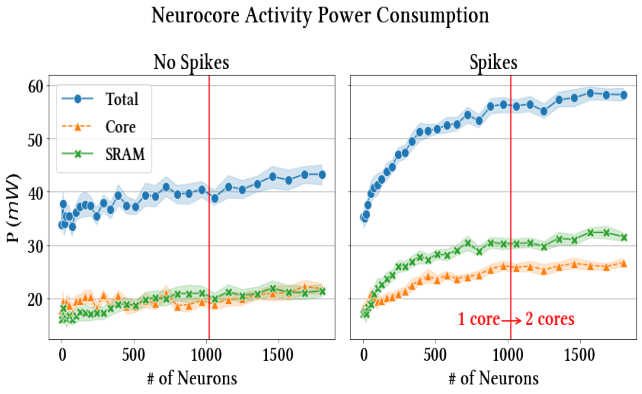

Figure 4: Power consumed by neurocores on Loihi as SNN increases in size.

Shown in Figure 4 is comparison of the neurocore power consumption under no-spiking and spiking conditions. As the number of neurons in the SNN increase, the number of required neuron-state updates increases, even without spikes needing to be processed (shown by the left plot in Figure 4). However, when the first layer has spikes to transmit, the densely-connected SNN consumes additional power due to increased neurocore activity.

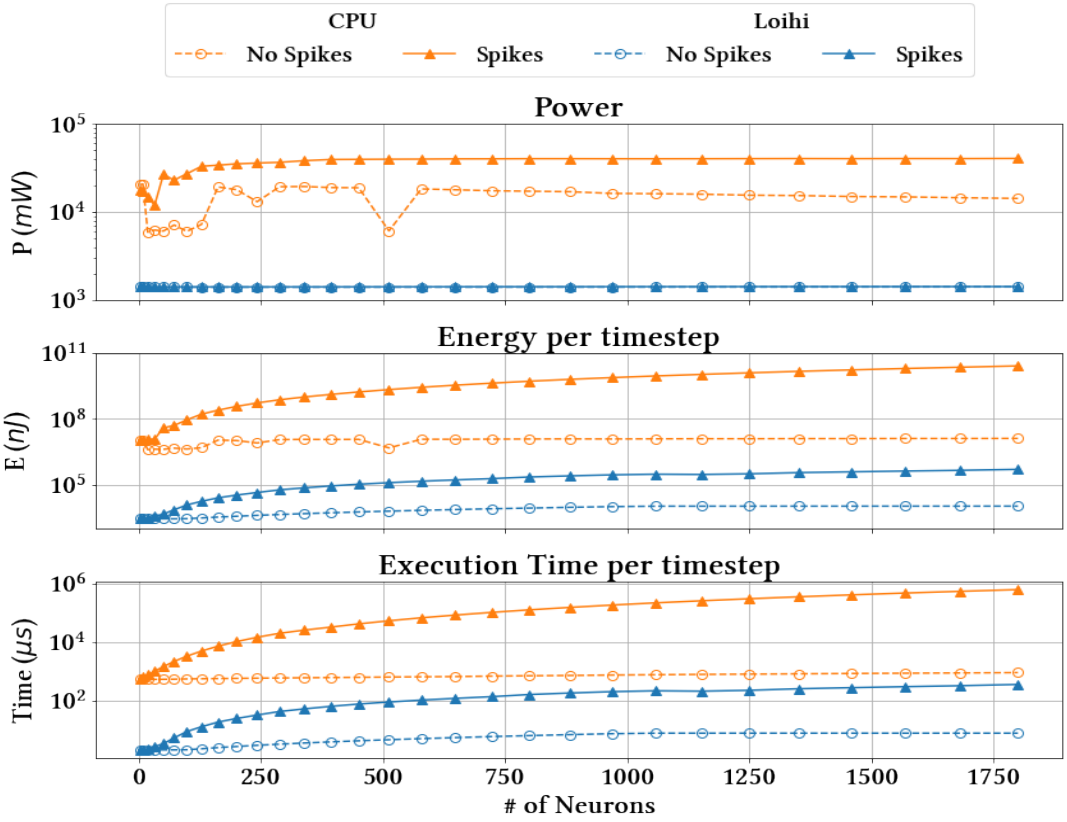

Figure 5: Comparison between Power, Energy, and Execution Time of SNN on both CPU and Loihi.

CPU Experiment

For the CPU, initial RAPL measurements were recorded for a short period of time before executing the workload script. These initial numbers provided power consumption in the CPU's "idle" state that account for the contribution of system background processes that are unrelated to the SNN's power consumption. Data from the entire CPU package and DRAM were recorded during each workload sweep, and the units of each measurement were scaled to match the units of the Loihi data.

RESULTS

Shown in Figure 5 is a comparison of the SNN's performance between Loihi and CPU implementations. For Loihi measurements, we used the total power, energy, and execution time per timestep.

Noting the logarithmic y-axes used in Figure 5, the Loihi implementation exhibits better performance in both *No Spikes* and *Spikes* conditions in energy used and execution time. Regarding energy per timestep, the average difference between the Loihi and CPU networks is on the order of 10^3 and 10^4 for *No Spikes* and *Spikes*, respectively. As for execution time, the differences are on the order of 10^2 and 10^3. Also, the average difference in power used in the Loihi network for both conditions is under 10 mW, which is significantly lower than the difference for the CPU network, indicated by the top plot of Figure 5.

Overall, these results provide additional evidence to the growing body of benchmarking literature for neuromorphic platforms [1] [2] [5] [7] [8], demonstrating that a device like Loihi can execute network calculations faster and with greater efficiency than a traditional computing device.

DISCUSSION

This brief inspection of a neuromorphic architecture's computational performance promotes the plausibility of executing bio-inspired neural networks, and more importantly highlights some of the advantages of executing such networks on alternative computing platforms. On Loihi, the energy consumed per timestep and execution time of the network are order of magnitudes smaller than the custom CPU implementation, even as the network size increases. Furthermore, the power consumed by Loihi is not significantly impacted by the SNN's spiking activity. This makes a neuromorphic device more appealing than a traditional processors for applications requiring low size, weight, and power constraints.

Scale and complexity are considerations left for future work, which will involve more complex neuron models, more advanced implementations on each platform, and larger, distributed network topologies. Also of interest are comparisons of Loihi with an optimized version of the custom CPU implementation and a custom SNN made for deployment on a mobile GPU device.

ACKNOWLEDGEMENTS

This work was supported by Laboratory Directed Research and Development program at Sandia National Laboratories. Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

This paper describes technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. SAND Number: SAND202X-XXX

REFERENCES

[1] Peter Blouw, Xuan Choo, Eric Hunsberger, and Chris Eliasmith. 2019. Benchmarking Keyword Spotting Efficiency on Neuromorphic Hardware. In *Proceedings of the 7th Annual Neuro-Inspired Computational Elements Workshop* (Albany, NY, USA) *(NICE '19)*. Association for Computing Machinery, New York, NY, USA, Article 1, 8 pages. https://doi.org/10.1145/3320288.3320304

[2] Peter Blouw and Chris Eliasmith. 2020. Event-Driven Signal Processing with Neuromorphic Computing Systems. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 8534–8538. https://doi.org/10.1109/ICASSP40776.2020.9053043

[3] Frances S. Chance. 2020. Interception from a Dragonfly Neural Network Model. In *International Conference on Neuromorphic Systems 2020* (Oak Ridge, TN, USA) *(ICONS 2020)*. Association for Computing Machinery, New York, NY, USA, Article 21, 5 pages. https://doi.org/10.1145/3407197.3407218

[4] Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, Yuyun Liao, Chit-Kwan Lin, Andrew Lines, Ruokun Liu, Deepak Mathaikutty, Steven McCoy, Arnab Paul, Jonathan Tse, Guruguhanathan Venkataramanan, Yi-Hsin Weng, Andreas Wild, Yoonseok Yang, and Hong Wang. 2018. Loihi: A Neuromorphic Manycore Processor with On-Chip Learning. *IEEE Micro* 38, 1 (2018), 82–99. https://doi.org/10.1109/MM.2018.112130359

[5] Mike Davies, Andreas Wild, Garrick Orchard, Yulia Sandamirskaya, Gabriel A. Fonseca Guerra, Prasad Joshi, Philipp Plank, and Sumedh R. Risbud. 2021. Advancing Neuromorphic Computing With Loihi: A Survey of Results and Outlook. *Proc. IEEE* 109, 5 (2021), 911–934. https://doi.org/10.1109/JPROC.2021.3067593

[6] Spirals Research Group. 2021. *pyJoules: A Python library to captuer the energy consumption of code snippets*. University of Lille and Inria. https://github.com/powerapi-ng/pyJoules

[7] Garrick Orchard, E. Paxon Frady, Daniel Ben Dayan Rubin, Sophia Sanborn, Sumit Bam Shrestha, Friedrich T. Sommer, and Mike Davies. 2021. Efficient Neuromorphic Signal Processing with Loihi 2. In *2021 IEEE Workshop on Signal Processing Systems (SiPS)*. 254–259. https://doi.org/10.1109/SiPS52927.2021.00053

[8] Guangzhi Tang, Neelesh Kumar, and Konstantinos P. Michmizos. 2020. Reinforcement co-Learning of Deep and Spiking Neural Networks for Energy-Efficient Mapless Navigation with Neuromorphic Hardware. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 6090–6097. https://doi.org/10.1109/IROS45743.2020.9340948