Creating a Pipeline to Graphically Design and Execute Spiking Network Algorithms*

Niels van Harten[1], Arne Diehl[2,3][0000−0001−9702−1083], and Johan Kwisthout[2,3][0000−0003−4383−7786]

[1] Radboud University
[2] Donders Institute for Brain, Cognition, and Behaviour
arne.diehl@donders.ru.nl

Abstract. Contemporary graphical tools for constructing spiking neural networks (SNNs) primarily focus on groups of neurons rather than individual neurons, which overlooks the need for finer control required for implementing spiking algorithms in SNNs. Such hand-crafted SNNs excel in performing precise computations and may hold great potential as energy-efficient co-processors for various computational applications. However, the lack of graphical tools for precise SNN construction acts as a barrier to entry into the field of neuromorphic computing, as new developers may benefit from visualisation for building intuition in crafting SNNs.

Keywords: Neuromorphic Computing · Spiking Neural Networks · Graphical Modelling Tool · Educational Technology

1 Introduction

This work introduces a novel graphical tool that facilitates the construction of SNNs consisting of discrete-timed individual neurons and synapses as in the Sandia Model [3]. The tool enables execution on both conventional and neuromorphic hardware. For conventional hardware, we utilise the "simsnn" [1] Python package, while for neuromorphic hardware, we present a unique proof-of-concept compiler that enables network translation and execution on the Loihi 2 architecture [2].

2 Key Findings

The developed software solution successfully addresses the identified requirements. The graphical interface provides the necessary fine-grained control over individual neurons and synapses, making it a valuable resource for developers in the field of neuromorphic computing. Additionally, the compiler demonstrates the capability to translate a substantial subset of networks for efficient execution on the Loihi 2 hardware.

* Supported by Intel.

3 Educational Use and Demonstration

During the demonstration (see video attached)[3], we showcased the functionality of the modelling tool developed in this research. The tool offers a graphical user interface (GUI) accessible via web browsers and a back end server to facilitate the design and execution of spiking network algorithms. The demonstration illustrated the following key features:

Network Construction: Users can construct spiking networks comprising individual neurons and synapses with weight and delay parameters using the intuitive GUI.
Real-time Visualisation: The tool provides real-time visualisation of network behaviour, including voltage plots and spiking patterns, enabling users to observe the network's operation.
Export and Import: Users can export created networks for sharing with others and import pre-designed networks for further experimentation.
Configurable Features: The tool offers optional features, such as specifying random generator seeds, which can enhance reproducibility and experimentation.

4 Conclusion and Future Directions

In conclusion, this research provides a valuable contribution to the field of neuromorphic computing by introducing a graphical modelling tool and a proof-of-concept compiler for executing spiking network algorithms. While the tool and compiler exhibit promising capabilities, there are opportunities for future work, such as expanding the tool's modular components and addressing limitations in the compiler to support more complex networks. Additionally, comparative analysis with other neuromorphic platforms may further enhance our understanding of their suitability for specific problem domains.

5 System Requirements

To access and utilise the modeling tool and compiler, users require a standard web browser for the GUI interface. The back end server operates seamlessly, and no additional dependencies are needed. Detailed system requirements and access information are available upon request.

6 Economic and Societal Impact

This research not only contributes to advancing the field of neuromorphic computing but also holds the potential for significant economic and societal impact. By providing accessible tools for spiking neural network design and execution, we empower researchers, developers, and educators to explore the capabilities of neuromorphic computing, fostering innovation and knowledge dissemination.

[3] https://drive.google.com/file/d/1NsaX-EEGt9is4vKduldzQF5XtTmLzpu7/view?usp=sharing

References

1. Chateau-Laurent, H., Diehl, A., Kwisthout, J.: simsnn, version 1.0.2 (2019). https://gitlab.socsci.ru.nl/neuromorphiccomputing/simsnn.
2. Orchard, G., Frady, E.P., Rubin, D.B.D., Sanborn, S., Shrestha, S.B., Sommer, F.T., Davies, M.: Efficient Neuromorphic Signal Processing with Loihi 2. In: 2021 IEEE Workshop on Signal Processing Systems (SiPS), pp. 254–259 (2021). https://doi.org/10.1109/SiPS52927.2021.00053
3. Severa, W., Parekh, O., Carlson, K.D., James, C.D., Aimone, J.B.: Spiking Network Algorithms for Scientific Computing. In: 2016 IEEE International Conference on Rebooting Computing (ICRC), pp. 1–8. IEEE Computer Society (2016). https://doi.org/10.1109/ICRC.2016.7738681. https://www.computer.org/csdl/proceedings-article/icrc/2016/07738681/12OmNqFJhXg (visited on 03/21/2023)