NENS Exchange Grant - Report

Lukas Braun
mail@lukasbraun.com, www.lukasbraun.com

Host Institution: Vogels Lab, University of Oxford
Home Institution: Bernstein Center for Computational Neuroscience Berlin

Training Period: March and April 2020

1 Introduction

Nervous systems encode and process sensory information, memories and motor outputs with spiking neural networks. However, building functional artificial networks with biologically plausible dynamics is an ongoing challenge. While there are algorithms that improve connectivity in artificial networks (Bengio, Léonard, & Courville, 2013; LeCun, Bengio, & Hinton, 2015; Song, Miller, & Abbott, 2000; Vogels, Sprekeler, Zenke, Clopath, & Gerstner, 2011), they are applied to rate models or to networks of neurons with homogeneous properties. However, rate models do not explain how dynamics arise from individual neurons and further, it remains unclear, how plausible network dynamics can be generated by only changing the connectivity in networks of homogeneous neural computations. While there are computationally efficient models that can generate a vast variety of neurocomputational properties (Izhikevich, 2004), it remains an open question, how they can be used as faithful models for biological phenomena.

2 Aim of training stay

The aim of my training stay at the Vogels Lab was to gain an understanding of the methods that could be used to extend surrogate gradients (Neftci, Mostafa, & Zenke, 2019; Zenke & Ganguli, 2018) to optimise the synaptic connectivity and neural computations performed by single resonate-and-fire neurons (Izhikevich, 2001) to networks of neurons.

3 Training stay

Extending gradient-based learning to multi-layer networks of spiking-neurons is non-trivial since the real gradient is ill defined due to the non-differentiable spiking threshold. The surrogate gradient method proposes to backpropagate errors directly through the weight matrix without propagating errors through the surrogate or synaptic activities. To show how reliable this gradient approximation is, I designed a teacher-student paradigm (Biehl & Schwarze, 1995), in which I tried to recover the parameters of the teacher network by optimising the parameters of a student network. By using this paradigm, I knew the exact solution of the learning problem and could thus investigate the performance of a learning algorithm.

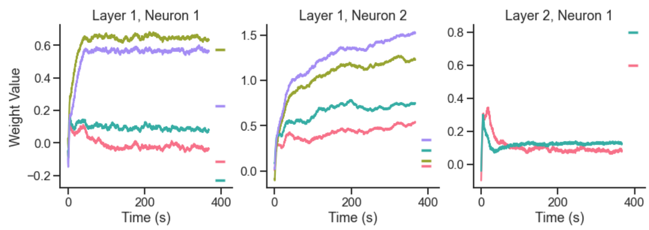

Figure 1: The approximations of the real gradient suggested by the surrogate gradient method fail to reproduce the parameters of a multi-layer teacher network in a student network with two hidden neurons and one output neuron. Target weight values are plotted at the end of each plot as horizontal lines.

During my time, I also learned and reproduced methods to tackle the so called weight transport problem, which is how the backward pass in a gradient update step can be implemented in a biologically plausible fashion Akrout, Wilson, Humphreys, Lillicrap, and Tweed (2019); Lillicrap, Cownden, Tweed, and Akerman (2016); Lillicrap, Santoro, Marris, Akerman, and Hinton (2020).

Finally, I learned how to port my code to Jax (Bradbury et al., 2018). Jax is an automatic differentiation library with functionality to automatically vectorise operations and with parallel programming capabilities to run multiple simulations simultaneously on multiple GPUs. Also, I have vastly improved the logging and visualization capabilities of my code.

Besides learning about different implementations of training multi-layer neural networks, I was actively involved in lab activities. That is: Attending and preparing journal clubs, lab meetings and casual get-togethers. Three weeks after my training stay started, the United Kingdom did go into lockdown due to the COVID-19 pandemic. So, unfortunately, a large part of my training stay had to be moved online. However, thanks to the theoretical nature of the lab's work, we were able to move large parts of the lab routines and accompanying social activities online. My supervisor and colleagues were very helpful and we supported each other through this unusual times, making my training stay a full success.

4 Outlook

The preliminary results of my training stay are the starting point for my Master's thesis. Now being equipped with fast and reliable code to simulate and analyse deep spiking neural networks, I will continue investigating how gradient-based learning can be used to optimise computations performed by individual neurons and compare multiple approaches that extend training to multi-layer networks. Thus, my NENS Exchange Training Stay was key to require the understanding and tools required for future work.

I am going to present my results at the conference "From Neuroscience to Artificially Intelligent Systems" (which has been postponed to November 2020 due to COVID-19) in form of a poster.

5 Acknowledgments

I would like to thank the Network of European Neruoscience Schools for providing me with this stipend and by doing so, enabling me to have this insightful training stay. Especially, I would like to thank Professor Tim Vogels for his kind, patient and thoughtful supervision and for sharing his expertise. And last but not least, I would like to thank the Vogels Lab, which, despite the start of an unforeseeable pandemic, provided me with a very warm, collaborative and supportive learning environment.

References

Akrout, M., Wilson, C., Humphreys, P., Lillicrap, T., & Tweed, D. B. (2019). Deep learning without weight transport. In *Advances in neural information processing systems* (pp. 974–982).

Bengio, Y., Léonard, N., & Courville, A. (2013). Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*.

Biehl, M., & Schwarze, H. (1995). Learning by online gradient descent. *Journal of Physics A*, *28*, 643–656.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., & Wanderman-Milne, S. (2018). *JAX: composable transformations of Python+NumPy programs.* Retrieved from http://github.com/google/jax

Izhikevich, E. M. (2001). Resonate-and-fire neurons. *Neural networks*, *14*(6-7), 883–894.

Izhikevich, E. M. (2004). Which model to use for cortical spiking neurons? *IEEE transactions on neural networks*, *15*(5), 1063–1070.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016). Random synaptic feedback weights support error backpropagation for deep learning. *Nature communications*, *7*(1), 1–10.

Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., & Hinton, G. (2020). Backpropagation and the brain. *Nature Reviews Neuroscience*, 1–12.

Neftci, E. O., Mostafa, H., & Zenke, F. (2019). Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, *36*(6), 51–63.

Song, S., Miller, K. D., & Abbott, L. F. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature neuroscience*, *3*(9), 919–926.

Vogels, T. P., Sprekeler, H., Zenke, F., Clopath, C., & Gerstner, W. (2011). Inhibitory plasticity balances excitation and inhibition in sensory pathways and memory networks. *Science*, *334*(6062), 1569–1573.

Zenke, F., & Ganguli, S. (2018). Superspike: Supervised learning in multilayer spiking neural networks. *Neural computation*, *30*(6), 1514–1541.