Implementation of a spiking NN producing polychronous patterns in Python

1 Description

The study of spike-timing dynamics in the brain is of interest to neuroscientists and the artificial network community. Investigation of the relative timing of spikes of multiple neurons [1] and the role they play in temporal coding in the brain is an important issue issue. Similarly, spiking networks can serve as powerful supervised learning and memory representation tools with several potential applications in the machine learning domain [2].

Spiking network polychronization refers to reproducible time-locked but not synchronous firing patterns with millisecond precision. It was originally studied by Izhikevich [3] in a model of cortical spiking neurons with axonal conduction delays and spike-timing plasticity. It has been proved that conduction delays play a role in the computational power and learnability of spiking neural networks (SNNs)[2].

2 Objectives

The goal of the project is to implement the spiking NN model by Izhikevich in Python. As in other projects, a report should describe the characteristics of the design, implementation, and results. A Jupyter notebook should include calls to the implemented function that illustrate the way it works.

3 Suggestions

- There are available implementations of the model in C++ and Matlab `https://www.izhikevich.org/publications/spnet.htm` and a number of very recent papers that implement or propose applications of Izhikevich's model [4, 5, 6, 7].
- Implementations can use any Python library.
- Visualization of the network spiking generation process is encouraged as an additional, but not required, delivery of the project.

References

[1] E. M. Izhikevich and G. M. Edelman. Large-scale model of mammalian thalamocortical systems. *Proceedings of the National Academy of Sciences (PNAS)*, 105(9):3593–3598, 2008.

[2] Hélène Paugam-Moisy, Régis Martinez, and Samy Bengio. Delay learning and polychronization for reservoir computing. *Neurocomputing*, 71(7-9):1143–1158, 2008.

[3] E. M. Izhikevich. Polychronization: Computation with spikes. *Neural Computation*, 18(2):245–282, 2006.

[4] Joseph Chrol-Cannon, Yaochu Jin, and André Grüning. An efficient method for online detection of polychronous patterns in spiking neural network. *CoRR*, abs/1702.05939, 2017.

[5] R. Santana, C. Bielza, and P. Larrañaga. Maximizing the number of polychronous groups in spiking networks. In *Companion Proceedings of the 2012 Genetic and Evolutionary Computation Conference GECCO-2012*, pages 1499–1500, Philadelphia, US, 2012. ACM Press.

[6] R. Santana, C. Bielza, and P. Larrañaga. Changing conduction delays to maximize the number of polychronous groups with an estimation of distribution algorithm. Technical Report UPM-FI/DIA/2013-1, Department of Artificial Intelligence, Faculty of Informatics, Technical University of Madrid, 2013.

[7] Runchun M Wang, Tara J Hamilton, Jonathan C Tapson, and André van Schaik. A mixed-signal implementation of a polychronous spiking neural network with delay adaptation. *Frontiers in neuroscience*, 8, 2014.