Analyzing the framework of “Reservoir Computing” for Hardware Implementation

Arfan Ghani, T.M. McGinnity, L. P Maguire, J.G. Harkin
Faculty of Engineering, Intelligent Systems Engineering Laboratory, University of Ulster,
Northland Road, Derry, BT487JL, N. Ireland, UK
{Ghani-a1@ulster.ac.uk}

Abstract

This paper investigates a framework of Liquid State Machine (LSM) and its applicability for recognition problems. Different learning algorithms were employed including linear regression, backpropagation, linear classification and parallel delta rule. In order to observe the computational power of the LSM, different parameters were changed including the degree of recurrence, size of reservoir, static/dynamic synapse, different spiking neuron models, and networks. This information is important in order to decide the best setup of this model to get a higher performance. In order to observe the behavior of the circuit, different noise levels were included in the network to observe the degradation in over all performance. In order to ensure that all experiments carried out are fair, statistical techniques are employed. For fair comparison between the different algorithms, the same randomly generated input is applied. Different liquid structures were used in order to evaluate the size of reservoir required to gain the reasonable performance. This comprehensive analysis is important because the goal of this research is to realize this paradigm on hardware in order to exploit the speedup with parallel networks and readouts.

1 Introduction

Liquid State Machine [1,2] is a novel framework that theoretically allows for real-time computation on continuous input streams in parallel. Randomly generated recurrent networks of neurons with realistic biological properties can be used for processing inputs. The most important aspect in this approach is that these networks do not have to be trained for specific tasks. Relatively simple techniques can be used to extract features from the liquid state of these networks. The readout units can be trained for temporal pattern recognition. The paradigm is biologically plausible, as parts of the cerebral cortex have been identified to perform sensory integration tasks in small and homogenous columns of neurons [4]. There is a resemblance in liquid state machine and finite state machine. However, in the finite state machine the full state space and transitions are clearly defined which is unlike the LSM. The liquid state machine could be seen as a universal finite state machine embedded with many complex sub finite state machines. The advantage of the LSM is that it is not required to define the states because the readout units can be trained to extract the state from the variable state of the liquid.

The dynamics in the medium (liquid) are of great importance, as it allows for information to be temporally stored and integrated. It is shown that randomly generated and connected networks with fixed weights perform really well. Biologically plausible neural networks are believed to excel at noisy temporal pattern recognition tasks therefore experiments have been set up to provide insight in the capabilities of this framework for the recognition of complex patterns in noisy input streams.

2 Discussion

The framework provides a promising approach for temporal pattern recognition on continuous input streams. Theoretically all sorts of complex systems can be applied but there is no clear indication about what systems would perform better and what parameters allow for the best information pre-processing. Despite the computational strength of this paradigm, the real complexity of these tasks is hard to estimate.

Performance of static and dynamic synapses were compared for all experiments and different networks with different models have been tested as liquids, however, there is no clarity on whether performance scales up with column/grid size and whether network size can be expected to provide good performance for certain types of tasks.

Network connectivity is very important factor and good results are achieved with different lambda values and it is found that values above 1.0 leads to better performance. If connectivity is reduced then it will leads to lower network activity and reduces chaotic effects in the network dynamics. The experiments show that more local connectivity provides more direct mapping from the input for the readout neurons. Different sizes of the networks were used on distributed grids and it is observed that small networks can also perform better.

The framework provides a promising approach for temporal pattern recognition on continuous input streams. The LSM theory says that larger and more dynamic networks should lead to better performance. However with the experiments carried out so far didn’t give us a remarkable difference.

3 Future Direction

For the simulations of LSM, especially during training and evaluation high simulation speeds are highly preferred and this leaves a big question how fast such large, dynamic, and parallel networks can be simulated on standard sequential machines. Spiking neurons can relatively easily be implemented in hardware therefore in the next phase a hardware/software partitioning strategy would be realized for high speed simulations. In this design, part of the network would be implemented on reconfigurable hardware (FPGA) and rest (readouts and front-end processing) would be performed in software. In this regard area efficient hardware architectures of abstract and more detailed spiking neuron models have been implemented and reported in [5].

References

[1] Maass, W., Natschläger, T. & Markram, H. Real-time computing without stable states: *A new framework for neural computation based on perturbations*, Neural Computation, vol. 15(11), pp.2531-2560 (2002).

[2] Maass, W., Natschläger, T. & Markram, H. *A Model for Real-Time Computation in Generic Neural Microcircuits*, in: Becker, S. et al (eds.), Proc. of NIPS 2002, Advances in Neural Information Processing Systems, vol. 15, pp.229-236. MIT Press, (2003).

[3] Eugene M. Izhikevich, “*Simple model of spiking neurons*”, IEEE transactions on neural networks, VOL. 14, NO. 6, November 2003.

[4] Kandel, E.R., Schwartz, J.H. & Jessell, T.M. (Eds.) *Principles of Neural Science* 3rd edition, Appleton & Lange, Connecticut (1991).

[5] Arfan Ghani, T. Martin McGinnity, Liam P. Maguire, JG Harkin, “*Area efficient Architecture for Large Scale Implementation of Biologically Plausible Spiking Neural Networks on Reconfigurable Hardware*” IEEE International Conference on Field Programmable Logic and applications (FPL), 2006.