An Agent Model Using Polychronous Networks*

Júlio L. R. Monteiro[1], Philippe Caillou[2], Marcio Lobo Netto[1]

[1]University of São Paulo – EPUSP
Av. Prof. Luciano Gualberto, t3, 380
05508-970 - São Paulo – SP - Brazil

[2]INRIA Saclay- Île-de-France
Laboratoire de Recherche en Informatique (LRI)

Université Paris Sud XI - IUT de Sceaux

julio.monteiro@poli.usp.br, caillou@lri.fr, lobonett@lsi.usp.br

Abstract. *In this paper, we present an agent model based on computation with polychronous groups on spiked neural networks, that is able to learn to return to known initial situations, without any guidance.*

1. Introduction

The recent developments on biologically inspired spiked neural networks [Gerstner and Kistler 2002] along with plasticity models such as STDP [Froemke and Yang 2002] allows the exploration of a new way to store and manipulate information spread over networks, exploring a feature known as polychronous groups, suggested by [Izhikevich 2006].

In this kind of neural network, neurons fire (or spike) when their activation reaches a defined threshold. Incoming synapses carry potentials from neurons that just fired, or spiked, according to a weight, that determines how much of the firing potential will be transported to the connected neurons.

The synaptic weights vary in accordance to some plasticity model, such as STDP (spike timing dependent plasticity), where the weights increase if the pre-synaptic neuron fires just before the post-synaptic one, and decrease if it fires afterward.

A key feature that allows the emergence of neuronal groups is the existence of nonzero propagation times in the synapses, meaning that once a neuron fire, a post-synaptic neuron will only receive the potential change after some specified propagation time. Another important requisite for having polychronous groups is the existence of excitatory and inhibitory neurons, with different firing equations, in a way that once the global network activation rate is over a threshold, and a certain amount of excitatory neurons are activated, they reach most of the inhibitory neurons, causing the network to reset. This process iterated over time causes the network to have a rhythmic behavior.

The polychronous groups can be understood as assemblies of neurons, that fire in the same temporal pattern, repeatedly in accordance to network inputs. Groups are dynamic entities due to the plasticity model. As soon as the synaptic weights begin to

* This research was made possible by a grant from the Internship program at INRIA Saclay Île-de-France, from June to October in 2008.

change, some groups disappear and new ones emerge. If the same inputs are presented over and over, it is observed that the number of groups attain an equilibrium [Izhikevich 2006], as the network "aligns" to the given inputs. In this sense, then, groups can be understood as "memories" in the network, as the network learns to associate this groups to sets of inputs and coordinate them with related outputs.

So far, polychronous network has been studied for theoretical purposes [Izhikevich 2006] [Maier and Miller 2008] and reservoir computing [Paugam-Moisy et al. 2008] implementations. However, embedding this kind of network in an agent in a simulated virtual world had not yet been attempted.

Our contribution in this work is to simulate one such network embedded into an agent that roams a virtual environment in search of food, receiving stimuli about its own energy level and from the environment (the nearest food direction), and being able to choose the direction to turn. Without any kind of rule to indicate that the agent should feed, we study if the use of Izikevitch network, STDP plasticity and the initial conditions will influence the behavior of the agent and if some "food searching" strategy will emerge.

2. Agent Model

The simulated agent is a blue tetrahedron that starts in the center and moves about with constant speed in a flat 20 x 20 field containing 50 randomly placed small green spheres that represent food. The agent also has an energy level, that starts at 1.0 and gradually decreases towards 0.0. Nothing special happens if the energy reaches zero.

As soon as the agent touches some food, its energy level is increased and the food is replaced with another one placed at random on the field. The agent is blocked from leaving the field by invisible walls.

The agent has two sensors, the first, called **directional sensor**, indicates the angle towards the nearest food object, with a 30 degrees arc and a radius of 10 units of distance. The second sensor is called **energy sensor** and measures the current energy level of the agent. A challenging aspect of this kind of simulation is converting the inputs from the environment to neural activation patterns, and convert neural patterns into output values to command the effectors. Our approach, not detailed here, was based on the way the desert scorpion finds its prey, using 8 neurons located in the tips of its legs, that measures oscillations from waves in the sand [Stürzl 2000]. Using only 8 excitatory and 8 inhibitory neurons, in a way that each 3 inhibit the opposite one, it is possible to convert an incoming wave into a firing pattern. In our case, with a simple adaptation of the same algorithm, we convert numbers representing angle or energy levels to a linear pattern of activation of neurons representing inputs and convert timed activations into numbers in the reverse way.

To act in the world, the agent may choose the angle to turn using its **directional actuator**, that received a value from 0.0 to 1.0 indicating the turning direction, in a 30 degrees arc.

Each of the sensors and actuators comprise a defined set of neurons, usually 5 for each that can be activated in patterns depending on the inputs or the network activity.

3. Experimental Settings

To study the effect of Izhikevich network and STDP plasticity on the behavior of the agent, we define several scenarios. We want especially to attest if by setting the energy initially to its maximum value, it will induce the behavior of "feeding" in the agent, or moving toward the food. On the other hand, if by setting the agent's initial energy to the minimum value would induce the behavior of avoiding food.

Below is a list of the used simulation scenarios:

- **Rand**: as a benchmark, we simulate an agent which moves entirely at random (*Random scenario*).
- **IZN**: a random 180 neurons network is generated using Izhikevich's neurons equations, with 80% of excitatory neurons and 20% of inhibitory ones. Each neuron has 30% of chance of being connected to each one of the others. This network is simulated for 10 min simulated time. This scenario has STDP *disabled*.
- **IZS**: This scenario uses IZN as a base, however having the STDP feature *enabled*.
- **ENEMAX**: This is similar to IZS, but starts with maximum energy for 10 min, after that the energy is allowed to change normally for another 10 min.
- **ENEMIN**: To compare with ENEMAX, this scenario starts with energy fixed at zero for 10 min, then enable variation for another 10 min.

4. Results and Conclusions

Each of the experiments was run 20 times using 20 different random seeds. The same 20 seeds were used for all the scenarios, allowing similar initial food setup and energy progression. All comparisons below were done on the averages of the 20 simulations.

We observe that by enforcing initial values for a long period on the network we can direct the polychronous group selection, thus allowing some form of training for the agent.

Simply by having STDP active, the groups start to change and take shape in accordance to the inputs. For that reason it is not enough to have the energy start at maximum and then immediately begin to decrease, because as we can observe in the Rand scenario, the energy quickly tends back to low values, and the agent will end up in fact learning to remain in that low energy values.

This is why we developed the scenarios ENEMAX and ENEMIN, that enforce the "desired" energy input for an extended period, allowing the network to stabilize and select the best groups that are compatible with that energy input.

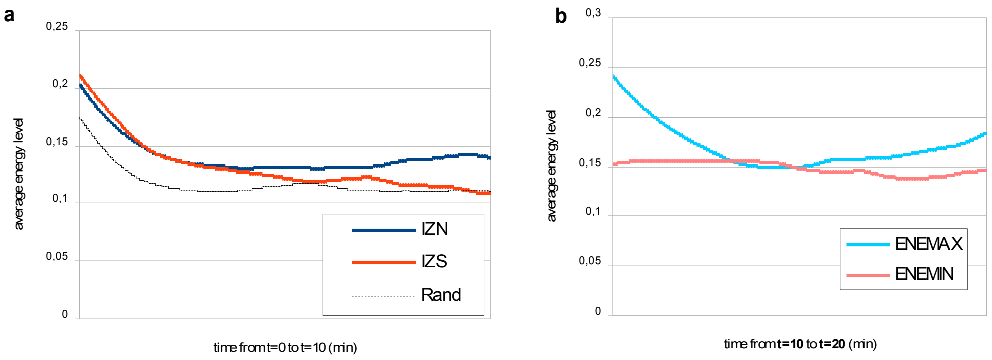

Figure 1. For each time step t, average energy level of the agent between t and the end of the simulation (end=10mn). In (a) we compare results for Rand(black, thin), IZN(blue) and with IZS(red). In (b) we compare ENEMAX (light blue) and ENEMIN(light red) scenarios, only showing the evolution of energy after it becomes variable.

With the simulation scenarios used, we can observe in Figure 1a that has STDP enabled right from the start of the simulation(IZS) will actually behave less efficiently than without STDP at all (IZN), because since the energy level will drop significantly right at start, the STDP will strengthen the polychronous groups associated to lower energy input patterns, inducing the agent to select output angles to avoid food.

On the other hand, on Figure 1b, when keeping the energy level at maximum for 10 minutes (ENEMAX), after a brief adaptation period, up to the inflection point in ENEMAX(around the middle part of the graph), the STDP reenforces groups related to full energy input patterns, in this way, most of the agent's output actions will tent to direct him towards food, so that the high energy levels can be maintained.

When the energy is kept at minimum for 10 min (ENEMIN), however, the behavior is even worse than IZS, because the network was taught to remain in lower energy states (bu reenforcing the lower energy polychronous groups), then the agent will learn to move away from food to remain in these states.

This shows that Izhikevich networks with STDP can be used as an alternative agent model, with abstract trainable inputs patterns that can direct the agent behavior.

5. Future Work

Presently, we are working on improving the performance of the simulator to be able to do even more tests and experiments to confirm our findings in larger scale networks and in other less random spatial configurations.

Another interesting possibility, aimed for future releases, is to include poisonous food (that reduces the energy level by an amount) to see how the agents react and what kind of learning behavior could be observed. If the agent learns to avoid poisonous food, we could be close to understanding how higher level concepts can be expressed within a spiked neural network.

This research continues as a collaboration between the Cognitio Research team at the Polytechnic School in University of São Paulo and researchers at LRI in INRIA Saclay Île-de-France, enabling interchange and joint work, considered relevant to both sides.

6. References

Izhikevich, E. M. 2006. Polychronization: computation with spikes. Neural Computation, 18:245-282.

Stürzl, W., Kempter, R. & van Hemmen, JL 2000. Theory of Arachnid Prey Localization. Physical Review Letters 84, pp.5668 – 5671.

Froemke, R. C. & Yang D. 2002. Spike-timing-dependent synaptic modification induced by natural spike trains. Nature Vol. 416, pp. 433-438, March 2002.

Gerstner, W. & Kistler, W. 2002. Spiking Neuron Models. Cambridge Univ. Press, Cambridge.

Paugam-Moisy, H., Martinez, R. & Bengio, S. 2008. Delay learning and polychronization for reservoir computing. Neurocomputing 71, 1143–1158

Maier, W. & Miller, B. 2008. A minimal model for the study of polychronous groups. arXiv:0806.1070v1 [cond-mat.dis-nn]