# A STDP RULE THAT FAVOURS CHAOTIC SPIKING OVER REGULAR SPIKING OF NEURONS

Mario Antoine Aoun

Montréal, Quebec, Canada

#### ABSTRACT

We compare the number of states of a Spiking Neural Network (SNN) composed from chaotic spiking neurons versus the number of states of a SNN composed from regular spiking neurons while both SNNs implementing a Spike Timing Dependent Plasticity (STDP) rule that we created. We find out that this STDP rule favors chaotic spiking since the number of states is larger in the chaotic SNN than the regular SNN. This chaotic favorability is not general; it is exclusive to this STDP rule only. This research falls under our long-term investigation of STDP and chaos theory.

#### Keywords

Competitive Hebbian Learning, Chaotic Spiking Neural Network, Adaptive Exponential Integrate and Fire (AdEx) Neuron, Spike Timing Dependent Plasticity, STDP, Chaos Theory, Synchronization, Coupling, Chaos Control, Time Delay, Regular Spiking Neuron, Nearest neighbour, Recurrent Neural Network

## **1. INTRODUCTION**

Spiking Neural Networks (SNNs) [1] are gaining much attention in the scientific literature, specifically neuromorphic engineering [2], due to their low energy consumption when implemented on electronic circuits; compared to artificial neural networks using conventional neuron models [2]. For instance, deep classical artificial neural networks, using gate activation functions as neurons, consume great amount of power in their training when implemented on Field Programmable Gate Arrays (FPGAs) [2]. On the contrary, spiking neurons implemented on Very Large Scale Integration (VLSI) chips consume way less energy [2]. The reason behind this difference is that spiking neurons are event driven systems [2]. Furthermore, SNNs are considered the third generation of neural networks because, as their name imply, they use spiking neuron models, where the first generation is based on perceptrons using threshold gates. and the second started with Hopfield nets using activation functions [3]. On the other side, Hebbian learning is the binding phenomenon that takes place between neurons after continuous stimulation they make upon each other [4]. Spike Timing Dependent Plasticity (STDP) incorporates exact timing of stimulations as a condition for the binding to occur [5]. In this work we will study STDP in the context of SNNs where STDP governs the connections weights inside the SNN. The SNN considered here (Fig. 1.a) is composed from spiking neurons that are recurrently connected to each other via time-delayed connections. Neurons have no selffeedback connections. The time delay is fixed and same for all the connections in the network. Connections have weights, too. When neurons connect and update their connections' weights via STDP, then they synchronize their activity to a same repetitive spiking pattern (Fig. 1.b). The latter is considered the network's state. In simple terms, this state is a synchronized spiking output pattern of all the neurons composing the network. In this paper, we are going to study the number of these states for a recurrent SNN composed from chaotic spiking neurons and a recurrent SNN composed from regular spiking neurons while both networks are implementing

same STDP rule. We will give experimental evidence that a *chaotic recurrent SNN* has indeed more states than a *regular recurrent SNN* while both are incorporating same STDP rule. But, we should emphasize that such claim and the results are exclusive to this specific STDP rule only. This research is part from our investigation of STDP and chaos theory, especially chaotic spiking neurons, which started in early 2006-2007 [6], spanned many years [7, 8, 9, 10] and continues to date.

The paper is divided as follows: In section 2 (Methods), we sketch the neural network architecture and explain the neuron model used. Also, we introduce our synaptic plasticity model based on STDP. In section 3 (Results), we provide a comparable study of the number of states that can be stabilized in network of chaotic spiking neurons implementing STDP versus the number of states that can be stabilized in a network of regular spiking neurons implementing the same STDP. Section 4 is a discussion of the results and section 5 concludes the paper.

## 2. METHODS

The neural network architecture that will be used in this paper is composed of n spiking neurons that are recurrently connected to one another, hence a recurrent spiking neural network. We use the Adaptive Exponential (AdEx) integrate and fire neuron model [11], as the elementary component of the network, because it is a neuron model that can simulate regular spiking or chaotic spiking by changing its parameters [12]. Each connection, between any two neurons in the network, has a fixed time delay and an adjustable weight. The time delays are kept fixed because the network will synchronize its spiking activity in a window period equal to the time delay. The weights are adjusted using synaptic plasticity, as we will see next. The network architecture is illustrated in Figure 1.a.



Figure 1. a. Neural Network Architecture: *n* AdEx Neurons  $(A_1, ..., A_n)$  are recurrently connected. Every connection has a varying weight " $\omega$ " and a fixed time delay " $\tau$ ". Note that " $\tau$ " is identical for all connections. b. Synchronized spikes Output: When the network stabilizes, then all the neurons synchronize their output to a same repetitive spiking pattern of length " $\tau$ ". In this example, " $\tau$ " is equal to 300 and the repetitive pattern is composed from three spikes. Time resolution is in ms. Time steps from 10000 to 14500. Pattern Blocks: (10000-10299, 10300-10599, 10600-10899...). Spikes: (10002, 10074, 10298, 10302, 10374, 10598, 10602, 10674, 10898...). This repetitive pattern is considered as the *network state*.

As we can see in Figure 1.a, we have a neural network of *n* AdEx Neurons (e.g.,  $A_1$  to  $A_n$ ) where the subscript "*i*" is the index of an AdEx Neuron "*A*". Each neuron has *n*-1 connections because there are no self-feedback connections. A connection from neuron  $A_j$  to neuron  $A_i$  has a time delay " $\tau_{i,j}$ " and a weight " $\omega_{i,j}$ ".

We rewrite the AdEx Neuron equations [12] in their Euler form:

$$V_{i}(t) = V_{i}(t-1) + \frac{dt \left[ -g_{L}(V_{i}(t-1) - E_{L}) + g_{L} \Delta_{T} \exp\left(\frac{V_{i}(t-1) - V_{T}}{\Delta_{T}}\right) + Ic - \psi_{i}(t-1) \right]}{C}$$
(1)  
$$\psi_{i}(t) = \psi_{i}(t-1) + \frac{dt [a(V_{i}(t-1) - E_{L}) - \psi_{i}(t-1)]}{\tau_{w}}$$

Where  $V_i(t)$  is the neuron's voltage at time t and  $\Psi_i(t)$  is its adaptation variable. The subscript "i" indicates the neuron's index inside the network as we mentioned earlier. dt is the Euler time step which is set to 0.1 ms for good precision. The parameters of the AdEx Neuron for both regular and chaotic firing are retrieved from [12] and are shown in Table 1, next.

 Table 1. Configuration of the AdEx Neuron parameters for regular and chaotic firing modes

 [12].

Mode \ Parameters	С	$g_L$	$E_L$	$V_T$	$\Delta_T$	а	$ au_w$	b	$V_r$	Ic	θ
Regular	200	10	-70	-50	2	2	30	0	-58	500	0
Chaotic	100	12	-60	-50	2	-11	130	30	-48	160	0

*C* is membrane capacitance in *pF*,  $g_L$  is leak conductance in *nS*,  $E_L$  is leak reversal in *mV*,  $V_T$  is threshold potential in *mV*,  $\Delta_T$  is rise slope factor in *mV*, *a* is 'sub-threshold' adaptation conductance in *nS*,  $\tau_w$  is adaptation time constant in *ms*, *b* is spike trigger adaptation current increment in *pA*,  $V_r$  is reset potential in *mV*, *Ic* is input current in *pA* and  $\theta$  is reset threshold in *mV* [12].

The neuron initial conditions are:

$$V_{i}(0) = V_{r}$$
  

$$\psi_{i}(0) = 0$$

$$Ic_{i} = LIc + (UIc - LIc) * R_{i}$$
(2)

Where,

 $R_i$  is a random decimal between 0 and 1.

*LIc* is the lower bound of the default input current (*Ic*).

*UIc* is the upper bound of the default input current (*Ic*).

We note that *LIc* and *UIc* are set according to the firing mode (i.e. regular or chaotic) requested. For instance, if we want regular firing, we can set *LIc* to 400 and *UIc* to 600, which are in fact 100 units away from their mean *Ic* which is 500. As for chaotic firing, we can set *LIc* to 150 and *UIc* to 170, which are 10 units away from their mean *Ic* that is 160. Note that in the case of chaotic firing, it was observed in [12] that the neuron's chaotic firing is sensitive to the input current 160 (e.g. Ic = 160 pA) and values far away from 160 would ruin the chaotic behaviour of the neuron. In fact, we need the neurons of the network to have slight variations in their initial conditions (i.e. initial value of the Input current *Ic*) so their output will be different from one another; which will offer great opportunity to the synaptic plasticity model that depends on the firing time of the neurons, to operate properly as we will see next.

When the neuron's voltage passes its threshold (i.e.  $V_i > \theta$ ), then the neuron fires a spike that is represented by  $\gamma_i$ :

$$\gamma_i(t) = \begin{cases} 1, V_i(t) > \theta \\ 0, V_i(t) \le \theta \end{cases}$$
(3)

The neuron receives time-delayed spikes from other neurons through its incoming connections, which are scaled by each connection's weight  $\omega_{i,j}$  and summed up as a total input  $S_i(t)$  which would be added to the neuron's voltage  $V_i(t)$ .

But, before evaluating  $S_i(t)$  and adding it to  $V_i(t)$ , the weights of the connections should be updated where the weight change  $\Delta \omega_{i,j}$  is calculated and added to  $\omega_{i,j}$ .

So, in order to update the weights of the connections of the network then we implement a competitive scenario of STDP that is similar to the approach made by Song et al, in 2000 [13], in their work on Competitive Hebbian learning based on STDP [13]. Our implementation of a competitive STDP protocol uses nearest neighbour spikes and is described as:

$$If \quad (t - \tau_{i,j}) - t_{pre(j)} \ge t_{post(j)} - (t - \tau_{i,j}) \& V_i(t) > \theta \quad then \quad \Delta w_{i,j} = \Delta w^-$$

$$If \quad (t - \tau_{i,j}) - t_{pre(j)} < t_{post(j)} - (t - \tau_{i,j}) \qquad then \quad \Delta w_{i,j} = \Delta w^+$$

$$(4)$$

Where,

 $t_{pre(j)}$  is the time of the spike, of input neuron *j*, that occurred before  $t - \tau_{i,j}$ .  $t_{post(j)}$  is the time of the spike, of input neuron *j*, that occurred after  $t - \tau_{i,j}$ .  $\Delta \omega^{-}$  is the negative decrease of the weight change and equal to:

$$\Delta w^{-} = A^{-} * e^{\left(\frac{t - \tau_{i,j} - t_{post(j)}}{\tau_{i,j}}\right)}$$
(5)

 $\Delta \omega^+$  is the positive increase of the weight change, equal to:

$$\Delta w^{+} = A^{+} * e^{\left(-\frac{t-\tau_{i,j}-t_{pre(j)}}{\tau_{i,j}}\right)}$$
(6)

 $A^-$ ,  $A^+$  are the lower and upper boundaries, of the negative and positive weight change, respectively, which are defined as the following:

$$A^- = V_r * \mu^- \tag{7}$$

And,

$$A^{+} = \left( \left( \theta - V_{r} \right) - w_{i,j} \right) * \mu^{+}$$

$$\tag{8}$$

Where,

 $\mu^-$ ,  $\mu^+$  constants that are experimentally set to 0.01 and 0.1, respectively. We noticed that setting  $\mu^-$  to 0.01 then the neuron's voltage wouldn't decrease to very low values during start of STDP phase.

The reason behind this choice in defining  $A^-$  and  $A^+$  is supported by the fact that these parameters should be voltage dependent as suggested in [14]. Furthermore, by setting  $A^- = V_r *$ 

 $\mu^-$  then we are sure that this parameter is always negative, which is a must for Long Term Depression (LTD) to take place in the STDP protocol. Second, by setting  $A^+ = [(\theta - V_r) - \omega_{i,j}] * \mu^+$ , then we are sure that Long Term Potentiation (LTP) will settle down once the weight of a connection has reached a maximal value equal to  $\theta - V_r$ .

Last but not least, the weight of every connection is updated according to the following:

$$w_{i,j}(t) = w_{i,j}(t-1) + \Delta w_{i,j}$$
(9)

Afterwards, the Input to neuron *i* is calculated according to the following:

$$S_{i}(t) = \sum_{j=1, i \neq j}^{n} w_{i,j}(t) * \gamma_{j}(t_{(pre, post)(j)})$$
(10)

Where,

$$t_{(pre,post)(j)} = \begin{cases} t_{pre(j)} \text{ in case of } \Delta w^+ \\ t_{post(j)} \text{ in case of } \Delta w^- \\ t - \tau_{i,j} \text{ in case of } \Delta w^0 \end{cases}$$
(11)

 $\Delta \omega^0$  is the case of  $\Delta \omega = 0$ , which occurs when the neurons had synchronized their neural states over a period  $\tau$ . This means there is no more change in the weights that is necessary.

After calculating the input  $S_i(t)$  to the neuron, then it is added to the neuron's voltage  $V_i(t)$ :

$$V_i(t) = V_i(t) + S_i(t)$$
 (12)

When the neuron's voltage crosses its threshold  $\theta$ , it is reset to its reset value  $V_r$  and its adaptation variable is set to the current value of the latter plus the adaptation reset parameter b (spike trigger adaptation current increment):

If 
$$V_i(t) > \theta$$
 then  $V_i(t) = V_r$  and  $\psi_i(t) = \psi_i(t) + b$  (13)

The weights of all the connections are initially set to  $\theta$ , so the neurons inside the network run in isolation and evolve their dynamics. Then, synaptic plasticity starts to operate by updating the connection weights of every neuron. For instance, in the following experiments, the neurons run in isolation for 5000 time steps before their connections weights start to get updated.

### **3. RESULTS**

We studied the number of different states, depicted by synchronous repetitive firing patterns (e.g., Figure 1.b), that can be stabilized in a recurrent SNN (as in Figure 1.a) composed from P spiking neurons, such that P is an arbitrary positive integer that dictates the network's cardinality (i.e., the size of the network), where STDP is managing the connections weights between the neurons, as we explained in the previous section. We compared the number of stabilized states of a P size recurrent SNN composed from P chaotic spiking neurons and another P size recurrent SNN composed from P regular spiking neurons according to the settings of Table 1. For simplicity, the recurrent SNN composed from regular spiking neurons is referred as chaotic SNN and the recurrent SNN composed from regular spiking neurons is referred as regular SNN. The number of neurons that constitute both the network of chaotic spiking neurons and the network of regular spiking neurons will be between 10 and 50:

$$P \in \{10, 20, 30, 40, 50\}$$
(14)

We aim to find out that the capacity of the chaotic SNN is larger than the capacity of a regular SNN (i.e., The number of stabilized states inside recurrent SNN of chaotic spiking neurons is superior than the number of stabilized states inside recurrent SNN of regular spiking neurons). To do this, we run twice, five sets of one hundred random experiments each (Random in the sense that an experiment starts with random initial conditions as mentioned in the previous section – Equation 2). The total will be 2 \* 5 \* 100 = 1000 experiments. Each experiment is composed of a recurrent SNN of *P* Neurons while increasing *P* by 10 for each set, starting from P = 10 to P = 50. The first five sets of experiments will be executed on recurrent SNN composed of regular spiking neurons, while the second five sets of experiments will be executed on recurrent SNN composed of regular spiking neurons. The results of the two 'five sets' of experiments are illustrated in a bar graph in Figure 4.c. Also, we show neurons activity (e.g., voltage outputs), evolution of connections weights and STDP windows for two runs of both networks (Figures 2.a, 2.b, 3.a, 3.b, 4.a and 4.b, respectively).



Figure 2. (a) and (b) show Voltage activity for a chaotic and regular Spiking Neural Network (SNN), respectively, composed of 10 neurons each. Neurons start synchronizing after time step 5000. Time resolution is 1ms.



Figure 3. (a) and (b) show average of all connections weights inside a chaotic and regular Spiking Neural Network (SNN), respectively. Each network is composed from 10 neurons. Neurons start synchronizing after time step 5000. Weights stabilization is fast and reaches a steady value. Time resolution is 1ms.



Figure 4. (a) and (b) show STDP windows for a chaotic and regular Spiking Neural Network (SNN), respectively. (c) is a comparison of the number of different stabilized neural states for both types of networks through 1000 total experiments (each network runs a set of 100 experiments with increasing number of neurons for each set).

By analysing the graph of Figure 4.c, we observe that the number of different stabilized states of recurrent SNN composed from chaotic spiking neurons is always superior to the number of different stabilized states of recurrent SNN composed from regular spiking neurons for any number of neurons composing the network.

#### 4. DISCUSSION

We should note that the number of states that can be reached within this type of network (Figure 1.a) is theoretically infinite because it increases by increasing the time delay of the neural network connections. In our experiments, the time delay was fixed ( $\tau_{i,j}$  is equal to 300). We notice (Figure 4.c) that by increasing the number of neurons then the number of states, in both types (e.g., chaotic and regular) of networks, decreases. This is true, because for a fixed time delay, increasing the number of neurons, that compose the network, causes more saturation of spikes that eventually reduces the number of unique patterns. Also, we noticed very high

excitation in both networks; this is because  $\Delta \omega^+$  (Equation 4) is always active. Thus, the synchronized output of the network can be assigned to a "Not" logic gate, this way the spikes pattern generated by the network is reversed, as shown in Figure 1.b.

#### **5.** CONCLUSIONS

Experimental simulations were conducted in the aim of comparing the number of different stabilized states (i.e., synchronized firing activity of repetitive spiking patterns) through a network of chaotic spiking neurons vs. a network of regular spiking neurons whilst the connections between the neurons inside both networks followed same competitive nearest neighbour STDP rule that we created. We tested networks composed of 10, 20, 30, 40 and 50 neurons. The results of the experiments confirmed that this STDP rule favours chaotic spiking over regular spiking of neurons because the numbers of different stabilized states within networks of chaotic spiking neurons were shown to be way larger than the numbers of different stabilized states within the networks of regular spiking neurons. We have to note that this behaviour is not general but restricted to this STDP rule only.

#### REFERENCES

- [1] Maass, Wolfgang, and Christopher M. Bishop, eds. *Pulsed neural networks*. MIT press, 2001.
- [2] Greengard, Samuel. "Neuromorphic chips take shape." *Communications of the ACM* 63, no. 8 (2020): 9-11.
- [3] Maass, Wolfgang. "Networks of spiking neurons: the third generation of neural network models." *Neural networks* 10, no. 9 (1997): 1659-1671.
- [4] Hebb, Donald Olding. *The organization of behavior: A neuropsychological theory*. Psychology Press, 2005.
- [5] Markram, Henry, Wulfram Gerstner, and Per Jesper Sjöström. "Spike-timing-dependent plasticity: a comprehensive overview." *Frontiers in synaptic neuroscience* 4 (2012): 2.
- [6] Aoun, Mario Antoine. *Temporal difference method with hebbian plasticity rule as a learning algorithm in networks of chaotic spiking neurons*. [Master's Thesis], Notre Dame University, Louaize, Lebanon, 2007.
- [7] Aoun, Mario Antoine. "Stdp within nds neurons." In *International Symposium on Neural Networks*, pp. 33-43. Springer, Berlin, Heidelberg, 2010.
- [8] Aoun, Mario Antoine, and Mounir Boukadoum. "Learning algorithm and neurocomputing architecture for NDS Neurons." In 2014 IEEE 13th International Conference on Cognitive Informatics and Cognitive Computing, pp. 126-132. IEEE, 2014.
- [9] Aoun, Mario Antoine, and Mounir Boukadoum. "Chaotic liquid state machine." *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)* 9, no. 4 (2015): 1-20.
- [10] Aoun, Mario Antoine. *Learning and memory within chaotic neurons*. [Phd Thesis], University of Quebec in Montreal, Canada, 2019.
- [11] Brette, Romain, and Wulfram Gerstner. "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity." *Journal of neurophysiology* 94, no. 5 (2005): 3637-3642.
- [12] Naud, Richard, Nicolas Marcille, Claudia Clopath, and Wulfram Gerstner. "Firing patterns in the adaptive exponential integrate-and-fire model." *Biological cybernetics* 99, no. 4 (2008): 335-347.
- [13] Song, Sen, Kenneth D. Miller, and Larry F. Abbott. "Competitive Hebbian learning through spike-timing-dependent synaptic plasticity." *Nature neuroscience* 3, no. 9 (2000): 919-926.
- [14] Clopath, Claudia, and Wulfram Gerstner. "Voltage and spike timing interact in STDP-a unified model." *Frontiers in synaptic neuroscience* 2 (2010): 25.

## AUTHOR

Mario Antoine Aoun has Phd. in Cognitive Informatics from UQAM. His research interests are: Theoretical Computer Science, Cognitive Computing, Cognitive Informatics, Machine Learning, Mathematical Modelling, Evolutionary Algorithms, Genetic Algorithms, Spiking Neurons, Chaotic Spiking Neural Networks, Deep Learning, Deep Neural Network, Recurrent Neural Network, Cognitive Neurodynamics, Neuro-economics, Synaptic Plasticity, Spike Timing Dependent Plasticity, Nonlinear Dynamics, Chaotic Neurodynamics, Computational Neuroscience, Neural Computing Architectures, Synchronization, Quantum Computing, Relativistic Computing, Reservoir Computing, Chaos Theory, Chaos Control and Hypercomputation. His email address is: <u>mario@live.ca</u>

