[Re] Connectivity reflects coding: a model of voltage-based STDP with homeostasis

René Larisch[1]

1 Professorship for Artificial Intelligence, Department of Computer Science, Chemnitz University of Technology, D-09107 Chemnitz, Germany

rene.larisch@informatik.tu-chemnitz.de

🖝 **Article repository**

🖝 **Code repository**

A reference implementation of

→ Connectivity reflects coding: a model of voltage-based STDP with homeostasis, C. Clopath, L. Büsing, E. Vasilaki and W. Gerstner, In: Nature Neuroscience 13.3 (2010), pp. 344–352, doi= 10.1038/nn.2479

Introduction

Since the first description of spike timing-dependent plasticity (STDP) [2], different models of STDP have been published to reproduce different experimental findings. Early implementations such as pair-based STDP learning rules failed to reproduce some experimental observations, for example triplet or quadruplets experiments [5].

Clopath et al. [3] introduced a STDP model able to reproduce the experimental findings of triplet studies. They propose a biologically-motivated model with a voltage-based learning rule, where the occurrence of long term depression (LTD) or long term potentiation (LTP) depends on the postsynaptic membrane voltage. Clopath and colleagues could reproduce how the occurrence of LTD and LTP depends on the depolarizing of the postsynaptic membrane potential, as observed in voltage-clamp [4] and stationary-depolarization experiments [1].

Further, they could reproduce experimental finding from spike pair repetition and triplet experiments [6]. They were able to show that their learning rule can develop stable weights, as needed for learning the receptive fields of V1 simple cells. Therefore, they implemented a homeostatic mechanism to control the amount of generated LTD, based on the relationship between the average postsynaptic membrane potential and a reference value. Their model led to two different connectivity structures, depending on the spiking behavior of the neurons: if the neurons fire strongly at the same time, they build strong bidirectional connections (as in rate-coded Hebbian learning). If they fire in a specific temporal order, their connectivity structure follows that order (temporal coding).

Methods

Overview

The original model was implemented in Matlab (http://modeldb.yale.edu/144566) to demonstrate the stable learning of weights. This model reimplementation is written in Python (v2.7) with the help of the neuro-simulator ANNarchy [7], numpy (v1.11.0)

and matplotlib (v1.5.1). The reimplementation is mainly based on the description of neuron model and learning rule in the original publication [3]. Because of the lack of further description of the homeostatic mechanism and the neural behavior after a emitted spike, the Matlab code is used as the second reference for this reimplementation.

Model description

Neural model

Clopath et al. [3] used an adaptive exponential integrate-and-fire (AdEx) neuron in their model. The exact neural model is mainly derived from the description in the Matlab source code (see **Eq.** 1):

$$C\frac{du}{dt} = -g_L\left(u - E_L\right) + g_L\,\Delta_T\,e^{\frac{u-V_T}{\Delta_T}} - w_{ad} + z + I \tag{1}$$

where u is the membrane potential, C the membrane capacitance, the leak conductance is g_L and E_L is the resting potential. The slope factor (Δ_T) and the spiking threshold (V_T) are describing the behavior of the exponential term.

If the membrane potential (u) is above V_T, the neuron spikes and the membrane potential increases exponentially. To simulate the spike upswing for $2ms$ after a spike is emitted, they used the so called 'resolution trick'. For this, they simulated once the complete change in the membrane potential through a spike with high precision and integrated. For simulations, they used the integrated number and clamped the membrane potential for $2ms$.

This means, the membrane potential is set to $29.4mV$ after a spike, one millisecond later to $29.4mV + 3.462mV$ and another millisecond later to $E_L + 15mV + 6.0984mV$.

The depolarizing spike afterpotential is z and it decays over time to zero (see **Eq.** 2).

$$\tau_z\frac{dz}{dt} = -z \tag{2}$$

The hyperpolarization current described by w_{ad} (see **Eq.** 3). After a spike, w_{ad} is increased by the amount b and decays down to the resting potential E_L otherwise.

$$\tau_{w_{ad}}\frac{dw_{ad}}{dt} = a(u - E_L) - w_{ad} \tag{3}$$

The adaptive spiking threshold (V_T) is set to $V_{T_{max}}$ after the neuron spiked and will decay to $V_{T_{rest}}$ (see **Eq.** 4).

$$\tau_{V_T}\frac{dV_T}{dt} = -(V_T - V_{T_{rest}}) \tag{4}$$

The values of the parameters of the neural model are taken from the original paper.

Synaptic model

The proposed learning rule consists of two terms: a long term potentiation (LTP) term (**Eq.** 5) controls increases in synaptic efficiency:

$$LTP_{Term} = A_{LTP}\,\bar{x}_i\,(u - \theta_+)_+\,(\bar{u}_+ - \theta_-)_+ \tag{5}$$

Therefore, A_{LTP} is the learning rate for the LTP term. The parameters θ_+ and θ_- are plasticity thresholds for the membrane potential (u), respectively for the over time averaged version ($\bar{u}_+$ (see **Eq.** 6)). The original paper they did not mentioned the meaning of both thresholds. With $\theta_+ = -45.3mV$ it is above the spiking threshold. This suggest, that the threshold avoid the occurrence of LTP if the postsynaptic neuron

not spiked already. With $\theta_+ = E_L$, LTP only occurs if the membrane potential is above the resting potential.

$$\tau_+ \frac{d\bar{u}_+}{dt} = -\bar{u}_+ + u \tag{6}$$

On every spike of the presynaptic neuron, the spike train $\bar{x}_i$ is increased by one and will decay over time with a time constant τ_x (see **Eq.** 7). If the presynaptic neuron spikes to a time point t, the spike counter X_i is 1, otherwise 0.

$$\tau_x \frac{d\bar{x}_i}{dt} = -\bar{x}_i + X_i \tag{7}$$

With this term, LTP occurs when the presynaptic spike trace ($\bar{x}_i$) is above zero, the postsynaptic membrane potential u is over the threshold θ_+ and the membrane potential trace $\bar{u}_-$ is above θ_-. This happens when the postsynaptic neuron spikes shortly after the presynaptic neuron or if the membrane potential is high long enough, so that $\bar{u}_-$ exceeds θ_-.

The second term is the long term depression (LTD) term (**Eq.** 8) and governs decreases in synaptic efficiency:

$$LTD_{Term} = A_{LTD} \, (\frac{\bar{\bar{u}}}{u_{ref}^2}) \, X_i \, (\bar{u}_- - \theta_-)_+ \tag{8}$$

The presynaptic spike counter (X_i) is set to one after a spike and zero otherwise (as mentioned above). $\bar{u}_-$ is a second trace of the postsynaptic membrane potential similar to $\bar{u}_+$, but with $\tau_- > \tau_+$. If it exceeds the θ_- threshold, and a presynaptic spike is emitted, LTD occurs. This happens when the presynaptic neurons spikes after the postsynaptic one.

The amplitude of the LTD term, and with that the balance between LTP and LTD, is adjusted with respect to the ratio between $\bar{\bar{u}}$ and a reference value (u_{ref}^2), hence implementing a homeostatic mechanism (**Eq.** 9).

$$\tau_{\bar{\bar{u}}} \frac{d\bar{\bar{u}}}{dt} = [(u - E_L)^2] - \bar{\bar{u}} \tag{9}$$

This mechanism is computed over the quadratic distance of the membrane potential and the resting potential E_L. **Sentence not very clear** With a higher activity, $\bar{\bar{u}}$ increases, leading to a higher amount of LTD and a weight decrease. In contrast, a lower activity decreases the amount of LTD and the weights can increase. Through the ratio of $\bar{\bar{u}}$ with u_{ref}^2, this mechanism can enforce the connections to decrease down to the minimum weight bound or increase to the maximum weight bound. This requires a hard upper and lower bound for the weights and leads to a binomial distribution of the weights. The weight change over time depends on the positive LTP and the negative LTD terms (**Eq.** 10):

$$\frac{dw}{dt} = -LTD_{Term} + LTP_{Term} \tag{10}$$

All parameters of the neuron model and the basis set of parameters for the learning rule are taken from the original publication. Some parameters of the learning rule differ from experiment to experiment, in particular the value of the homeostatic mechanism (**be more precise**) and the maximum weight value. A table with the different parameters for each task is presented in **Tab.** 2 and **Tab.** 1.

Reproduction of experiments

In the original publication, the authors reproduce spike timing triplet experiments in the visual cortex of rats [6]. Furthermore, they investigate the emerged structure of the connectivity depending on the spiking behavior.

To validate the reimplementation, we reproduce the classical spike timing-dependent learning window (**Fig.** 2a in [3]), the frequency repetition task to reproduce a triplet experiment (**Fig.** 2b in [3]) and the influence of spiking order to connectivity (**Fig.** 4a, down and **Fig.** 4b, down in [3]). In the available matlab source code, they investigate the stable learning of weights using 500 input neurons and one postsynaptic neuron. The firing rate of these neurons follow a Gaussian distribution and the spike timing a Poisson process (similar to **Fig.** 5a in [3]). This task is reimplemented as well. With this analysis, the functionality of the reimplementation is shown on the main feature of this learning rule. We reproduced pair-based and triplet STDP data and analyzed the obtained connection patterns depending on the neuronal spiking behavior.

TODO: are there experiments you did not reproduce and why? It is important for the partial/full replication

The experiment protocols are based on the description on the publication of Clopath et al. [3]. The implementation of the learning rule was mainly made according to the available matlab source code. Despite the effort to be as close as possible to the original implementation and description, the internal processing of the equations by ANNarchy can lead to a different execution order. Therefore, differing results occurred.

Be more specific: what is different? why?

Further, the chosen integration time step can have an influence of the computation result. On all reimplemented analysis, a time step of $dt = 1ms$ is chosen. **Why? What is it in the original paper?** Because of that, some parameters had to be adapted to reproduce the original results. **Which ones?**

To reproduce the STDP learning window **Which Fig?**, we create a list of discrete time points where the pre- or postsynaptic neurons should emit spikes. The presynaptic neuron spikes every $50ms$. The postsynaptic neurons spikes in a range from $1ms$ to $15ms$ before or after the presynaptic neuron. For the repetition frequency experiment or the triplet experiment **Figs?**, the number of pre- and postsynaptic spike pairs increases from a pair frequency of $0.1Hz$ to $50Hz$. The time between a pre- and postsynaptic spike of a pair is $10ms$. To reproduce this experiments, it was necessary to set $\bar{\bar{u}}$ to a fixed value. **Is it different from the paper?** The parameter changes are shown in **Tab.** 2.

To analyze the connectivity depending the number of spikes, a small network with ten neurons connected with each other is build. Every neuron receives input from one additional neuron, with Poisson-distributed spike patterns. The firing rate of each Poisson neuron is increased from 2Hz to 20Hz, influencing the firing rate of the 10 corresponding neurons in the network.

Because the reimplementation of the model is mainly based on the Matlab source code from modelDB, the emergence of stable weights was achieved by presenting a Gaussian input over 500 presynaptic neurons and one postsynaptic neuron (**contrary to what in the original paper?**. For every trial (125ms) ten Gaussian patterns are created to determine the activity of the 500 input neurons.

As in the Matlab source code, the learning rates (A_{LTP} and A_{LTD}) are increased by a factor ten to speed up the learning.

The experiments for stable weight learning, to show a specific connectivity pattern, require the original homeostatic mechanism. Changes to the default parameters for this tasks are shown in **Tab.** 1.

Table 1: Changed parameters for connectivity experiments.

Task	Parameter	Value
Rate based connectivity	w_{max}	$0.25nA$
Temporal based connectivity	w_{max}	$0.30nA$
Stable weight by Gaussian input	w_{max}	$3.0nA$
Stable weight by Gaussian input	A_{LTD}	$1.4 * 10^{-3}$
Stable weight by Gaussian input	A_{LTP}	$0.8 * 10^{-3}$

Task	Parameter	Value

Table 2: Changed parameters for weight change experiments.

Task	Parameter	Value
STDP learning window	$\bar{\bar{u}}$	$80mV^2$
STDP learning window	θ_-	$-60.0mV$
triplet experiment	$\bar{\bar{u}}$	$120mV^2$

Reimplementation

What is missing is description of the reimplementation itself, i.e. present ANNarchy, which structures are used (I suppressed the references to PoissonPopulation and co in the previous part), what was hard, why ANNarchy and not pure Python?, etc

You can put code in the article:

```
params = """
EL = -70.4      :population
VTrest = -50.4  :population
taux = 15.0     :population  """
eqs = """
dg_vm/dt = EL/1000 : min = EL, init=-70.4
Spike = if state == 1: 1.0 else: 0.0
dReset/dt = if state == 1: +1 else: -Reset
dxtrace/dt = if state == 1: +1/taux else: -xtrace/taux : init = 0.0
state = if state >0: -1 else: 0"""
neuron = Neuron(parameters = params,
                equations = eqs,
                reset = """ g_vm = EL
                            state = 1""",
                spike = """g_vm > VTrest""")
```

Results

Perhaps structure more the results

Pair-based and triplet STDP experiments

The classic-pair based spike timing learning window is presented in **Fig. 1 left**. If the postsynaptic neuron spikes before the presynaptic one, LTD occurs (red line). If the postsynaptic neuron spikes after the presynaptic one, LTP occurs (blue line). The x-axis represents the time difference between pre- and post-synaptic spikes, relative to the postsynaptic spike. The resulting graph is similar to the presented one in the original publication. A small difference can be seen in the higher positive and negative change (**From how much?**). This could be caused by a different internal processing of ANNarchy. (**Be more precise...**)

The analysis of the pairing repetition frequency task is shown on **Fig. 1 right**. With lower repetition frequency, post-pre pairs (red line) lead to LTD. At a repetition
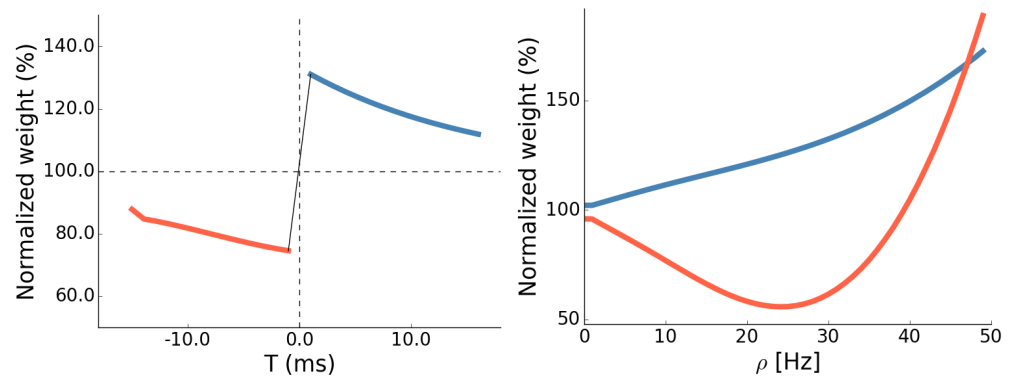
Figure 1: Replication of experimental findings. Left, the classic STDP learning window. On the x-axis is the time of a postsynaptic spike in relation to the presynaptic spike presented. **Right**, weight changes as a function of pair frequency repetition. Pre-post pairs are the blue line and post-pre pairs the red line.

frequency around $30Hz$, the post-pre pairs are under the influence of the next post-pre pair and the post-pre-post triplets lead to LTP. If the repetition frequency of post-pre pairs is around $50Hz$, the same amount of LTP emerges as in pre-post pairs. These results are similar to the original paper.

Connectivity analysis

In addition to the replication of experimental findings of pair-based and triplet STDP experiments, Clopath et al. [3] presented how the synaptic connectivity, emerging from the proposed learning rule, is influenced by the spiking behavior of the neurons. **Fig. 2 left** shows the obtained connection structure if neurons fire with different frequencies. Here, the color scheme is different from the original publication (**why?**). Weak connections (above $\frac{3}{4}$ of the maximum activity) are yellow. Light green represents strongly unidirectional connections while dark green represents strong bidirectional connections. Neurons with similarly high firing rates develop strong bidirectional connections, because they are often active at the same time. This suggests that learning is based on the correlation between the neuronal activities in a Hebbian manner. This is in line with the connection pattern presented in the original paper. (**be more quantitative there**)

If the neurons fire in a specific temporal order, this sequence is reflected in the connection pattern (see **Fig. 2 left**). As in the original paper, the connections from neurons which are firing a long time after or before the post-synaptic neuron are weak, while they are strong to neurons which fired a short time after the neurons.

The development of the weights over time, by presenting Gaussian shaped inputs, is presented in **Fig. 3**, After 500 epochs, stable weights begin to emerge, similar to the original matlab source code. **More details**

Conclusion

Our reimplementation of voltage based STDP learning rule from Clopath et al. [3] is able to reproduce the experimental data and the emergent connectivity structures proposed in the original paper.

The description of the learning rule in the original publication comprises enough details to understand the different components and their interaction. However, two main components have not been described adequately to allow a direct reimplementation: the 'resolution trick' of the membrane potential after spike emission and the
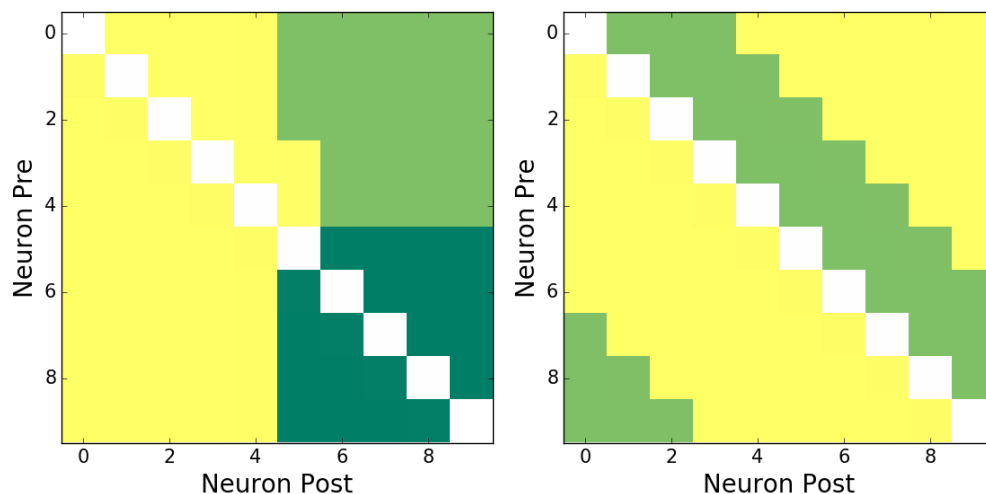
Figure 2: Different connectivity patterns. Depending on the spiking activity, different connectivity patterns emerge between the neurons. The color scheme is different from the original publication. Weak connections are yellow, strong unidirectional connections are light green and dark green are strong bidirectional connections. ****Why not use the same color code, then?**** **Left**, Neurons with similar high firing rates develop strong bidirectional connections. **Right**, connection pattern follows the temporal order of the occurred spikes.
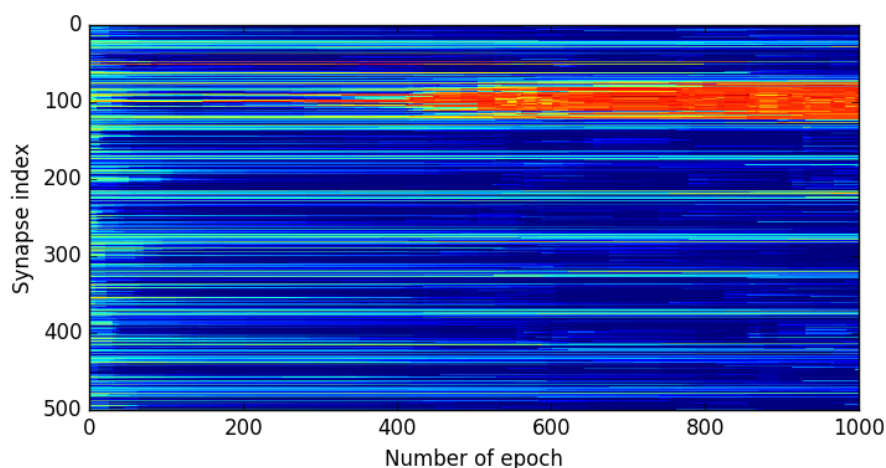
Figure 3: Stable weights on Poisson distributed Input. Colors show the weight value at the end of the current epoch from the presynaptic neuron to a single postsynaptic neuron. On the y-Axis is the presynaptic index indicated and the x-axis shows the number of epoch.

equation for the homeostatic mechanism ($\bar{\bar{u}}$).

Stable learning is only possible with a working homeostasis mechanism, that regulates the amount of LTD. **Was it described in the paper, or did you have to guess it from the Matlab code?** The dependency of $\bar{\bar{u}}$ on the homeostatic mechanism is necessary to implement the right behavior of the membrane potential. The reimplementation has greatly benefited from the release of the source code on modelDB.

Acknowledgment

References

[1] A. Artola, S. Bröcher, and W. Singer. "Different voltage-dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex". In: *Nature* 347 (1990), pp. 69–72. DOI: 10.1038/347069a0. URL: http://dx.doi.org/10.1038/347069a0.

[2] Guo-qiang Bi and Mu-ming Poo. "Synaptic Modifications in Cultured Hippocampal Neurons: Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type". In: *Journal of Neuroscience* 18.24 (1998), pp. 10464–10472. ISSN: 0270-6474. eprint: http://www.jneurosci.org/content/18/24/10464.full.pdf. URL: http://www.jneurosci.org/content/18/24/10464.

[3] Claudia Clopath et al. "Connectivity reflects coding: a model of voltage-based STDP with homeostasis". In: *Nature Neuroscience* 13.3 (2010), pp. 344–352. ISSN: 1097-6256. DOI: 10.1038/nn.2479. URL: http://www.nature.com/doifinder/10.1038/nn.2479.

[4] Anaclet Ngezahayo, Melitta Schachner, and Alain Artola. "Synaptic Activity Modulates the Induction of Bidirectional Synaptic Changes in Adult Mouse Hippocampus". In: *Journal of Neuroscience* 20.7 (2000), pp. 2451–2458. ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.20-07-02451.2000. eprint: http://www.jneurosci.org/content/20/7/2451.full.pdf. URL: http://www.jneurosci.org/content/20/7/2451.

[5] Jean-pascal Pfister and Wulfram Gerstner. "Beyond Pair-Based STDP: a Phenomenological Rule for Spike Triplet and Frequency Effects". In: (2006). Ed. by Y. Weiss, B. Schölkopf, and J. C. Platt, pp. 1081–1088. URL: http://papers.nips.cc/paper/2923-beyond-pair-based-stdp-a-phenomenological-rule-for-spike-triplet-and-frequency-effects.pdf.

[6] Per Jesper Sjöström, Gina G Turrigiano, and Sacha B Nelson. "Rate, Timing, and Cooperativity Jointly Determine Cortical Synaptic Plasticity". In: *Neuron* 32.6 (2001), pp. 1149–1164. ISSN: 0896-6273. DOI: https://doi.org/10.1016/S0896-6273(01)00542-6. URL: http://www.sciencedirect.com/science/article/pii/S0896627301005426.

[7] Julien Vitay, Helge Dinkelbach, and Fred Hamker. "ANNarchy: a code generation approach to neural simulations on parallel hardware". In: *Frontiers in Neuroinformatics* 9 (2015), p. 19. ISSN: 1662-5196. DOI: 10.3389/fninf.2015.00019. URL: https://www.frontiersin.org/article/10.3389/fninf.2015.00019.