Research Article

Modified STDP Triplet Rule Significantly Increases Neuron Training Stability in the Learning of Spatial Patterns

Dalius Krunglevicius

Faculty of Mathematics and Informatics, Vilnius University, Naugarduko Street 24, 03225 Vilnius, Lithuania

Correspondence should be addressed to Dalius Krunglevicius; dalius.krunglevicius@gmail.com

Spike-timing-dependent plasticity (STDP) is a set of Hebbian learning rules which are based firmly on biological evidence. STDP learning is capable of detecting spatiotemporal patterns highly obscured by noise. This feature appears attractive from the point of view of machine learning. In this paper three different additive STDP models of spike interactions were compared in respect to training performance when the neuron is exposed to a recurrent spatial pattern injected into Poisson noise. The STDP models compared were all-to-all interaction, nearest-neighbor interaction, and the nearest-neighbor triplet interaction. The parameters of the neuron model and STDP training rules were optimized for a range of spatial patterns of different sizes by the means of heuristic algorithm. The size of the pattern, that is, the number of synapses containing the pattern, was gradually decreased from what amounted to a relatively easy task down to a single synapse. Optimization was performed for each size of the pattern. The parameters were allowed to evolve freely. The triplet rule, in most cases, performed better by far than the other two rules, while the evolutionary algorithm immediately switched the polarity of the triplet update. The all-to-all rule achieved moderate results.

1. Introduction

Spiking neural networks (SNNs) are based on the physiological function of action potential in the biological cell. Action potential is a brief event where the electrical membrane potential of a cell rapidly rises and falls. In neurons, the trajectory of action potential takes the shape of a spike. SNNs are considered to be the third generation of artificial neural networks (ANNs) [1]. When compared to previous generations of artificial neural networks, SNNs are more complicated and require more computing power to execute a task, so that the application of SNNs in pattern recognition or in other kinds of machine learning is impractical currently. It is reasonable to expect, however, that this is only a temporary obstacle. The main motivation for this paper is to discover how well SNNs perform when used for spatial pattern recognition specifically.

Action potentials in the chemical synapses of a neuron in most animal species trigger the release of chemical messengers called neurotransmitters. Neurotransmitters interact with receptors located on the other side of a synaptic gap

(postsynaptic neuron). Thus sequences of action potentials, or in other words spike trains, can be seen as a form of communication. There are two major approaches to interpreting neural spikes as data. One is rate coding, where data are encoded in an averaged count of spikes over a specific time frame. The other is temporal coding, where data are encoded in precise timing of individual spikes. Findings from biological research suggest that rate coding alone does not have a sufficient data bandwidth to account for the speed of data transfer in some sensory systems in living organisms [2, 3]. In contrast to rate coding, temporal coding has a significantly greater data bandwidth because it requires a minimal time for the neuron to respond. It is debatable whether temporal coding takes place in living neural systems [4], but there is experimental evidence to support the idea [5–8]. Moreover, the discovery of spike-timing-dependent plasticity (STDP) suggests that the timing of spikes is important.

STDP is a form of Hebbian learning induced by tight temporal correlations between presynaptic and postsynaptic spikes. At present several different STDP rules for the

different types of synapses [9] are known. The STDP rule which controls change of synaptic strength in the excitatory-to-excitatory type of chemical synapses appears to target coincidences of incoming spikes. Over a wide range of parameters the additive STDP rule leads to a bimodal distribution of synaptic strengths [10]: presynaptic spikes which induce an action potential in a postsynaptic neuron result in increased synaptic strength, while the strength of other synapses decays. Spikes arriving simultaneously excite the neuron simultaneously and are therefore more likely to cause an action potential in a postsynaptic neuron.

In contrast to additive STDP update, where the amount of change is taken directly from the STDP function and synaptic strength is limited to a range of between 0 and 1, in multiplicative update, where the amount of change is proportional to synaptic strength, synaptic strengths tend to organize themselves in a unimodal distribution [11]. Although it is possible to achieve bimodality using multiplicative update, during the training of the neuron for a spatiotemporal pattern, peaks of distribution are much closer together than is the case in additive update. This paper deals with the additive update rule exclusively.

Probably the simplest example of temporal code is a spatial pattern. Input from multiple afferent neurons in a short temporal window can be conceived as a binary map, where the presence of a presynaptic spike is denoted as 1 and its absence as 0. Alternatively, instead of 1, we can assign a large probability of a spike and instead of 0 a small fixed probability, which would mimic spontaneous neural activity and produce Poisson noise. When such a pattern is repeatedly injected into Poisson noise, STDP learning would result in strong synapses associated with ones and weak synapses associated with zeros. In other words, the individual neuron acts as a coincidence detector [12]. In the simplest possible case this sort of training could be reduced to supervised learning as a simple assignment operation: if the input synapse is associated with the pattern, then set strength to 1; otherwise set it to 0.

In order to build a useful pattern recognition machine based on STDP learning, it is important to understand the limitations of the model; for example, ask the simple question: how small can the spatial pattern be in respect to the number of overall incoming synapses? By making assumptions about presynaptic and postsynaptic processes, such as assuming that presynaptic and postsynaptic spike trains are Poisson distributed, it is possible to gain theoretical insight into the STDP training process. Izhikevich and Desai [13], for example, derived equations for the expected value of change of synaptic weight for multiple models of spike neighborhood interactions for uncorrelated and weakly correlated Poisson-distributed spike trains. Other authors used the Fokker-Plank equation to predict the evolution of synaptic weights [10, 14, 15]. These studies are based on assumptions of a Poisson process or on modeling membrane potential through the Ornstein-Uhlenbeck process [15], again assuming that membrane potential is a Gaussian process. While such methods might be valuable, they are limited to simple distributions, which is not the case in a Poisson noise and spatial pattern mixture. Moreover, the

Spike-Response Model I use in my research cannot produce Poisson-distributed postsynaptic spikes because membrane potential process has a momentum, except when postsynaptic potentials are modeled by unit impulses, which is biologically implausible (see Section 2). Also, membrane potential is not a Gaussian process because of the skewness induced by relative hyperpolarization. Relative hyperpolarization also deforms the distribution of postsynaptic latencies, in some cases resulting in a bimodal distribution.

The main goal of this research was to answer the question of how small the spatial pattern can be relative to the overall amount of presynaptic inputs. Instead of taking a theoretical approach, in this work, I benchmarked three different STDP implementations experimentally. The benchmark was made with respect to the relative size of the pattern and training success rate. For this purpose I used a basic genetic algorithm to optimize neuron and STDP parameters in training for a spatial pattern in a simulation. The size of the pattern, that is, the number of synapses containing the pattern, was gradually decreased from what was a relatively easy task down to the point training failed. The parameters were optimized for each size of pattern. Parameters were allowed to evolve freely, without any restrictions. Such optimization was made for two different setups: in the first setup afferents participating in the pattern fired at a rate of 64 Hz and others fired at 39 Hz; in the second setup all afferents fired at 64 Hz. Also, I conducted limited experiments with firing rates at 39 Hz/39 Hz and 25 Hz/39 Hz. I compared three different additive STDP implementations: all-to-all interaction, nearest-neighbor interaction with immediate pairings [16], and the same nearest-neighbor interaction with triplet update [17]. The results from both the 64 Hz/39 Hz and 64 Hz/64 Hz setups were quite unexpected: in the case of triplet update, the pattern was successfully scaled down to a single synapse; there was no significant degradation in performance over the entire range of patterns when the setup was 64 Hz/39 Hz. In the case of the single synapse transmitting a periodic spike, obviously no spatial pattern remained, and the neuron was tuned to detect either differing rates in the incoming synapse or, as it seems, the periodic occurrence of the spike. The genetic algorithm immediately changed the polarity of the long-term depression of the third spike coefficient. Also, in the case of the 64 Hz/64 Hz setup, the polarities of both third spike coefficients were changed (see Section 2). In the case of all-to-all interaction I did achieve a single synapse as well, but the training success rate was significantly reduced. The simple nearest-neighbor interaction rule reached a definite limit on pattern size and could not be optimized further. In the case of 39 Hz/39 Hz setup the all-to-all rule performed better than the triplet.

2. Materials and Methods

The neuron was trained for spatial patterns of different sizes. Neuron and STDP parameters were optimized for each size of pattern (see Section 2.1). Simulations were executed in discrete time steps at 1 ms precision. The pattern was created by a number of selected input neurons firing at the same time after each 40 ms of simulation (Figure 1). Also,
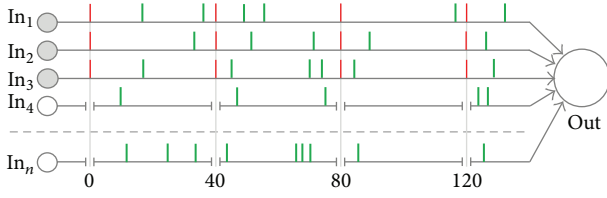
FIGURE 1: Sample pattern. Spikes generated by Poisson process are green; injected spatial pattern is red. Afferents which do not belong to the pattern did not fire during pattern exposure.

all neurons fired spontaneously with a fixed probability at each simulation step, thus producing Poisson noise. During pattern exposure all other neurons did not fire; thus at the time of the pattern exposure the variance of input spikes was zero. I did this because if I had not, the expected value of input spikes would be greater than the expected value during Poisson periods and that would make the training too easy. Three different spike neighborhood rules were compared: all-to-all, nearest-neighbor with immediate pairings [16], and nearest-neighbor with triplet update [17].

This experiment is similar to the one conducted by Masquelier and colleagues [18]. The key difference is that I used a spatial rather than a spatiotemporal pattern, and the patterns were inserted into the noise signal at a regular frame rate.

I measured training performance in three different setups of Poisson noise. Initially I set the probability of a noisy spike to $p = 0.04$ in all afferents. In this setup, the neurons which participated in the pattern fired more frequently than those which did not, at 64 Hz and 39 Hz, respectively. In the case of a spatiotemporal pattern of sufficient duration it is possible to eliminate this difference in firing rate, if the count of spikes in the pattern from individual synapses is equal to the expected value of Poisson noise. This is not the case, however, for spatial patterns: in order to maintain equal firing rates, synapses which belong to the pattern must fire less frequently during Poisson periods, thus reducing noise and, presumably, resulting in easier training. On the other hand, the difference in firing rates also has a great influence on the training: the heuristic optimization I used could tune the neuron with the triplet rule to detect increased firing rate instead of spatial pattern. This was the case when the pattern was sufficiently small, but not the case when it was large enough to cause the postsynaptic spike (see Section 3). Such a mix of spatial and rate coding is compatible to some extent with observations of the auditory cortex of primates [8], where a mix of different coding systems seems to convey more information than spatial or rate coding alone.

Later I reran the optimization with a 64 Hz/64 Hz setup, where the firing rate of afferents not participating in the pattern recognition was increased to 64 Hz by setting the probability of a noisy spike to $p = 64/975 \approx 0.066$. Here the triplet rule also performed better than the other two, but the rate of success was lower, and the behavior of the neuron when the pattern became small was very different (see Section 3).

I also conducted limited experiments with reduced noise in the afferents which did participate in the pattern at 39 Hz/39 Hz and 25 Hz/39 Hz. Here the probabilities of a noisy spike were set to $p = (39 - 25)/975 \approx 0.014$ and 0, respectively. In this case the triplet rule lost its advantage over all-to-all but still performed significantly better than the nearest-neighbor rule.

2.1. Heuristic Optimization. The basic idea was to discover the lower limits of STDP training with respect to the spatial pattern or in other words to answer the question of how small the spatial pattern can be. Since additive STDP tends to produce a bimodal distribution of synaptic strengths, the idea was to maximize the difference between the strengths of synapses which transmit the pattern and those which do not. In addition, the neuron must remain responsive at the end of the training and ideally selective only to the pattern. Instead of minimizing the firing latencies of the trained neuron, for the sake of simplicity, I made the assumption that the neuron firing rate should be approximately the same as the rate of pattern injection. For this purpose, I introduced a Gaussian component into the objective function:

$$
f = \begin{cases} \sum_t e^{-(\lambda-\xi)^2/\delta} \Delta\mu_w & \text{if } \Delta\mu_w > 0 \\ \sum_t \Delta\mu_w & \text{if } \Delta\mu_w \le 0, \end{cases} \tag{1}
$$

where $\Delta\mu_w$ is the observed difference between the means of strengths of synapses which were associated with the pattern and those which were not; λ is the observed firing rate (times per second); $\xi = 25$ is the target firing rate; and $\delta = 20$ defines the tolerated deviation from the target rate. At the beginning of the training all synaptic strengths were set to the same value w_0, so that at the very beginning of the training the value of $\Delta\mu_w$ was zero. The value of the objective function was the sum of observations at each time step in the simulation. In this way, the performance of the training was taken into account from the very beginning of the simulation, and, in this manner, the speed of the training was also increased by maximizing the objective function. The heuristic search to maximize the objective function (1) was executed in 7-dimensional space for nearest-neighbor and all-to-all rules and in 11-dimensional space for the triplet rule. The optimized parameters were ϑ, w_0, $w_{\min}$, η, A_{pre}, T_{pre}, and T_{post}. For the triplet rule there were four additional parameters: A_{pre3}, A_{post3}, T_{pre3}, and T_{post3} (see equations below). For the heuristic search I used a very basic genetic algorithm. There were 100 agents, and after each trial 60 agents were replaced by the offspring of the top 20 performers. Offspring were generated from the parent agent by applying normally distributed mutations. The mean of the normal distribution was the parent value; the standard deviation for mutations was 1 for time dimensions (T_{pre}, T_{pre3}, T_{post}, and T_{post3}) and 0.01 for all other dimensions. Each agent in each generation executed 10 independent trials and the values of the objective function were added together from all 10 trials. Each trial took 5,000 iterations (milliseconds). The heuristic search was executed for 1500 generations. There were 300 afferents in each agent.
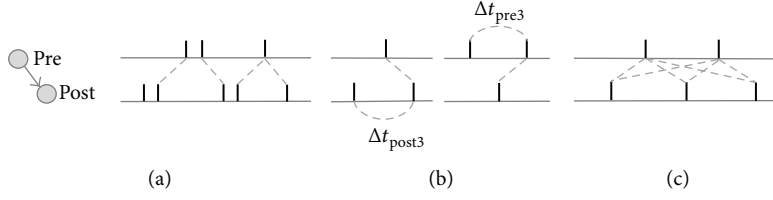
FIGURE 2: STDP neighborhood rules. (a) Nearest-neighbor with immediate pairings. (b) Triplet update. (c) All-to-all.

In my first 64 Hz/39 Hz experiment initial pattern size n was set to 24, equal to twice the expected spike count generated by Poisson noise. Pattern size was then decreased to 12, 8, 4, 2, and 1.

In the 64 Hz setup, pattern sizes were 19, 15, 12, 8, and 4. Pattern size $n = 19$ approximates the expected spike count generated by Poisson noise, which was 19.2. Here for the initial conditions I reused the parameters obtained for $n = 24$ in the 64 Hz/39 Hz setup, with an exception for the triplet rule (see Section 3).

For the 39 Hz/39 Hz and 25 Hz/39 Hz experiments, initial parameters were taken from the 64 Hz/39 Hz setup $n = 8$ results (an exception was again made for the triplet rule; see Section 3 for details) and the pattern size remained the same; $n = 8$.

After optimization of parameters, the success rate of training was evaluated by training the neuron for the same pattern 1,000 times. The success criterion was $\Delta\mu_w \geq 0.3$ and $12 < \lambda < 50$ at the end of the training (see (1)).

I reran the genetic optimization several times and the results were all similar.

2.2. The Neuron Model.

For the neuron model I used a version of the Spike-Response Model (SRM) [19]. The same model was used in my previous work [20]. In the particular model, a membrane potential $u(t)$ at time t is given by

$$
u(t) = \begin{cases} 5\vartheta & \text{if } t = t_{\text{spike}} \\ -W_r e^{(t_{\text{spike}} - t)/T_r} + \sum \epsilon(t) & \text{otherwise,} \end{cases} \quad (2)
$$

where W_r and T_r define the amplitude and duration of relative refraction and t_{spike} is the time of the last spike generated by the given (postsynaptic) neuron. At the beginning of the simulation t_{spike} was always set to -10^6. The spike occurs when the membrane potential reaches the threshold value ϑ. The postsynaptic potential $\epsilon_j(t)$ arriving from an individual synapse j is given by

$$
\epsilon_j(t)
$$
$$
= w_j \left(e^{-\Delta t/T_m} \left(1 + x_{m_j}(t)\right) - e^{-\Delta t/T_s} \left(1 + x_{s_j}(t)\right) \right), \quad (3)
$$

where w_j is the strength of the synapse; T_m and T_s are the time constants; and $\Delta t = t - t_{\text{pre}}$, where t_{pre} is the time of the last presynaptic spike. Note that when x_m and x_s are zero, (3) is reduced to the simple function of two exponentials: $\epsilon_j(t) =$

$w_j(\exp(-\Delta t/T_m) - \exp(-\Delta t/T_s))$. Variables x_m and x_s are introduced only to simplify the integration of exponentials during the simulation and are given by

$$
x_{m_j}(t)
$$
$$
= \begin{cases} \dfrac{w_{j(t-1)}}{w_{j(t)}} e^{-\Delta\tau/T_m} \left(1 + x_{m_j}(t-1)\right) & \text{if } t = t_{\text{pre}} \\ x_{m_j}(t-1) & \text{if } t \neq t_{\text{pre}} \end{cases} \quad (4)
$$

$$
x_{s_j}(t) = \begin{cases} \dfrac{w_{j(t-1)}}{w_{j(t)}} e^{-\Delta\tau/T_s} \left(1 + x_{s_j}(t-1)\right) & \text{if } t = t_{\text{pre}} \\ x_{s_j}(t-1) & \text{if } t \neq t_{\text{pre}}, \end{cases}
$$

where $\Delta\tau$ is the time difference between the previous and the last presynaptic spikes. Initial values for x_m and x_s are zero and are updated only at the moment of the presynaptic spike. For derivation of (4) please see [20].

During all experiments time constants were set to $T_r = 10$, $T_m = 10$, and $T_s = 0.5$. W_r was dependent on a threshold value and was set to 2ϑ.

2.3. Plasticity.

Spike-timing-dependent plasticity is a form of Hebbian learning; it can be modeled as a function of time difference between presynaptic and postsynaptic spikes; the value of the function is the amount of change in synaptic strength. Persistent strengthening of a synapse is referred to as long-term potentiation (LTP) and persistent reduction of synaptic strength is referred to as long-term depression (LTD).

In this work I investigate only one of the STDP rules that is known to be common in excitatory-to-excitatory synapses. It should be noted that a number of different STDP rules have been discovered which vary depending on synapse type or even according to their position on the dendrite [9, 12, 21, 22]. Neighborhood functions used for comparison are represented in Figure 2. Triplet update (Figure 2(b)) was used in combination with the nearest-neighbor interaction in Figure 2(a). STDP updates were modeled by (5), (6), and (7). Nearest-neighbor is

$$
\Delta w_j = \begin{cases} \eta e^{-\Delta t/T_{\text{post}}} & \text{if } \Delta t > 0 \\ -\eta A_{\text{pre}} e^{-\Delta t/T_{\text{pre}}} & \text{if } \Delta t < 0 \\ 0 & \text{if } \Delta t = 0. \end{cases} \quad (5)
$$

Nearest-neighbor with the triplet update is

$$\Delta w_j$$

$$= \begin{cases} \eta \left(1 + A_{\text{post3}} e^{-\Delta t_{\text{post3}}/T_{\text{post3}}}\right) e^{-\Delta t/T_{\text{post}}} & \text{if } \Delta t > 0 \\ -\eta \left(A_{\text{pre}} + A_{\text{pre3}} e^{-\Delta t_{\text{pre3}}/T_{\text{pre3}}}\right) e^{-\Delta t/T_{\text{pre}}} & \text{if } \Delta t < 0 \\ 0 & \text{if } \Delta t = 0. \end{cases} \quad (6)$$

All-to-all is

$$\Delta w_j = \begin{cases} \eta e^{-\Delta t/T_{\text{post}}} \left(1 + y_{\text{post}}\right) & \text{if } \Delta t > 0 \\ -\eta A_{\text{pre}} e^{-\Delta t/T_{\text{pre}}} \left(1 + y_{\text{pre}}\right) & \text{if } \Delta t < 0 \\ 0 & \text{if } \Delta t = 0, \end{cases} \quad (7)$$

where Δw_j is the amount of change of strength of an individual synapse; η is the training step; Δt is the time difference between postsynaptic and presynaptic spikes; and A_{pre}, A_{pre3}, A_{post3}, T_{pre}, T_{pre3}, T_{post}, and T_{post3} are the parameters which control amplitudes and slopes of STDP functions. Variables y_{post} and y_{pre} were computed in the same way as the x variables in the neuron model in (4). The only difference was that weights were not present in this case. Synaptic strengths were kept between 1 and w_{min}, where $1 > w_{\text{min}} > 10^{-6}$ in order to avoid infinity in (4).

3. Results

The results from the 64 Hz/39 Hz and 64 Hz experiments are presented in Figure 3. The triplet rule in both setups performed much better than its competitors, although in the 64 Hz triplet experiment there was a significant degradation in performance for $n = 8$. The simple nearest-neighbor rule performed the worst, and heuristic search failed to find suitable parameters for the $n = 4$ spike pattern in the 64 Hz/39 Hz setup and for the $n = 12$ in the 64 Hz setup.

It has to be stated that in the 64 Hz setup (Figure 3(b)) the genetic algorithm initially failed to find the point where the triplet rule would perform better than all-to-all, and when the pattern size was 15, it performed worse than nearest-neighbor. This could not be the global optimum because nearest-neighbor is a special case of the triplet, where A_{pre3} and A_{post3} are zero; therefore at the optimal point the triplet rule should perform at least equally to nearest-neighbor. So here optimization got stuck in a local optimum. To validate this, I used the nearest-neighbor parameters obtained for $n = 15$ as initial parameters for the triplet, except A_{pre3} and A_{post3} that were set to zero; $T_{\text{pre3}} = 2T_{\text{pre}}$ and $T_{\text{post3}} = 2T_{\text{post}}$. The results were significantly better: the triplet performed better than the other two. In order to eliminate possible unfair competition, I reran genetic optimization for nearest-neighbor and all-to-all for 3,000 generations, with no success in improving the parameters. Although these results cannot be conclusive, they strongly suggest that the triplet rule can perform better.

When pattern size was relatively large, results from both the 64 Hz/39 Hz and 64 Hz experiments were quite similar: the trained neuron was selective to the pattern and fired mostly after the pattern time with 2-millisecond latency. The latency was caused by the PSP kernel function chosen (see (3)). In the 64 Hz/39 Hz setup the triplet rule retained selectivity down to the $n = 4$ pattern and in the case of all-to-all this was $n = 8$ (see Figure 4); in the 64 Hz setup selectivity was lost sooner: the triplet rule retained selectivity down to $n = 8$ and $n = 12$ for the all-to-all rule.

When the pattern became too small, the genetic algorithm found conditions where STDP training would result in certain equilibria of synaptic strengths, and consequently the neuron firing rate was more or less constant, but even so, the synaptic strengths associated with the pattern tended to grow close to the maximal value, which was 1, while the remaining strengths were distributed above the minimal value. The neuron was not selective because the combined strength of the synapses associated with the pattern was not sufficient to cause the postsynaptic spike. Particularly when $n = 1$, a spatial pattern does not even exist. In the case of the simple nearest-neighbor rule, this kind of behavior was not observed.

It must be stated that in the 64 Hz setup genetic optimization could not improve the training success rate for the triplet and all-to-all, as the pattern became too small and neuron was not selective to it. In Figure 3(b) the dashed line indicates that optimization was discontinued, and for measuring success rate parameters were taken from previous optimization results, which were $n = 4$ for the triplet and $n = 8$ for the all-to-all.

When the pattern is relatively small (see Figure 3, black markers), training with the parameters obtained from the 64 Hz/39 Hz and 64 Hz experiments for the triplet rule results in very different behaviors. Parameters from the 64 Hz/39 Hz experiment were tuned to detect an increased rate: I replaced the spatial pattern with a pure Poisson process with firing rates of 64 Hz and 39 Hz, respectively, and repeated the triplet experiment with the same parameters. Training failed when $n > 4$, and thus a coincidence of spikes was required to train the neuron under the given parameters. Training was successful, however, when $n \leq 4$, and therefore when the pattern was small, synapses grew stronger because of increased input firing rate, not because of coincidences of input spikes. This, however, was not the case with the 64 Hz experiment. I made a few tests with the triplet rule and $n = 1$ pattern size. In my experiment spatial pattern consists of spikes and gaps. When spikes and gaps are replaced with the pure 64 Hz Poisson noise, training obviously fails; the success rate is simply equal to the measured probability for a random synapse to grow stronger than the mean value plus 0.3 (see Section 2.1 for training success criteria). When only gaps were replaced with Poisson noise, the training success was reduced from 0.99 to 0.7. When only spikes in the pattern were replaced with noise, but gaps were still persistent, the training success rate reduced to 0.14, which seems to be slightly above random chance (measured probability of a random chance of success was ~0.12). This suggests that parameters were tuned to detect deformations of a Poisson process, and these deformations could be induced by either a periodic spike or periodic gap. However, I cannot claim with certainty that STDP can detect periodic gaps in a Poisson process.
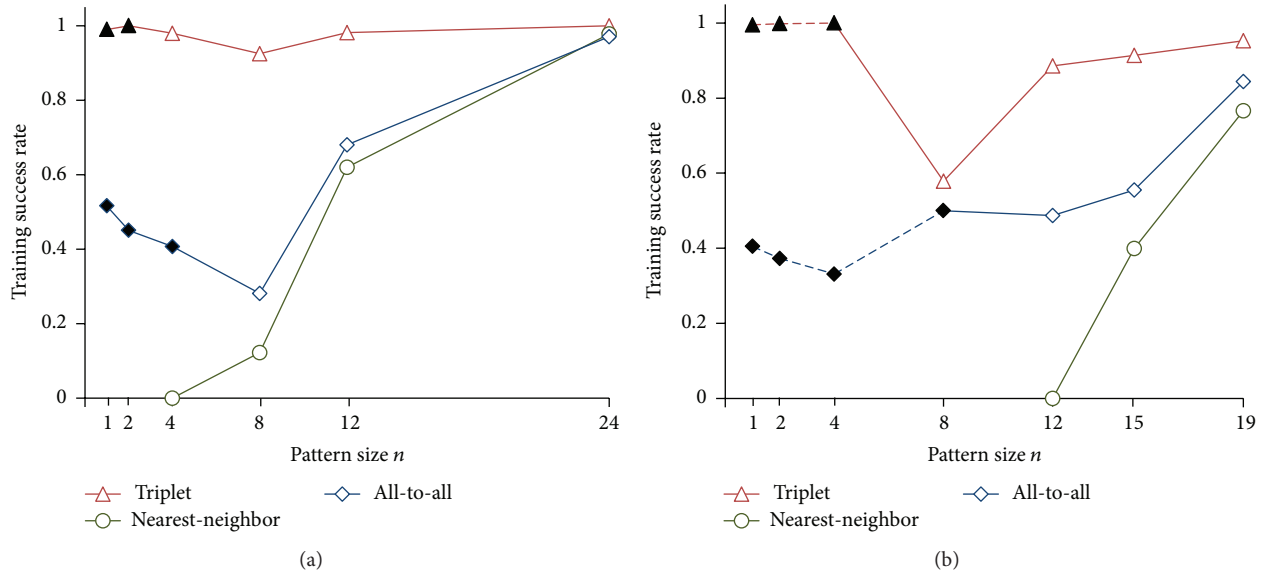
(a)

(b)

FIGURE 3: Training success rate versus pattern size. Black markers denote the training when synaptic strengths were bimodal, but the neuron was not selective to the pattern. (a) Results from the 64 Hz/39 Hz setup. (b) Results from the 64 Hz setup. Dashed lines indicate that there was no heuristic optimization made, but previous optimized parameters were reused.
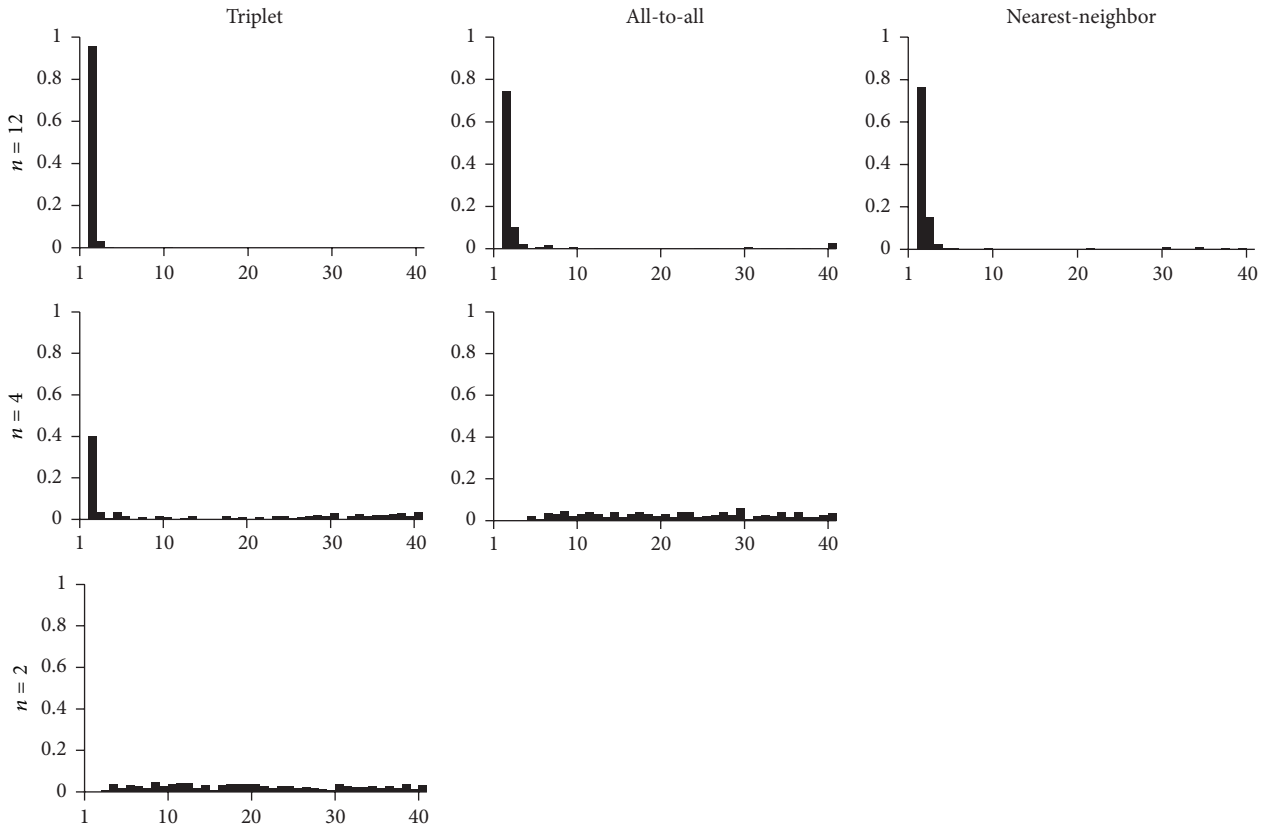

FIGURE 4: Distribution of trained neuron response latencies. Zero point is the sample pattern occurrence time. Results are from the 64 Hz/39 Hz setup.
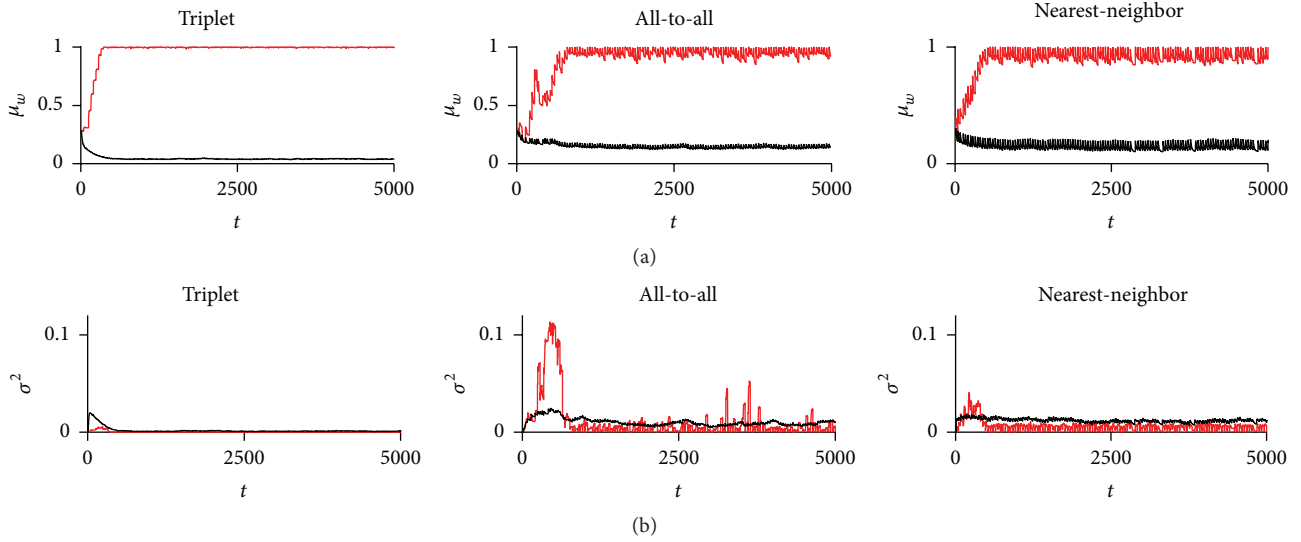
FIGURE 5: Evolution of synaptic strengths. Results were taken from singular successful trainings when pattern size was $n = 12$ for the 64 Hz/39 Hz setup. (a) is means of synaptic strengths; (b) shows variances. Red denotes synapses associated with the pattern; black denotes synapses not associated with the pattern.

I also conducted an additional test with the triplet rule in the 64 Hz setup: I replaced the spatial pattern with a spatiotemporal one, by distributing spikes of a pattern in time with 10 ms latency; gaps from noisy afferents were removed. In this case, training was successful for $n = 4$ with a success rate of ~0.5 and for $n = 8$ of ~0.2. Training failed for $n = 12$. This indicates that parameters obtained for relatively small patterns were appropriate for detecting certain deformations of a Poisson process of spikes from an individual afferent, but not coincidences of spikes.

In the all-to-all rule and the small pattern in the 64 Hz/39 Hz setup where the STDP window was inverted (see Figure 3(a), black markers; see Table S1 of the Supplementary Material available online at http://dx.doi.org/ 10.1155/2016/1746514), training failed when the pattern was replaced with pure noise. I tried to preserve only the gaps or only the periodic spike, with no success in training. In the 64 Hz setup, however, removing gaps and replacing the spatial pattern with a spatiotemporal one with 10 ms latency between spikes boosted the performance of the all-to-all rule to a success rate above 0.8 for $n \leq 12$. It should be noted that when $n = 8$ and $n = 12$ the spatiotemporal pattern overlaps, so that there were coincidences of two or three spikes correspondingly. Training failed when $n = 15$.

The nearest-neighbor rule in all cases could only be trained to the spatial pattern, and all attempts to replace spatial pattern with Poisson noise or a spatiotemporal pattern resulted in training failure.

It is worth noting that the number of input neurons easily could be scaled up by any factor, by scaling the threshold value ϑ and the size of the spatial pattern by the same factor and keeping the STDP parameters unchanged, except for training step η, which requires additional tuning in the case of the 64 Hz/39 Hz setup and $n = 1$ (in this particular case, I had to change the value of η from 0.662 to 0.9; otherwise

training was not successful). This nonlinear dependency of the training step needs further research. In the case of a small pattern in nonselective mode, the pattern size may remain unchanged after scaling. For the 64 Hz/39 Hz setup I successfully scaled the triplet model up by a factor of 20, which is 6,000 inputs with pattern sizes of $n = 1$ and $n = 8$. The neuron was trained thus without any noticeable degradation of the success rate of the training. In fact I observed a slight improvement in the training success rate. That is, the neuron was able to find a single synapse with increased firing rate among the 5,999 others and to learn a spatial pattern made by 160 input neurons among the 5,840 others. I also successfully repeated the same scaling by a factor of 20 with the 64 Hz setup, for $n = 1$ and $n = 19$.

Figure 5 is a comparison of the synaptic strength evolutions of successful trainings where pattern size was $n = 12$ for the 64 Hz/39 Hz setup. Results were gathered from single runs of 5,000 ms duration. The three columns represent the three rules, and (a) shows means of synaptic strengths and (b) shows variances. In the case of the triplet rule, synaptic strengths were much more stable. When training with the triplet rule, variance of strengths was almost one-tenth that of nearest-neighbor or all-to-all. Comparing the all-to-all rule to the simple nearest-neighbor rule, synaptic strengths were a bit more stable in the case of the all-to-all. Strength evolutions in the 64 Hz setup were very similar (see Supplementary Material, Figure S1).

From the parameters obtained through heuristic optimization, we can make a few interesting observations. In the case of the triplet rule and the 64 Hz/39 Hz setup (see Figure 6; Supplementary Material, Table S1), LTP occurred at the left side of the STDP window (Figure 7), that is, where $t_{\text{pre}} > t_{\text{post}}$, and presynaptic spikes were closely correlated to postsynaptic ones. The right side of the STDP window shows a very steep slope, and its amplitude diminishes as the pattern
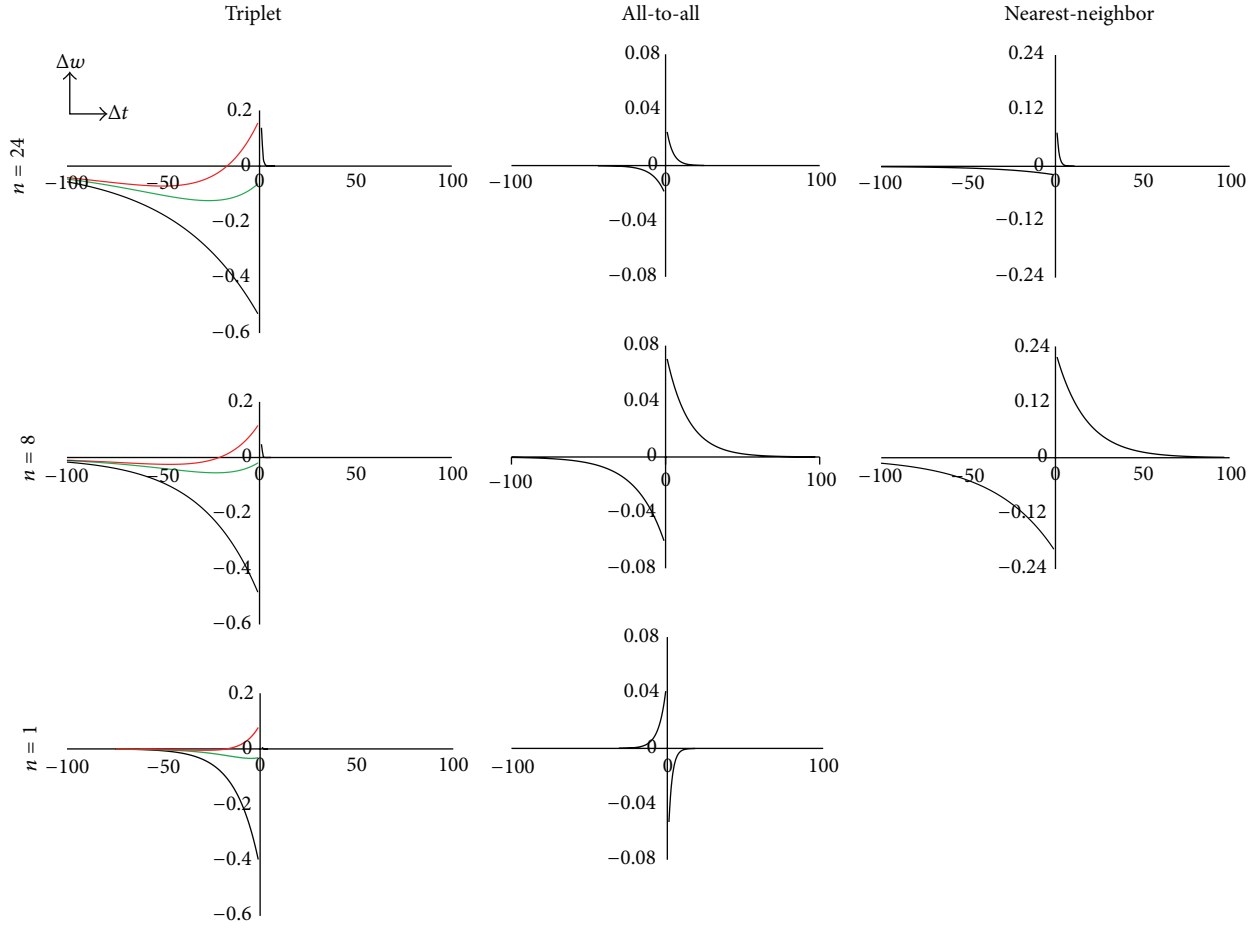
FIGURE 6: Optimized STDP windows for the 64 Hz/39 Hz setup. In the case of the triplet rule, red lines denote $\Delta t_{\text{pre3}} = \Delta t + 1$, green lines $\Delta t_{\text{pre3}} = \Delta t + 25$, and black lines $\Delta t_{\text{pre3}} = \infty$.
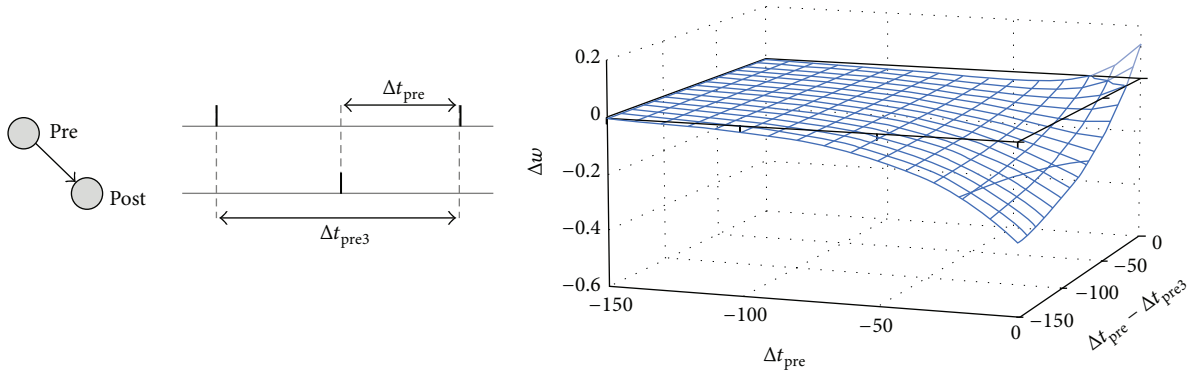

FIGURE 7: The left side of the STDP triplet update window. Values obtained by genetic optimization when pattern size is $n = 12$, 64 Hz/39 Hz setup.

size becomes smaller; in the case of the pattern size $n = 1$, the right side of the STDP window showed very little or no influence.

In the 64 Hz setup LTP also occurred at the left side of STDP window, but the value of this LTP was significantly lower (see Figure 8). Also, in comparison to the 64 Hz/39 Hz

experiment, the right side of STDP window was not diminished, and LTD occurred on the right side when presynaptic and postsynaptic spikes were close in time.

Also, in the case of all-to-all rule and 64 Hz/39 Hz setup, LTP and LTD switched places when pattern became small at $n = 4$, and at the same time the neuron lost its selectivity
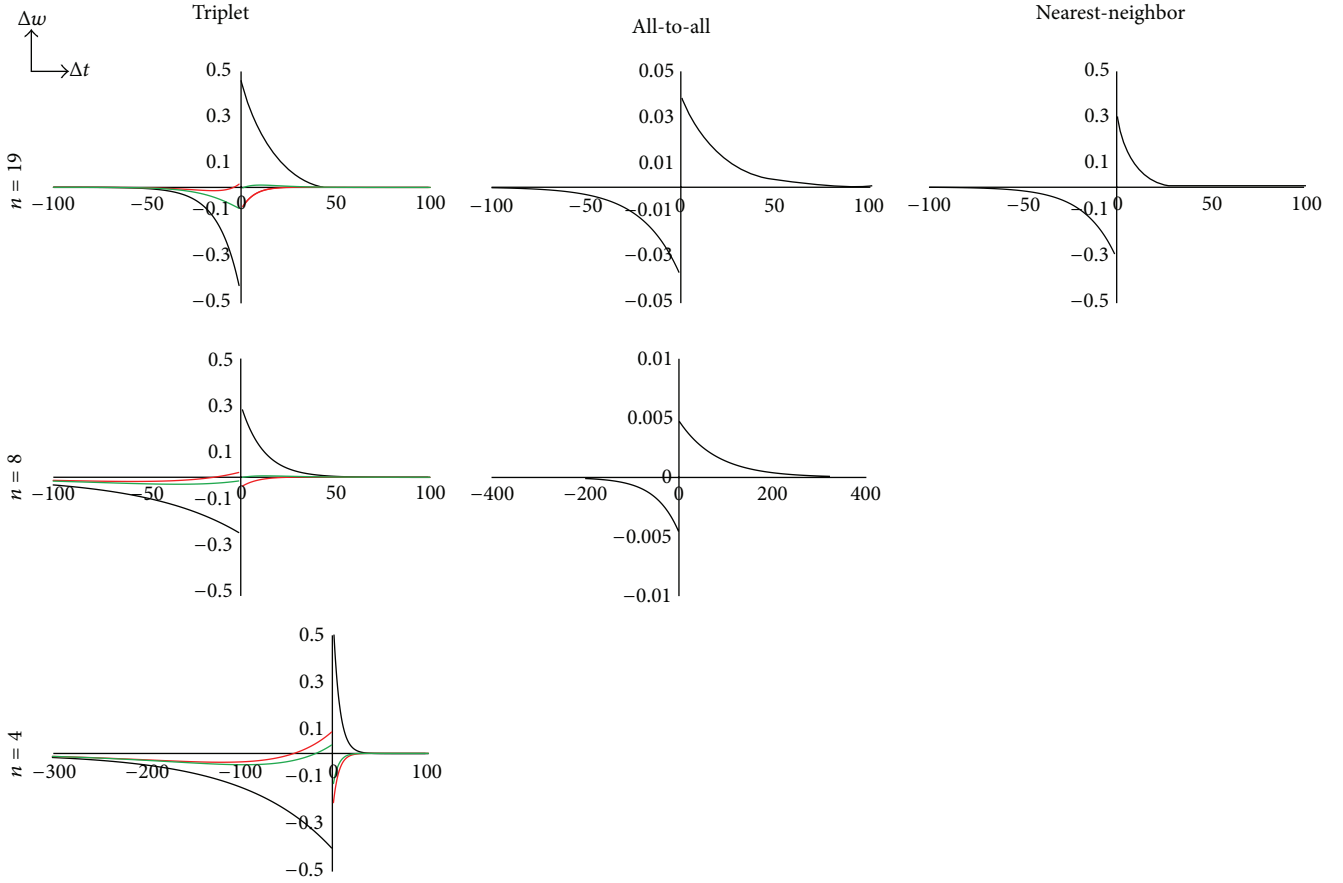
FIGURE 8: Optimized STDP windows for the 64 Hz setup. In the case of the triplet rule, red lines denote $\Delta t_{\text{pre3}} = \Delta t + 1$, green lines $\Delta t_{\text{pre3}} = \Delta t + 25$, and black lines $\Delta t_{\text{pre3}} = \infty$.

to the pattern (see Figure 6). It is interesting to note that switches of LTP and LTD in the synapses of the same neuron have been observed in biology: synapses distant from the soma have different STDP window polarity than synapses proximate to the soma. This has been observed in the visual cortex [23] and in the barrel cortex [24]. Another interesting observation about the all-to-all rule is that, in the case of a small pattern, the neuron fires at a persistent rate, despite the w_{min} value approaching close to zero. This indicates that such an inverted STDP window is capable of attaining equilibrium in synaptic strengths when exposed to Poisson noise. It was also interesting to note that the behavior of the inverted STDP window for all-to-all interaction contradicted the equilibrium properties predicted by the Izhikevich and Desai equation [13]: equilibria for parameters found by heuristic optimization should not be stable. The Izhikevich and Desai equations, however, are based on the assumption of a Poisson-distributed postsynaptic spike train, which was not the case for an SRM neuron with relative refraction. At this time I have no solid explanation for why the neuron retained a stable firing rate; this requires additional research.

In the 64 Hz setup, the all-to-all rule did not switch the polarity of LTP and LTD (see Figure 8). The behavior,

however, was somewhat similar: the all-to-all rule attained equilibria in synaptic strengths and postsynaptic neuron fired at persistent rate.

I conducted a limited experiment with the 39 Hz/39 Hz and 25 Hz/39 Hz setups, where noise in afferents participating in the pattern produced reduced Poisson noise or no noise at all (see Figure 9; Supplementary Material, Table S3).

In this experiment, the all-to-all and nearest-neighbor rules resulted in an increased success rate as the noise of the afferents participating in the pattern was reduced, but the success rate of the triplet rule decreased at the point of 39 Hz; the triplet rule performed worse than the all-to-all rule, but better than nearest-neighbor.

Initially in this experiment I took the optimized parameters from the 64 Hz/39 Hz and $n = 8$ results as the initial conditions for all three rules and evolved parameters with changed firing rate, but maintaining $n = 8$. Later, in the same way as in the 64 Hz/64 Hz setup, I reused optimized parameters from the nearest-neighbor for initial conditions for the triplet. This helped improve the performance of the triplet, but not to the point where it could perform better than all-to-all at the 39 Hz. At 25 Hz triplet and all-to-all had a very similar success rate.
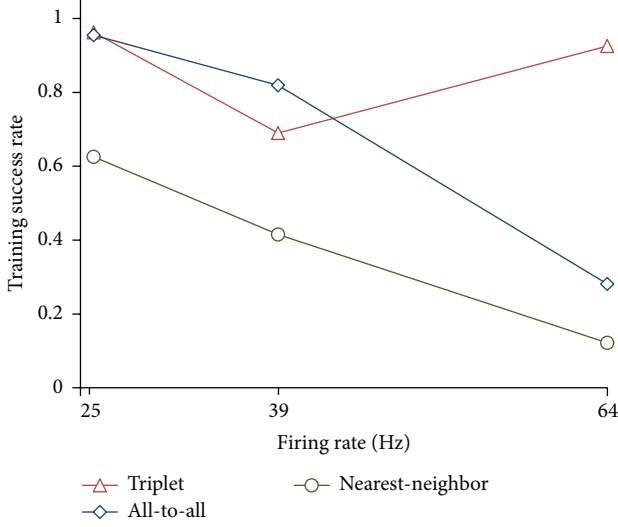
FIGURE 9: Training success rate versus firing rate in the 39 Hz/39 Hz and 25 Hz/39 Hz setups; $n = 8$. The firing rate of all eight afferents participating in the pattern was reduced from 64 Hz to 39 Hz and 25 Hz. Other afferents fired at 39 Hz. Success rate values at 64 Hz are taken from Figure 3(a).

4. Discussion

At the moment, the practical application of STDP learning to real-world data is problematic. One cannot simply take a set of real-world processes, convert these into parallel spike trains, and expect STDP to detect correlations. The parameters of the neuron must be tuned according to the properties of the input spike trains and/or vice versa. From a practical point of view this makes little sense: if good prior knowledge about the data is required before applying STDP training, then one may use other traditional tools which are much more efficient than STDP. The only way to make STDP useful for real-world applications is to build adaptive neural networks significantly more sophisticated than the one I used in this paper. Even such a simple task as differentiating mutually inclusive spatial patterns, for example, requires a complicated neural circuit. I presented one possible solution for mutually inclusive patterns in my previous work [20]. The results of my experiment do not suggest the use of the triplet STDP implementation for direct applications, but rather that the triplet might be a good candidate for consideration in the design of artificial spiking networks.

In this paper I applied STDP learning to static spatial patterns. This is not necessarily the best application for STDP, and it is known that STDP can be applied for different problems. For instance, STDP can detect rate-modulated patterns [25], which are more biologically plausible than static spike patterns. Also, it was shown that STDP in a winner-take-all network can approximate expectation-maximization [26].

It should be said that this work covers only a fraction of the variety of the phenomenological models of STDP. Spike pairings in the nearest-neighbor can be implemented differently by using symmetric or postsynaptic centered

interpretations [11]; I did not research reduced multiplicative update [27]; also I did not include the all-to-all version of the triplet [17].

The main purpose of this paper is to demonstrate that changing the sign of additional trace variables of triplet STDP implementation potentially can result in a far better coincidence detector than STDP implementations based on two trace variables. The triplet rule (Figure 2(b)) was originally suggested by Pfister and Gerstner [17]. In their original work they successfully reproduced STDP behavior found in biological neurons, in both the visual cortex [28] and hippocampal culture [29]. Pfister and Gerstner used positive values for A_{pre3} and A_{post3} (see Section 2.3, (7)). In the case of training for spatial patterns, however, genetic optimization immediately changed the polarity of A_{pre3}; and in the 64 Hz setup, it changed A_{post3} as well (see Supplementary Material, Tables S1 and S2). Taking a closer look at the LTD side of the original triplet rule (Figure 2(b)), it is evident that positive A_{pre3} increases LTD in cases where the previous presynaptic spike was strongly correlated to the postsynaptic one and therefore reduces the existing correlation. This feature, while it might be biologically plausible, has a negative impact on training for spatial patterns. If A_{pre3} is set to a negative value, the result is the opposite, and LTD is either lessened or replaced by LTP. Moreover, this setup of the triplet rule favors spike triplets in a window of a specific duration and therefore is suitable for selecting synapses with a higher spiking rate because the higher the rate is, the higher the probability of the occurrence of a triplet in a smaller temporal window is. Thus it was not surprising in the least that the heuristic search changed the polarity of A_{pre3}. What was surprising, however, was the magnitude of positive impact on training overall. At the same time, I have no solid explanation for why the genetic algorithm changed the polarity of A_{post3} in the case of the 64 Hz setup and why exclusively for this setup. Such negative A_{post3} would cause LTD when two postsynaptic spikes are close in time and presynaptic spike is closely correlated to the last postsynaptic one (see Figure 8, triplet). At this time I can only speculate that this LTD would be induced mostly when the postsynaptic neuron fires frequently, thus helping prevent too high a firing rate.

Another interesting observation that follows from my experiment is that the triplet implementation of STDP can achieve stable equilibria of synaptic strengths when exposed to a Poisson process of input spikes. At the same time, STDP can detect an increased spiking rate or a certain deformation of a Poisson process even in a single synapse, when the influence of that individual synapse on the overall postsynaptic membrane potential is negligible. In such a case the neuron is incapable of encoding the data, which makes it difficult to apply this feature to competitive learning, for example, in a single winner-take-all circuit, such as the one used by Masquelier and colleagues [30]. There are no reasons why synaptic weights cannot be modified after the training by increasing the contrast of synaptic weights, however, thus making the neuron selective to the input of even a single synapse.

When the pattern was relatively small (Figure 3, black markers), the neuron was incapable of detecting a spatial

pattern; thus it failed at coincidence detection, even in the case of inverted all-to-all rule in the 64 Hz/39 Hz setup, since training was successful at $n = 1$. Nevertheless, it demonstrates a variety of interesting properties of STDP learning which require additional research. It is important to understand that STDP training may increase synaptic strength for multiple reasons. Without a good understanding why and when synaptic strengths grow or decay, the interpretation of STDP training results could be problematic.

The results of this experiment need to be accepted with caution because the results of heuristic optimization are only approximate and there is no proof that heuristic optimization approached global optima rather than getting stuck in a local optimum. While it shows that using the triplet rule makes very good results possible, it does not prove that it is impossible to achieve better results with the all-to-all or nearest-neighbor rule. At this stage of research the amount of data is insufficient to draw solid conclusions other than the fact that the triplet rule can perform extremely well under certain conditions. For this reason, and to limit the scope of the paper, I have not represented the dynamics of variables during genetic optimization. The results of this experiment should therefore be accepted as evidence, but not as proof.

The biological plausibility of the triplet parameters discovered is questionable, but this experiment was not intended to validate biological hypotheses. The heuristic search discovered parameters appropriate for a mixture of the Poisson process and periodic spatial patterns. Such conditions do not necessarily exist in the biological realm. In the case of the Poisson process, for example, intervals between input spikes are distributed exponentially, while this is questionable in the case of the actual postsynaptic potential process [31]. The Spike-Response Model with relative refraction cannot even produce a Poisson spike train. The results of this experiment should nonetheless be interesting from the perspective of machine learning.

Competing Interests

The author declares having no competing interests.

Acknowledgments

References

[1] W. Maass, "Networks of spiking neurons: the third generation of neural network models," *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, 1997.

[2] W. Gerstner, R. Kempter, J. L. van Hemmen, and H. Wagner, "A neuronal learning rule for sub-millisecond temporal coding," *Nature*, vol. 383, no. 6595, pp. 76–78, 1996.

[3] R. Van Rullen and S. J. Thorpe, "Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex," *Neural Computation*, vol. 13, no. 6, pp. 1255–1283, 2001.

[4] E. T. Rolls, N. C. Aggelopoulos, L. Franco, and A. Treves, "Information encoding in the inferior temporal visual cortex:

contributions of the firing rates and the correlations between the firing of neurons," *Biological Cybernetics*, vol. 90, no. 1, pp. 19–32, 2004.

[5] Y. Prut, E. Vaadia, H. Bergman, I. Haalman, H. Slovin, and M. Abeles, "Spatiotemporal structure of cortical activity: properties and behavioral relevance," *Journal of Neurophysiology*, vol. 79, no. 6, pp. 2857–2874, 1998.

[6] J.-M. Fellous, P. H. E. Tiesinga, P. J. Thomas, and T. J. Sejnowski, "Discovering spike patterns in neuronal responses," *Journal of Neuroscience*, vol. 24, no. 12, pp. 2989–3001, 2004.

[7] R. VanRullen, R. Guyonneau, and S. J. Thorpe, "Spike times make sense," *Trends in Neurosciences*, vol. 28, no. 1, pp. 1–4, 2005.

[8] C. Kayser, M. A. Montemurro, N. K. Logothetis, and S. Panzeri, "Spike-phase coding boosts and stabilizes information carried by spatial and temporal spike patterns," *Neuron*, vol. 61, no. 4, pp. 597–608, 2009.

[9] N. Caporale and Y. Dan, "Spike timing-dependent plasticity: a Hebbian learning rule," *Annual Review of Neuroscience*, vol. 31, pp. 25–46, 2008.

[10] J. Rubin, D. D. Lee, and H. Sompolinsky, "Equilibrium properties of temporally asymmetric Hebbian plasticity," *Physical Review Letters*, vol. 86, no. 2, pp. 364–367, 2001.

[11] A. Morrison, M. Diesmann, and W. Gerstner, "Phenomenological models of synaptic plasticity based on spike timing," *Biological Cybernetics*, vol. 98, no. 6, pp. 459–478, 2008.

[12] L. F. Abbott and S. B. Nelson, "Synaptic plasticity: taming the beast," *Nature Neuroscience*, vol. 3, pp. 1178–1183, 2000.

[13] E. M. Izhikevich and N. S. Desai, "Relating STDP to BCM," *Neural Computation*, vol. 15, no. 7, pp. 1511–1523, 2003.

[14] R. Gütig, R. Aharonov, S. Rotter, and H. Sompolinsky, "Learning input correlations through nonlinear temporally asymmetric Hebbian plasticity," *Journal of Neuroscience*, vol. 23, no. 9, pp. 3697–3714, 2003.

[15] H. Câteau and T. Fukai, "A stochastic method to predict the consequence of arbitrary forms of spike-timing-dependent plasticity," *Neural Computation*, vol. 15, no. 3, pp. 597–620, 2003.

[16] A. N. Burkitt, H. Meffin, and D. B. Grayden, "Spike-timing-dependent plasticity: the relationship to rate-based learning for models with weight dynamics determined by a stable fixed point," *Neural Computation*, vol. 16, no. 5, pp. 885–940, 2004.

[17] J.-P. Pfister and W. Gerstner, "Triplets of spikes in a model of spike timing-dependent plasticity," *Journal of Neuroscience*, vol. 26, no. 38, pp. 9673–9682, 2006.

[18] T. Masquelier, R. Guyonneau, and S. J. Thorpe, "Spike timing dependent plasticity finds the start of repeating patterns in continuous spike trains," *PLoS ONE*, vol. 3, no. 1, Article ID e1377, 2008.

[19] W. Gerstner and W. M. Kistler, *Spiking Neuron Models*, Cambridge University Press, Cambridge, UK, 2002.

[20] D. Krunglevicius, "Competitive STDP learning of overlapping spatial patterns," *Neural Computation*, vol. 27, no. 8, pp. 1673–1685, 2015.

[21] G.-Q. Bi and M.-M. Poo, "Synaptic modifications in cultured Hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type," *The Journal of Neuroscience*, vol. 18, no. 24, pp. 10464–10472, 1998.

[22] M. A. Woodin, K. Ganguly, and M.-M. Poo, "Coincident pre- and postsynaptic activity modifies GABAergic synapses by postsynaptic changes in Cl- transporter activity," *Neuron*, vol. 39, no. 5, pp. 807–820, 2003.

[23] P. J. Sjöström and M. Häusser, "A cooperative switch determines the sign of synaptic plasticity in distal dendrites of neocortical pyramidal neurons," *Neuron*, vol. 51, no. 2, pp. 227–238, 2006.

[24] J. J. Letzkus, B. M. Kampa, and G. J. Stuart, "Learning rules for spike timing-dependent plasticity depend on dendritic synapse location," *Journal of Neuroscience*, vol. 26, no. 41, pp. 10420–10429, 2006.

[25] M. Gilson, T. Masquelier, and E. Hugues, "STDP allows fast rate-modulated coding with Poisson-like spike trains," *PLoS Computational Biology*, vol. 7, no. 10, Article ID e1002231, 2011.

[26] B. Nessler, M. Pfeiffer, L. Buesing, and W. Maass, "Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity," *PLoS Computational Biology*, vol. 9, no. 4, article e1003037, 2013.

[27] R. Gütig, R. Aharonov, S. Rotter, and H. Sompolinsky, "Learning input correlations through nonlinear temporally asymmetric Hebbian plasticity," *The Journal of Neuroscience*, vol. 23, no. 9, pp. 3697–3714, 2003.

[28] P. J. Sjöström, G. G. Turrigiano, and S. B. Nelson, "Rate, timing, and cooperativity jointly determine cortical synaptic plasticity," *Neuron*, vol. 32, no. 6, pp. 1149–1164, 2001.

[29] H.-X. Wang, R. C. Gerkin, D. W. Nauen, and G.-Q. Bi, "Coactivation and timing-dependent integration of synaptic potentiation and depression," *Nature Neuroscience*, vol. 8, no. 2, pp. 187–193, 2005.

[30] T. Masquelier, R. Guyonneau, and S. J. Thorpe, "Competitive STDP-based spike pattern learning," *Neural Computation*, vol. 21, no. 5, pp. 1259–1276, 2009.

[31] B. Lindner, "Superposition of many independent spike trains is generally not a Poisson process," *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, vol. 73, no. 2, Article ID 022901, 2006.