What Can a Neuron Learn with Spike-Timing-Dependent Plasticity?

Robert Legenstein
legi@igi.tugraz.at
Christian Naeger
naeger@gmx.de
Wolfgang Maass
maass@igi.tugraz.at
Institute for Theoretical Computer Science, Technische Universitaet Graz,
A-8010 Graz, Austria

Spiking neurons are very flexible computational modules, which can implement with different values of their adjustable synaptic parameters an enormous variety of different transformations F from input spike trains to output spike trains. We examine in this letter the question to what extent a spiking neuron with biologically realistic models for dynamic synapses can be taught via spike-timing-dependent plasticity (STDP) to implement a given transformation F. We consider a supervised learning paradigm where during training, the output of the neuron is clamped to the target signal (teacher forcing). The well-known perceptron convergence theorem asserts the convergence of a simple supervised learning algorithm for drastically simplified neuron models (McCulloch-Pitts neurons). We show that in contrast to the perceptron convergence theorem, no theoretical guarantee can be given for the convergence of STDP with teacher forcing that holds for arbitrary input spike patterns. On the other hand, we prove that average case versions of the perceptron convergence theorem hold for STDP in the case of uncorrelated and correlated Poisson input spike trains and simple models for spiking neurons. For a wide class of cross-correlation functions of the input spike trains, the resulting necessary and sufficient condition can be formulated in terms of linear separability, analogously as the well-known condition of learnability by perceptrons. However, the linear separability criterion has to be applied here to the columns of the correlation matrix of the Poisson input. We demonstrate through extensive computer simulations that the theoretically predicted convergence of STDP with teacher forcing also holds for more realistic models for neurons, dynamic synapses, and more general input distributions. In addition, we show through computer simulations that these positive learning results hold not only for the common interpretation of STDP, where STDP changes the weights of synapses, but also

for a more realistic interpretation suggested by experimental data where STDP modulates the initial release probability of dynamic synapses.

1 Introduction

Spike-timing-dependent plasticity (STDP) has emerged in recent years as the experimentally most studied form of synaptic plasticity (see Abbott & Nelson, 2000; Frégnac, 2002; Gerstner & Kistler, 2002, for reviews). Numerous modeling studies have related STDP to various important learning rules and learning mechanisms such as Hebbian learning, short-term prediction (Mehta, 2001; Rao & Sejnowski, 2002), gain adaptation (Song, Miller, & Abbott, 2000), and boosting of temporally correlated inputs (Kempter, Gerstner, & van Hemmen, 1999; Song, et al., 2000; Gütig, Aharonov, Rotter, & Sompolinsky, 2003). The question of how a neuron can learn to fire at a prescribed time, given some presynaptic spike history, was investigated in the context of sequence learning, for example, in Gerstner, Ritz, and van Hemmen (1993) and Senn, Schneider, & Ruf, (2002). These two papers exploit tuning of synaptic delays to achieve timing precision. In this letter, we address the more general question to what extent STDP might support a more universal type of learning where a neuron learns to implement an "arbitrary given" map F from input spike trains to output spike trains. Obviously the goal to learn "arbitrary given" target transformations F is too ambitious, since there exist many maps F from input spike trains to output spike trains that cannot be realized by a neuron for any setting of its adjustable parameters. For example, no values of synaptic efficacies $\mathbf{w}$ could enable a generic neuron to produce a high-rate output spike train in the absence of any input spikes. Furthermore, a neuron can only learn to implement those transformations F in a stable manner that it can implement with a parameter setting that represents a equilibrium point for the learning rule under consideration (in this case, STDP). Since it is well known that the common version of STDP always produces bimodal distribution of weights, where each weight either assumes its minimal or its maximal possible value, we will consider in this article (with the exception of section 6) only the learning of target transformations F that can be implemented with such bimodal distribution of weights. Thus, we focus on those transformations F from input spike trains to output spike trains that can in principle be implemented by the neuron in a stable manner for some values $\mathbf{p}$ of its adjustable parameters, and ask which of these transformations F can be learned by such neuron, starting from some rather arbitrary values $\mathbf{p}_0$ of these adjustable parameters.

On the basis of the experimental literature, it is not at all clear which of the many parameters that influence the input-output behavior of a biologically realistic synapse should be viewed as being adjustable for a specific protocol for inducing synaptic plasticity (i.e., "learning"). For example, there exists strong empirical evidence (Markram & Tsodyks, 1996) that the common induction protocol with repeated pairings of pre- and postsynaptic spikes in a specific temporal relation does not change the scaling factors w of the amplitudes of excitatory postsynaptic potentials, (EPSPs, commonly referred to as "weight" or "synaptic efficacy"), but rather the synaptic release probability U for the first spike in a train of spikes. Whereas an increase of this parameter U will increase the amplitude of the EPSP for the first after a long inactive period spike in a spike train, just as an increase of the scaling factor w would do, it tends to decrease the amplitudes of shortly following EPSPs. We examine in this article through computer simulations both the case where scaling factors w and the case where initial release probabilities U are adjusted by STDP.

In contrast to most preceding modeling studies for STDP, we will consider in the computer simulations of this article only biologically realistic models for dynamic synapses—those that are subject to short-term plasticity such as paired-pulse depressions and paired-pulse facilitation, in addition to the long-term plasticity induced by STDP. We assume that during learning, the neuron is taught to fire at particular points in time via extra input currents, which could, for example, represent synaptic inputs from other cortical or subcortical areas. Such a learning scenario is particularly compelling in cases where a neuron learns to predict input from other cortical or subcortical areas. This learning protocol is identical to the experimental paradigm investigated by Yves Frégnac and his collaborators (Frégnac, Schulz, Thorpe, & Bienenstock, 1988, 1992; Frégnac & Shulz, 1999), where synaptic plasticity is induced through the injection of currents into the postsynaptic neuron at particular points in time (relative to the time of the stimulus). We will refer to the conjecture that STDP enables neurons to learn (starting with arbitrary initial values of its parameters $\mathbf{p}$) under this protocol any input-output transformation F that the neuron could in principle implement in a stable manner for some values $\mathbf{p}$ of its adjustable parameters as the spiking neuron convergence conjecture (SNCC) for STDP. Obviously this conjecture is closely related to the well-known perceptron convergence theorem (Rosenblatt, 1962; Haykin, 1999; Duda, Hart, & Storck, 2001), which asserts that the corresponding statement is true for the much simpler case of perceptrons (i.e., McCulloch-Pitts neurons or threshold gates with static synapses, static batch inputs, and static batch outputs—instead of time-varying input and output streams).

We will specify the models for neurons and synapses and the rule for STDP that are examined in this article in section 2. In section 3, we discuss the relationship between STDP and the perceptron learning rule. Furthermore we prove in section 3 that the SNCC for STDP does not hold in a worst-case scenario for arbitrary distributions of input spike trains.

In section 4, we carry out an analytical average case analysis of supervised learning with STDP for Poisson input spike trains (for the case of linear Poisson neurons and synapses without short-terms dynamics), and we prove that the SNNC holds in an average case sense for arbitrary uncorrelated Poisson input spike trains. We also derive in section 4 a criterion that clarifies under which conditions the SNNC holds for correlated Poisson input spike trains. In some situations, this criterion can be formulated in terms of linear separability, like the well-known learning criterion for perceptrons, but applied to the columns of the correlation matrix for the Poisson input.

In sections 5 and 6, we demonstrate through computer simulations that the SNCC for STDP also holds for more general ensembles of uncorrelated and correlated Poisson spike trains as inputs and for more realistic models for neurons and synapses: for leaky integrate-and-fire neurons with dynamic synapses. In section 7, we show that such approximate convergence of learning also occurs when instead of weights, the initial release probabilities U of the synapses are modulated by STDP.

2 Models for Neurons, Synapses, and STDP

A standard leaky integrate-and-fire neuron model was used for our simulations. The membrane potential V_m of such neuron is given by

$$\tau_m \frac{dV_m}{dt} = -(V_m - V_{resting}) + R_m \cdot \left(I_{syn}(t) + I_{background} + I_{inject}(t)\right),$$

where $\tau_m = C_m \cdot R_m$ is the membrane time constant, R_m is the membrane resistance, $I_{syn}(t)$ is the current supplied by the synapses, $I_{background}$ is a constant background current, and $I_{inject}(t)$ represents currents induced by a "teacher." If V_m exceeds the threshold voltage V_{thresh}, it is reset to V_{reset} and held there for the length $T_{refract}$ of the absolute refractory period (see appendix A for details).

We modeled the short-term-synaptic dynamics according to the model proposed in Markram, Wang, and Tsodyks (1998), with synaptic parameters U, D, F. The model predicts the amplitude A_k of the excitatory postsynaptic current (EPSC) for the kth spike in a spike train with interspike intervals $\Delta_1, \Delta_2, \ldots, \Delta_{k-1}$ through the equations

$$\begin{aligned} A_k &= w \cdot u_k \cdot R_k \\ u_k &= U + u_{k-1}(1-U)\exp(-\Delta_{k-1}/F) \\ R_k &= 1 + (R_{k-1} - u_{k-1}R_{k-1} - 1)\exp(-\Delta_{k-1}/D) \end{aligned} \tag{2.1}$$

with hidden dynamic variables $u \in [0, 1]$ and $R \in [0, 1]$ whose initial values for the first spike are $u_1 = U$ and $R_1 = 1$ (see Maass & Markram, 2002, for a justification of this version of the equation, which corrects a small error in Markram et al., 1998).

The parameters U, D, and F were randomly chosen from gaussian distributions that were based on empirically found data for such connections.

Depending on whether the input was excitatory (E) or inhibitory (I), the mean values of these three parameters (with D, F expressed in seconds) were chosen to be 0.5, 1.1, 0.05 (E) and 0.25, 0.7, 0.02 (I). The SD of each parameter was chosen to be 10% of its mean (with negative values replaced by values chosen from a uniform distribution between 0 and two times the mean).

The effect of STDP is commonly tested by measuring in the postsynaptic neuron the amplitude A_1 of the EPSP (or EPSC) for a single spike from the presynaptic neuron (after a longer resting period subsequent to the protocol for induction of STDP). Since $A_1 = w \cdot U \cdot R_1$, one can interpret any change ΔA in the amplitude of A_1 (in comparison with the value of A_1 before applying the protocol for STDP) as being caused by a proportional change Δw of the parameter w (with U unchanged), by a proportional change ΔU of the initial release probability $u_1 = U$ (with w unchanged), or by a change of both w and U (and possible even further synaptic parameters). The first case is the one that is most commonly assumed in modeling studies (see, e.g., Abbott & Nelson, 2000; Frégnac, 2002; Gerstner & Kistler, 2002), and is analyzed in sections 5 and 6 of this letter. The second case is strongly favored by the experimental data of Markram & Tsodyks (1996), and it is apparently not contradicted by any of the other experimental data (since one usually measures the efficacy of the synapse after induction of plasticity with just a single test spike). This case is examined in section 7 of this letter. The third case is not considered because of a lack of quantitative experimental data.

According to Abbott & Nelson (2000), the change ΔA_1 in the amplitude A_1 of EPSPs (for the first spike in a test spike train) that results from (usually repeated) pairing of the firing of the presynaptic neuron at some time t^{pre} and a firing of the postsynaptic neuron at time $t^{\mathrm{post}} = t^{\mathrm{pre}} + \Delta t$ can be approximated for many cortical synapses by terms of the form

$$A(\Delta t) = \begin{cases} W_+ \cdot e^{-\Delta t/\tau_+}, & \text{if } \Delta t > 0 \\ -W_- \cdot e^{\Delta t/\tau_-}, & \text{if } \Delta t \leq 0 \end{cases} \tag{2.2}$$

with constants $W_+, W_-, \tau_+, \tau_- > 0$ (and with an extra clause that prevents the amplitude A_1 from growing beyond some maximal value $A_{\max}$ or below 0).

For the theoretical analysis in section 4, spike trains $S(t)$ are represented by sums of Dirac-δ functions $S(t) = \sum_k \delta(t - t_k)$, where t_k is the kth spike time of the spike train. The leaky integrate-and-fire neuron is replaced here by a linear Poisson neuron model as in Kempter, Gerstner, & van Hemmen (2001) and Gütig et al. (2003). This neuron model outputs a spike train $S^{post}(t)$, which is a realization of a Poisson process with the underlying instantaneous firing rate $R^{post}(t)$. The effect of an input spike at input i at time t' is modeled by an increase in the instantaneous firing rate of an amount $w_i(t')\epsilon(t-t')$, where ϵ is a response kernel and $w_i(t')$ is the synaptic efficacy of synapse i at time t'. Thus, the response kernel ϵ models the time course of a postsynaptic potential elicted by an input spike. Since the neuron model is causal, we have the requirement $\epsilon(s) = 0$ for $s < 0$. We will consider plasticity only for excitatory connections so that $w_i \geq 0$ for all i and $\epsilon(s) \geq 0$ for all s. In addition, the response kernel is normalized to $\int_0^\infty ds\ \epsilon(s) = 1$. In the linear model, the contributions of all inputs are summed up linearly:

$$R^{post}(t) = \sum_{j=1}^{n} \int_0^\infty ds\ w_j(t-s)\,\epsilon(s)\, S_j(t-s)\,, \tag{2.3}$$

where $S_1, \ldots, S_n$ are the n presynaptic spike trains. Note that in this spike generation process, the generation of an output spike is independent of previous output spikes.

3 The Perceptron Convergence Theorem and a Counterexample to the Spiking Neuron Convergence Conjecture for STDP

If one assumes that STDP affects only the parameter w, then the change Δw of the weight (or efficacy) of the synapse is according to equation 2.2 proportional to:

$$\begin{cases} W_+ \cdot e^{-\Delta t/\tau_+}\,, & \text{if } \Delta t > 0 \\ -W_- \cdot e^{\Delta t/\tau_-}\,, & \text{if } \Delta t \leq 0, \end{cases} \tag{3.1}$$

with an extra clause that prevents w from becoming larger than some maximal value $w_{\max}$ or smaller than 0. Hence, STDP changes the value w_{old} of the synaptic weight to $w_{new} = w_{old} + \Delta w$ according to the rule

$$w_{new} = \begin{cases} \min\{w_{\max},\ \ w_{old} + W_+ \cdot e^{-\Delta t/\tau_+}\}, & \text{if } \Delta t > 0 \\ \max\{0,\ \ w_{old} - W_- \cdot e^{\Delta t/\tau_-}\}, & \text{if } \Delta t \leq 0\,, \end{cases} \tag{3.2}$$

with some parameters $W_+, W_- > 0$.

There exists some analogy between this STDP rule and common learning rules such as the Hebb rule, the perceptron learning rule, and the least-mean-square learning rule for strongly simplified neuron models that are used in the context of artificial neural networks. These simplified neuron models do not "fire." Instead, their inputs and outputs consist of real numbers, which may change their value at each discrete time step. If $\mathbf{x} = \langle x_0, \ldots, x_n\rangle \in \mathbb{R}^{n+1}$ denotes the input vector to an artificial neuron and $y \in \mathbb{R}$ the resulting output, then the basic Hebbian learning rule for changing the weights $\mathbf{w} = \langle w_0, \ldots, w_n\rangle \in \mathbb{R}^{n+1}$ of a linear neuron with $y = \mathbf{w} \cdot \mathbf{x}$ is

$$\Delta\mathbf{w} = \eta \cdot \mathbf{x} \cdot y, \tag{3.3}$$

where $\eta \geq 0$ is the learning rate.

For supervised learning in artificial neural networks, there exists in addition a target value $y_{teacher}$ for the output of a neuron, and by replacing y in the Hebb rule, equation 3.3, by $y_{teacher} - y$, one gets the rule

$$\Delta \mathbf{w} = \eta \cdot \mathbf{x} \cdot (y_{teacher} - y), \tag{3.4}$$

which is for a linear neuron model $y = \mathbf{w} \cdot \mathbf{x}$ the least-mean-square or Widrow-Hoff rule (see section 2.2 in Rosenblatt, 1962; Haykin, 1999). This learning rule implements gradient descent in weight space for the mean of squared errors $(y_{teacher} - y)^2$ if applied to a list of training examples.

In the context of McCulloch-Pitts neurons, also called perceptrons or threshold gates, whose output y assumes only values 0 or 1, this rule, equation 3.4, is the well-known perceptron learning rule (see Haykin, 1999, and Duda et al., 2001).[1] For this case one can write rule 3.4 equivalently in the form

$$\Delta \mathbf{w} = \begin{cases} \eta \cdot \mathbf{x}, & \text{if } y_{teacher} = 1 \quad \text{and } y = 0 \\ \eta \cdot (-\mathbf{x}), & \text{if } y_{teacher} = 0 \quad \text{and } y = 1 \\ 0, & \text{otherwise .} \end{cases} \tag{3.5}$$

The first line of this perceptron learning rule implements learning from a positive example $\mathbf{x}$ (which is in this case a positive counterexample hypothesis defined by the current weight vector $\mathbf{w}$, since $y_{teacher} = 1$ but $y = sign(\mathbf{w} \cdot \mathbf{x}) = 0$). The second line implements learning from a negative example $\mathbf{x}$ (i.e., from a negative counterexample to the current hypothesis defined by $\mathbf{w}$). The seemingly trivial third line of the rule makes sure that $\mathbf{w}$ is not changed for the current example $\mathbf{x}$ if it is correctly classified with the current weight vector $\mathbf{w}$ (i.e., $y_{teacher} = sign(\mathbf{w} \cdot \mathbf{x})$).

The main result about this perceptron learning rule is the perceptron convergence theorem (see Rosenblatt, 1962; Haykin, 1999; Duda et al., 2001). It states that learning with the perceptron learning rule converges for a given list L of examples if and only if the list L is linearly separable. If L is linearly separable, then the weight vector to which this learning rule converges is autonomically a solution of the corresponding classification

[1] If one defines $\operatorname{sign} z = 1$ if $z \geq 0$, else $\operatorname{sign} z = 0$ (as we do throughout this letter), then the output y of a perceptron can be defined in compact form as $y = \operatorname{sign}(\mathbf{w} \cdot \mathbf{x})$. One commonly uses the convention that in the context of perceptrons, the first component x_0 of any input vector $\mathbf{x}$ has a fixed value $x_0 = 1$. This implies that

$$\operatorname{sign}(\mathbf{w} \cdot \mathbf{x}) = \begin{cases} 1, & \text{if } \sum_{i=1}^{n} w_i x_i \geq -w_0 \\ 0, & \text{otherwise,} \end{cases}$$

and hence the weight w_0 for this dummy component x_0 of the input, multiplied with -1, assumes the effective role of an adaptive threshold for the perceptron.

problem. Obviously linear separability of L is a necessary condition for the convergence of perceptron learning. But this simple condition from linear algebra is also sufficient: if there exists a weight vector $\mathbf{w}^*$ that can classify the list $L = \langle\langle \mathbf{x}(1), y_1\rangle, \ldots, \langle \mathbf{x}(m), y_m\rangle\rangle$ of examples from $\mathbb{R}^{n+1} \times \{0, 1\}$ without error, (i.e., $\mathit{sign}(\mathbf{w}^* \cdot \mathbf{x}(k)) = y_k$ for $k = 1, \ldots, m$), then the perceptron learning rule, equation 3.5, will converge to some weight vector $\mathbf{w}'$ with the same property (i.e., $\mathit{sign}(\mathbf{w}' \cdot \mathbf{x}(k)) = y_k$ for $k = 1, \ldots, m$) after cycling some finite number of times through the list L of training examples (starting from *any* initial weights $\mathbf{w} \in \mathbb{R}^{n+1}$). The perceptron convergence theorem can be interpreted as a very positive result on learnability, since it implies that the perceptron learning rule enables a perceptron to learn any map from inputs $\mathbf{x}$ to outputs y that it could possibly implement in a stable manner.

Note that any weight vector that allows a perceptron to become consistent with a list L of training examples yields an equilibrium point for the perceptron learning rule, since in contrast to STDP, this learning rule automatically becomes inactive when errors no longer occur for the training examples. Hence, any setting of $\mathbf{w}$ that allows a perceptron to solve a given classification task is automatically stable with regard to the perceptron learning rule. Such automatic stability is not provided by STDP. Therefore, in order to make the spiking neuron convergence conjecture more meaningful (by giving it a larger chance to be true), we consider in this section only learning tasks for spiking neurons for which a solution exists that is stable with regard to STDP. In other words, we want to clarify whether in a supervised paradigm where the output is clamped to the teacher signal, STDP enables a spiking neuron, starting from any initial weights, to learn any transformation F from input spike trains to output spike trains that it can possibly implement in a stable manner (this is the spiking neuron convergence conjecture, or SNCC; see section 1).

One salient difference between the perceptron learning rule and STDP is caused by the different structure of inputs and outputs of perceptrons and spiking neurons: inputs and outputs to a perceptron are (static) vectors of numbers, whereas they are functions of time (spike trains) in the case of a spiking neuron. Thus, mathematically, the transformation F from inputs to outputs computed by a spiking neuron with n input channels is a filter that maps n functions S_i that represent n input spike trains $S_1, \ldots, S_n$ onto some output spike train S of the same form.[2] Apart from this basic difference regarding the types of inputs and outputs, the perceptron learning rule and STDP also differ in the following structural aspects:

i. The sign of any weight w_i of a perceptron can be changed by the perceptron learning rule, whereas one usually does not assume that

[2] Obviously a spiking neuron can implement only causal filters F, where for any time t, the value of $S(t)$ depends on only the initial segments of $S_1, \ldots, S_n$ up to time t.

STDP can turn an excitatory synapse into an inhibitory synapse, or vice versa.

ii. In the case where an example **x** that should be classified negatively is incorrectly classified through the current weight vector **w** (i.e., $y = \text{sign}(\mathbf{w} \cdot \mathbf{x}) = 1$ but $y_{teacher} = 0$), the perceptron rule changes **w** in a way that makes a reoccurrence of this mistake less likely. Something quite different happens in the analogous scenario for STDP if the neuron fires in response to an input for which it is not supposed to fire. In our training paradigm, where hyperpolarizing teacher currents suppress all undesired firing during training, no changes of synaptic parameters are triggered by such mistakes during training. Hence, this mistake is likely to show up again during testing (where there are no teacher currents anymore). For the alternative training paradigm where the teacher does not suppress this undesired firing during training, rules 2.1 to 2.3, 3.1, and 3.2 for STDP change the synaptic parameters in a way that positively reinforces future reoccurrences of this mistake.[3]

iii. The perceptron learning rule leaves the weights of the perceptron unchanged when it does not make a mistake (i.e., $y_{teacher} = y$; see the third line of equation 3.5), whereas STDP will continue to change synaptic parameters even if the neuron fires exactly at the desired times t (even if this firing occurs without the help of an extra "teaching current").

It had been shown in Amit, Wong, and Campbell (1989) that the first apparent difference (i) between perceptron learning and STDP is not crucial for the convergence of learning, since the perceptron convergence theorem also holds for a sign-constrained version of the perceptron learning rule.[4] We will show in the remainder of this section that the structural difference ii (even without difference, iii) is quite serious, and entails a falsification of the SNCC for STDP in some worst-case learning scenarios. To elucidate this fact, we first demonstrate in Figure 1 that the perceptron convergence theorem would no longer hold for certain learning scenarios if the second line of the perceptron learning rule, equation 3.5 (which specifies its response to negative counterexamples) is deleted, even if one starts with initial weights of value 0. The reason is that in this case, the resulting decision boundary

[3] This may be just a deficit of current formalizations of STDP, not of the biological reality of synaptic plasticity. Debanne et al. (1998) and Frégnac (2002) have provided evidence for synaptic plasticity resulting from teacher-induced suppression of firing (Figure 2D in Markram, Lübke, Frotscher, and Sakmann (1997) also shows this effect).

[4] Senn & Fusi (in press) have recently shown that the perceptron convergence theorem also remains valid for a learning rule that in addition keeps the sizes of positive weights bounded.

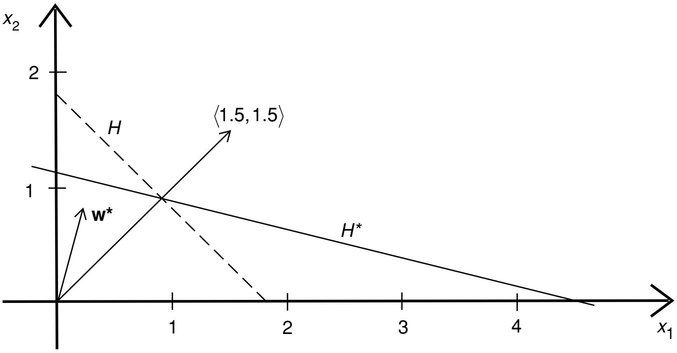

Figure 1: Demonstration that the perceptron convergence theorem fails if the second line of the perceptron learning rule, equation 3.5, is deleted, even if one starts with small initial weights. Assume that the hyperplane H^* generated by weight vector $\mathbf{w}^*$ is the target decision boundary (positive examples above H^*, negative examples below H^*), and that the list L of examples that occurs in the perceptron convergence theorem consists of just two examples: the positive example $\langle 1.5, 1.5\rangle$ and the negative example $\langle 4, 0\rangle$.* If one starts, for example, with the initial weight vector $\mathbf{w} = \langle 0, 0\rangle$, a decision boundary parallel to H will arise, no matter how long the training is continued, if the second line of the perceptron learning rule is deleted. Any such decision boundary will missclassify one of the two examples in the list L.

* Formally the perceptron learning rule is applied in this example to a list L consisting of the positive example $\langle 1, 1.5, 1.5\rangle$ and the negative example $\langle 1, 4, 0\rangle$, that is, $L = \langle\langle 1, 1.5, 1.5, 1\rangle, \langle 1, 4, 0, 0\rangle\rangle$. Thus, the points $\langle 1.5, 1.5\rangle$, $\langle 4, 0\rangle$ have to be expanded by an additional dummy coordinate with value 1, whose associated weight represents the (adjustable) constant term in the resulting hyperplane H (see note 1). But this formal detail does not affect the validity of the argument.

depends on accidental details of the positive examples in the training set L, and negative examples cannot have any impact on learning.

One can transfer the main idea of the counterexample illustrated in Figure 1 into the domain of spike trains and prove in this way that the SNCC for STDP is false, at least for certain learning scenarios (see Figure 2). If the set of possible spike inputs consists of only the two patterns shown in Figure 2, then STDP does not converge from all initial weight settings to a stable solution, although a stable solution exists. Details of the verification are in appendix B.

Since we consider in Figure 2 only inputs where each presynaptic neuron fires at most once before the target firing time t_3 of the postsynaptic neuron, the same example also proves that the SNCC for STDP fails if one assumes that STDP changes the initial release probabilities U instead of the scaling factors w (see the synapse model discussed in section 2, with a rule for ΔU

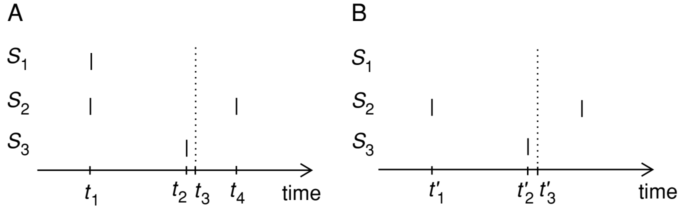

Figure 2: Construction of a counterexample to an analogous version (SNCC) of the perceptron convergence theorem for STDP. S_1, S_2, S_3 denote three input spike trains to three synapses of a neuron. (A) A positive example where firing of the postsynaptic neuron at time t_3 is desired. (B) A negative example where no firing of the postsynaptic neuron is desired. See appendix B for details.

that is analogous to the previously discussed rule, equation 3.1, for Δw; this rule will be discussed as rule 7.1). Thus we have proven that *the spiking neuron convergence conjecture for STDP is not generally valid for either the case where synaptic efficacies w are modulated by STDP or for the case where initial release probabilities U are modulated by STDP.*

4 Theoretical Results on STDP in the Context of Supervised Learning: Average Case Analysis

We showed in the preceding section that it is not possible to derive for STDP a convergence result that has the same mathematical structure as the perceptron convergence theorem (yielding a guarantee of convergence for any set of inputs just under the assumption that a suitable weight vector exists). Therefore, we now turn to an average case analysis of STDP for Poisson input spike trains.

The reason that the validity of the SNCC for STDP depends on the distribution of inputs can already be read off from the analogous scenario for the perceptron learning rule without line 2 of equation 3.5 that is shown in Figure 1. If the list L of training examples contained not just a single positive example (i.e., one example of a point that lies above the target decision boundary H^*) but rather a larger set of positive examples covering the area above H^*, then L would contain more positive examples $\langle x_1, x_2 \rangle$ with $x_2 > x_1$ than positive examples with $x_2 < x_1$. This asymmetry in the coordinates of positive examples is likely to cause a weight vector $\mathbf{w}$ with $w_2 > w_1$, since the perceptron learning rule in equation 3.8 without line 2 creates a weight vector $\mathbf{w}$ that is proportional to the sum of positive counterexamples that occur during learning. Hence, the angle between the resulting vector $\mathbf{w}$ and the target vector $\mathbf{w}^*$ is likely to get smaller for such more uniform distribution of inputs (compared with the worst-case scenario discussed in Figure 1). Analogously, if one generates positive training examples for a spiking neuron by injecting Poisson input spike trains, rather than constructing particular examples of spatiotemporal input patterns as in Figure 2, one creates a more uniform distribution of spatiotemporal input patterns for which the neuron is supposed to fire. In this way, the learning process via STDP also implicitly gets information about the distribution of negative examples, that is, spatiotemporal input patterns for which the neuron is not supposed to fire, and hence they can indirectly influence the learning process even without any explicit provision in the rule for STDP that discourages the firing of the neuron for such input patterns.

It turns out that for the average case, a form of the SNCC does, in fact hold (see theorem 1) if the output of the neuron is clamped to the teacher signal; hence, neither false positives nor false negatives arise during training. Furthermore, we show in theorem 2 that a general criterion for learnability can be given that has the form of a condition on the correlation matrix of Poisson inputs. Curiously enough, this condition has the form of a linear separability condition, just like the condition on learnability for perceptrons, although it arises here in a quite different context. In general, it turns out that all these provable convergence results for STDP require a suitable choice of the relationship between the parameters W_+ and W_- that scale the relative impact of synaptic facilitation and depression in STDP.

As a preparation for the subsequent average case analysis, we need to express weight changes resulting from STDP by suitable integrals. STDP exploits correlations between input and output spike trains on the timescale of the positive learning window. Inputs that are strongly correlated with the output spike train are reinforced. If the integral over the whole learning window is negative, inputs with correlations on chance level or slightly above are weakened. More formally, let S_i be the spike train of input i and let S^* be the output spike train. Both are represented in this section as sums of δ-functions (see the definitions at the end of section 2). We consider the total weight change $\Delta w_i(t) = w_i(t+T) - w_i(t)$ resulting from pre- and postsynaptic spikes within a given time interval of length T. Ignoring the effect of weight clipping, the total weight change is the integral over all individual weight changes resulting from learning rule 3.1:

$$\Delta w_i(t) = \int_t^{t+T} dt' \int_t^{t+T} dt'' \; A(t'-t'')S^*(t'')S_i(t'). \tag{4.1}$$

By substituting $s = t'' - t'$ we get

$$\Delta w_i(t) = \int_t^{t+T} dt' \int_{t-t'}^{t+T-t'} ds \; A(s)S^*(t'+s)S_i(t'). \tag{4.2}$$

But weight changes during the time interval $[t, t+T]$ can potentially also be caused by pre- or postsynaptic spikes that do not fall into this interval (especially if T is small). This can be taken into account by extending the integration range of the second interval to $(-\infty, \infty)$, so that one arrives at

$$\Delta w_i(t) = \int_t^{t+T} dt' \int_{-\infty}^{\infty} ds \; A(s) S^*(t'+s) S_i(t'). \tag{4.3}$$

This formula assumes for simplicity that all weight updates resulting from pre- and postsynaptic firing are always credited to the time point t' of the presynaptic firing. The error caused by this approximation is small if the learning rates defined by W_+, W_- (i.e., the size of each single weight update) are sufficiently small and the firing rates are sufficiently small so that the value of a weight cannot change too much over the length of a single learning window of STDP (i.e., during a single time interval of a length s for which $A(s)$ is still relatively large).

4.1 A Necessary Condition on Input Spike Trains. If we assume that the statistics of input and output spike trains are constant over a learning trial, the total weight change over a sufficiently long time interval T provides a good predictor for the end result of a learning process. Consider a neuron with n synapses and a set $M \subseteq \{1, \ldots, n\}$. Suppose that the neuron computes the target transformation F^* if and only if $w_i = w_i^{\max}$ for all $i \in M$ and $w_i = 0$ for all $i \notin M$. Then for learning F^*, the learning window should be such that all weights w_i with $i \in M$ have positive total weight change. On the other hand, all weights w_i with $i \notin M$ will need to have negative total weight change (if it is allowed that the initial weights can be nonzero). If one assumes a simple learning window with exponentially decaying positive and negative parts as given in rule 3.1, one can determine the possible range of W_-/W_+ by this criterion. For every $i \in M$, the total change of w_i has to be positive:

$$\begin{aligned} &\int_t^{t+T} dt' \int_0^{\infty} ds \; W_+ \; S^*(t'+s) S_i(t') \, e^{-s/\tau_+} \\ &- \int_t^{t+T} dt' \int_{-\infty}^{0} ds \; W_- \; S^*(t'+s) S_i(t') \, e^{s/\tau_-} > 0. \end{aligned} \tag{4.4}$$

Therefore, W_-/W_+ must satisfy

$$\frac{W_-}{W_+} < \frac{\int_t^{t+T} dt' \int_0^{\infty} ds \; S^*(t'+s) S_i(t') \, e^{-s/\tau_+}}{\int_t^{t+T} dt' \int_{-\infty}^{0} ds \; S^*(t'+s) S_i(t') \, e^{s/\tau_-}} \tag{4.5}$$

for all $i \in M$. Furthermore, for every $i \notin M$, the total change of w_i has to be negative, that is, W_- / W_+ must satisfy

$$\frac{W_-}{W_+} > \frac{\int_t^{t+T} dt' \int_0^\infty ds \; S^*(t'+s) S_i(t') \, e^{-s/\tau_+}}{\int_t^{t+T} dt' \int_{-\infty}^0 ds \; S^*(t'+s) S_i(t') \, e^{s/\tau_-}} \tag{4.6}$$

for all $i \notin M$. A value in the middle between these maximum and minimum values for W_- / W_+ seems desirable to minimize the effects of noise in the learning process.

4.2 Correlated and Uncorrelated Poisson Input. In general, the spike trains $S_1, \ldots, S_n, S^*$ may not be known, only the process that generated them. For example, one may only know the statistics of the inputs (e.g., correlated Poisson spike trains), but not the actual realizations. Furthermore, if we assume that the target spike train S^* is generated by some neuron with a certain target weight vector, the spike generation process might be stochastic and S^* is therefore not known explicitly. In these cases, the change Δw_i is a random variable with a mean drift and fluctuations around it. We will focus on the drift by assuming that individual weight changes are very small and only averaged quantities enter the learning dynamics (see Kempter et al., 1999).

The STDP rule, 3.2, avoids the growth of weights beyond bounds 0 and $w_{\max}$ by simple clipping. This leads to weights that tend to assume either of the clipping values 0 or $w_{\max}$. Alternatively, one can make the weight update dependent on the actual weight value,

$$\Delta w = \begin{cases} W_+ \cdot f_+(w) \cdot e^{-\Delta t/\tau_+}, & \text{if } \Delta t > 0 \\ -W_- \cdot f_-(w) \cdot e^{\Delta t/\tau_-}, & \text{if } \Delta t \leq 0, \end{cases} \tag{4.7}$$

with suitable functions $f_+(w)$ and $f_-(w)$ (see Kistler & van Hemmen, 2000; van Rossum, Bi, & Turrigiano, 2000; Rubin, Lee, & Sompolinsky, 2001). In Gütig et al. (2003), a general rule is suggested where the weight dependence has the form of a power law with a nonnegative exponent μ: $f_+(w) = ((w_{\max} - w)/w_{\max})^{\mu_+}$ and $f_-(w) = (w/w_{\max})^{\mu_-}$. For $\mu_+ = \mu_- = 0$ this rule recovers the basic additive update. The case $\mu_+ = \mu_- = 1$ corresponds to a multiplicative model where the update is linearly dependent on the current weight value.

In the remainder of this section, we assume for simplicity that $w_{\max} = 1$ and $\mu_+ = \mu_- = \mu$. Then the weight-dependent update factors simplify to $f_+^\mu(w) := (1-w)^\mu$ and $f_-^\mu(w) := w^\mu$. Thus, rule 4.7 becomes

$$\Delta w = \begin{cases} W_+ \cdot (1-w)^\mu \cdot e^{-\Delta t/\tau_+}, & \text{if } \Delta t > 0 \\ -W_- \cdot w^\mu \cdot e^{\Delta t/\tau_-}, & \text{if } \Delta t \leq 0. \end{cases} \tag{4.8}$$

With this synaptic update rule, the total weight change can be approximated by

$$\Delta w_i(t) = \int_t^{t+T} dt' \left[\int_0^{\infty} ds\ W_+ f_+^{\mu}(w_i(t)) e^{-s/\tau} S^*(t'+s) S_i(t') \right. \\ \left. - \int_{-\infty}^{0} ds\ W_- f_-^{\mu}(w_i(t)) e^{s/\tau} S^*(t'+s) S_i(t') \right], \tag{4.9}$$

where we have set $\tau_+ = \tau_- = \tau$ for convenience and replaced $f_+(w_i(t'))$ by $f_+(w_i(t))$, as well as $f_-(w_i(t'))$ by $f_-(w_i(t))$ (assuming that learning proceeds on a timescale larger than T—i.e., that $w_i(t)$ does not changes much during a time interval of length T).

Consider the ensemble of all possible realizations of input and output spike trains given by some fixed spike generation processes for input and output spike trains. The average over this ensemble is in the following denoted by $\langle . \rangle_E$ and called *ensemble average*. Taking the ensemble average over the weight change in equation 4.9 and dividing by T yields

$$\frac{\langle \Delta w_i \rangle_E(t)}{T} = \frac{1}{T} \int_t^{t+T} dt' \left[f_+^{\mu}(w_i(t)) \int_0^{\infty} ds\ W_+ e^{-s/\tau} \langle S^*(t'+s) S_i(t') \rangle_E \right. \\ \left. - f_-^{\mu}(w_i(t)) \int_{-\infty}^{0} ds\ W_- e^{s/\tau} \langle S^*(t'+s) S_i(t') \rangle_E \right], \tag{4.10}$$

where the function $\langle S_i(t') S^*(t'+s) \rangle_E$, which measures the correlation between S_i and S^*, is defined as the joint probability density for observing an input spike at synapse i at time t' and an output spike at time $t'+s$. A real neuron does not integrate over the whole ensemble; instead, learning is driven by a single realization of the stochastic process. But instead of averaging over several trials, we may also consider one single long trial during which input and output characteristics remain constant. In the following analysis, input and output spike trains will always be assumed to result from Poisson processes. Because disjoint time intervals are independent in a Poisson process, the integral in equation 4.9 decomposes into many independent events. Thus, for sufficiently small individual weight updates, learning is self-averaging (see also Kempter et al., 1999). This means that instead of learning on different examples from the ensemble, one can also learn from a long single example to achieve the mean drift in equation 4.10.

We can exchange the integrals in equation 4.10 and introduce a temporally averaged correlation function $C_i(s; t) := \frac{1}{T} \int_t^{t+T} dt' \langle S_i(t') S^*(t'+s) \rangle_E$. Since in the following we will assume that spike trains are homogeneous Poisson spike trains, the temporal average can be skipped, and we get

$$C_i(s; t) = \langle S_i(t) S^*(t+s) \rangle_E \tag{4.11}$$

for the temporal averaged correlation function. We approximate the left-hand side of equation 4.10 by $dw_i(t)/dt \equiv \dot{w}_i(t)$, and thereby obtain

$$\begin{aligned}\dot{w}_i(t) = {} & W_+ f_+^{\mu}(w_i(t)) \int_0^{\infty} ds\, e^{-s/\tau} C_i(s;t) \\ & - W_- f_-^{\mu}(w_i(t)) \int_{-\infty}^{0} ds\, e^{s/\tau} C_i(s;t). \end{aligned} \tag{4.12}$$

We call $\dot{w}_i(t)$ the synaptic drift of synapse i.

We now return to the previously discussed learning task. Consider an arbitrary set $M \subseteq \{1, \dots, n\}$ and assume that the target weight vector $\mathbf{w}^*$ satisfies $w_i^* = 1$ if $i \in M$ and $w_i^* = 0$ otherwise. The target output spike train S^* is produced by a neuron with synaptic efficacies $\mathbf{w}^*$ and input spike trains $S_1, \dots, S_n$. The question is whether a neuron with some rather arbitrary initial weight vector can learn the target transformation F^*, which maps inputs $S_1, \dots, S_n$ to the target output S^*, defined by $S_1, \dots, S_n, \mathbf{w}^*$. We assume that the neuron receives $S_1, \dots, S_n$ as inputs and is forced to spike only at times given by S^* during training. Note that for homogeneous Poisson spike trains as inputs and a stationary generation process of the target output S^*, $C_i(s;t)$ is constant over time. We will skip the dependence on t in the notation to emphasize this.

A precise mathematical characterization of those target transformations F^* (defined by some weight vector $\mathbf{w}^*$), which can be learned by STDP, turns out to be a bit complicated. One complication arises from the fact that a direct analysis of convergence for the STDP rules 3.1 and 3.2 is very difficult because the resulting fluctuations around the barriers 0 and $w_{\max}$ are hard to analyze. It turns out that rule 4.18, that is, rule 4.7 with $f_+(w) = f_+^{\mu}(w) = (1-w)^{\mu}$ and $f_-(w) = f_-^{\mu}(w) = w^{\mu}$, is easier to analyze. But this rule no longer yields convergence to the target vector $\mathbf{w}^*$ (in the case of supervised training with teacher-enforced output spike train S^*), but yields instead convergence to some other weight vector that is now dependent on μ. For example, equation 4.20 in the proof of theorem 2 will show that STDP with multiplicative updates according to rule 4.8 converges to a weight vector in $(0,1)^n$ even if $\mathbf{w}^* \in \{0,1\}^n$. We express this weight vector through a function $\mathcal{W} : \mathbb{R}^+ \to (0,1)^n$, which maps each $\mu > 0$ onto a weight vector $\mathcal{W}(\mu)$ (we set $\mathbb{R}^+ := \{x \in \mathbb{R} : x > 0\}$ in this article). For $\mu \to 0$, this rule, 4.8, approximates the original STDP rule, 3.1, and, accordingly, the function $\mathcal{W}(\mu)$ converges to the target vector $\mathbf{w}^*$. Thus, we have to replace a direct analysis of supervised learning with rule 3.1 by the analysis of the limit of supervised learning with rule 4.8 for $\mu \to 0$. This motivates the following definition of learnability:

Definition 1. *We say that a target weight vector* $\mathbf{w}^* \in \{0,1\}^n$ *can approximately be learned in a supervised paradigm where the output is clamped to the teaching* *signal by STDP with soft weight bounds on homogeneous Poisson input spike trains (short: "$\mathbf{w}^*$ can be learned") if and only if there exists a function $\mathcal{W} : \mathbb{R}^+ \to (0, 1)^n$ with $\lim_{\nu \to 0} \mathcal{W}(\nu) = \mathbf{w}^*$ and there exist W_+, $W_- > 0$, such that for all $\mu > 0$, the ensemble averaged weight vector $\langle \mathbf{w}(t) \rangle_E$ with learning dynamics given by equation 4.12 converges to $\mathcal{W}(\mu)$ for any initial weight vector $\mathbf{w}(0) \in [0, 1]^n$.*

The following theorem asserts that stability of target weight vectors under STDP already implies that they can be learned. This implies that each locally stable equilibrium point of the weight dynamics is a global attractor for the dynamical system defined by the learning equations.[5]

Theorem 1. *A target weight vector $\mathbf{w}^* \in \{0, 1\}^n$ can be learned if and only if there exists a function $\mathcal{W} : \mathbb{R}^+ \to (0, 1)^n$ with $\lim_{\nu \to 0} \mathcal{W}(\nu) = \mathbf{w}^*$ and there exist W_+, $W_- > 0$, such that for all $\mu > 0$, $\mathcal{W}(\mu)$ is a stable equilibrium point of the ensemble averaged weight vector $\langle \mathbf{w}(t) \rangle_E$ with learning dynamics given by equation 4.12.*

Proof. Due to teacher forcing, the integrals over the positive and negative learning window in equation 4.12 do not depend on $\mathbf{w}(t)$ and are therefore constant. We use the abbreviation C_i^{pos} for $\int_0^\infty ds\ e^{-s/\tau} C_i(s)$ and C_i^{neg} for $\int_{-\infty}^0 ds\ e^{s/\tau} C_i(s)$. The learning dynamics can therefore be separated into n independent one-dimensional dynamical systems.

To show the "if" part of theorem 1, we show that for any $\mu > 0$, the stable equilibrium point $\mathcal{W}(\mu) = \langle w_{\mu 1}, \ldots, w_{\mu n} \rangle$ is the only equilibrium point of the system. Consider an arbitrary $\mu > 0$ and an arbitrary synapse i. Since $w_{\mu i}$ is a stable equilibrium point, the synaptic drift for small perturbations from $w_{\mu i}$ is such that w_i converges to $w_{\mu i}$. We show that the synaptic drift has this property for all initial values $w_i(0) \in [0, 1]$ (since the system is time invariant, it suffices to consider perturbations at $t = 0$). For all $w_i(0) < w_{\mu i}$ with $w_i(0)$ sufficiently close to $w_{\mu i}$, we know that the synaptic drift is positive, because the equilibrium point is stable. From equation 4.12, we get $0 < \dot{w}_i(0) = W_+ C_i^{pos} (1 - w_i(0))^\mu - W_- C_i^{neg} w_i(0)^\mu$. By definition, we have $C_i^{pos}, C_i^{neg} \geq 0$, and $C_i^{pos} = C_i^{neg} = 0$ is impossible since this would imply $\dot{w}_i(0) = 0$ for all values of $w_i(0)$. Therefore, it holds for all $w_i'(0)$ with $0 \leq w_i'(0) < w_i(0)$ that $W_+ C_i^{pos} (1 - w_i(0))^\mu - W_- C_i^{neg} w_i(0)^\mu < W_+ C_i^{pos} (1 - w_i'(0))^\mu - W_- C_i^{neg} w_i'(0)^\mu$. Hence, the synaptic drift is positive for all weight values smaller than $w_{\mu i}$. A similar argument shows that the synaptic drift is negative for all weight values$w_i(0)$ with $w_{\mu i} < w_i(0) \leq 1$.

[5] A point $\mathbf{x}^*$ in the state space of a dynamical system is called an equilibrium point if it has the property that whenever the state of the system starts at $\mathbf{x}^*$, it remains at $\mathbf{x}^*$ for all future times. A equilibrium point $\mathbf{x}^*$ is said to be stable if the state of the system converges to $\mathbf{x}^*$ for all sufficiently small disturbances away from it.

Together, this implies that $\mathbf{w}_\mu$ is the only globally stable equilibrium point of the learning dynamics. Hence, the ensemble averaged weight vector $\langle \mathbf{w}(t) \rangle_E$ converges to $\mathcal{W}(\mu)$ for any initial weight vector $\mathbf{w}(0) \in [0, 1]^n$.

We now show the "only if" part of theorem 1. If the target vector can be learned, then for some W_+, $W_- > 0$, we know that for any $\mu > 0$, the ensemble averaged weight vector $\langle \mathbf{w}(t) \rangle_E$ converges to $\mathcal{W}(\mu)$ for any initial weight vector $\mathbf{w}(0) \in [0, 1]^n$. Since $\mathcal{W}(\mu) \in (0, 1)^n$, we can draw $\mathbf{w}(0)$ from a small surrounding of $\mathcal{W}(\mu)$ which is still in $[0, 1]^n$. This implies that $\mathcal{W}(\mu)$ is a stable equilibrium point of $\langle \mathbf{w}(t) \rangle_E$ under the learning dynamics. Hence, for these values of W_+ and W_-, it holds for all $\mu > 0$ that $\mathcal{W}(\mu)$ is a stable equilibrium point of $\langle \mathbf{w}(t) \rangle_E$ under the learning dynamics. This implies the "only if" part of theorem 1.

For a more thorough analysis of the learning equation, we will have to incorporate a specific neuron model. For the integrate-and-fire neuron, no closed formula exists that relates the correlation between inputs and outputs to the neuron parameters. We therefore give an analysis for the linear Poisson neuron model (see section 2; see also Gerstner & Kistler, 2002). The next theorem is the main result of this section. We define the normalized cross correlation between input spike trains S_i and S_j with a common rate $r > 0$ as

$$C_{ij}^0(s) = \frac{\langle S_i(t)\ S_j(t+s) \rangle_E}{r^2} - 1, \tag{4.13}$$

which assumes value 0 for uncorrelated Poisson spike trains. In our neuron model, correlations are shaped by the response kernel $\epsilon(s)$, and they enter the learning equation 4.12 with respect to the learning window. This motivates the definition of window correlations,

$$c_{ij}^+ = 1 + \frac{1}{\tau} \int_0^\infty ds\ e^{-s/\tau} \int_0^\infty ds'\ \epsilon(s') C_{ij}^0(s - s'), \tag{4.14}$$

for the positive learning window and

$$c_{ij}^- = 1 + \frac{1}{\tau} \int_{-\infty}^0 ds\ e^{s/\tau} \int_0^\infty ds'\ \epsilon(s')\, C_{ij}^0(s - s') \tag{4.15}$$

for the negative learning window. In these definitions, the second integral expresses a filtering of the correlation function with the response kernel ϵ. We call the matrices $C^\pm = \{c_{ij}^\pm\}_{i,j=1,\ldots,n}$ the window correlation matrices. Note that window correlations are nonnegative and that for homogeneous

Poisson input spike trains and a nonnegative response kernel, they are positive.[6] We are now ready to formulate an analytical criterion for learnability:

Theorem 2. *A weight vector* $\mathbf{w}^*$ *can be learned for homogeneous Poisson input spike trains with window correlation matrices* C^+ *and* C^- *to a linear Poisson neuron with nonnegative response kernel if and only if* $\mathbf{w}^* \neq 0$ *and*

$$\frac{\sum_{k=1}^{n} w_k^* c_{ik}^+}{\sum_{k=1}^{n} w_k^* c_{ik}^-} > \frac{\sum_{k=1}^{n} w_k^* c_{jk}^+}{\sum_{k=1}^{n} w_k^* c_{jk}^-}$$

for all pairs $\langle i, j\rangle \in \{1, \ldots, n\}^2$ *with* $w_i^* = 1$ *and* $w_j^* = 0$.

This theorem can be interpreted in the following way. The amount of correlation between input i and the output also depends on other inputs k, which are correlated with this input. Furthermore, the impact of such a correlated input depends on its weight. In the linear model, these effects are summed up. Theorem 2 asserts a criterion on the fraction of such summed correlations in the positive and negative learning window. This fraction needs to be larger for synapses that should be potentiated than for synapses that should be depressed.

Proof. We will prove theorem 2 with the help of theorem 1. We therefore first analyze the equilibrium points of equation 4.12 for the linear Poisson neuron model. Consider a linear Poisson neuron with the constant target weight vector w^*. We obtain the correlation function $\langle S_i(t)\ S^*(t+s)\rangle_E$ of input i with the output by inserting the instantaneous rate of the linear Poisson neuron with given input spike trains $S_1, \ldots, S_n$ (see equation 2.3) into Equation 4.11:[7]

$$C_i(s) = \langle S_i(t)\ S^*(t+s)\rangle_E = \sum_{j=1}^{n} w_j^* \int_0^\infty ds'\ \epsilon(s')\ \langle S_i(t)\ S_j(t+s-s')\rangle_E .$$

[6] From equation 4.14, it follows that $c_{ij}^+ = 0$ only if $\int_0^\infty ds\ \langle S_i(t)\ S_j(t+s)\rangle_E = 0$. According to Bayes' theorem, this equality can be rewritten as $\langle S_i(t)\rangle_E \int_0^\infty ds\ \langle S_j(t+s)|\text{spike in } S_i \text{ at time } t\rangle_E = 0$. This implies that either the rate of S_i or the rate of S_j is zero, which contradicts our assumption of positive rate. A similar argument can be applied for c_{ij}^-.

[7] To show that this is valid, we observe that $\langle S_i(t)S^*(t')\rangle_E = \langle S_i(t)\langle S^*(t')\rangle_{E'}\rangle_E$ (this is just a rearrangement of the summation terms). Here, $\langle\,.\,\rangle_{E'}$ indicates the ensemble average over the ensemble for given $S_1, \ldots, S_n$, that is, only $S^*(t')$ is varied.

With the use of simple mathematics, we can rewrite this equation as

$$C_i(s) = r^2 \sum_{j=1}^{n} w_j^* \left[1 + \int_0^{\infty} ds' \, \epsilon(s') \, C_{ij}^0(s - s') \right]. \tag{4.16}$$

Equation 4.16 describes the input-output correlations of a neuron with target weights $\mathbf{w}^*$. Since the output of the teached neuron in our setup is clamped to S^*, these correlations drive learning in the synapses of the taught neuron. Substituting equation 4.16 into 4.12 and using equation 4.13, we can calculate the synaptic drift as

$$\begin{aligned} \dot{w}_i = {} & r^2 W_+ f_+^{\mu}(w_i) \sum_{j=1}^{n} w_j^* \int_0^{\infty} ds \; e^{-s/\tau} \left[1 + \int_0^{\infty} ds' \epsilon(s') \, C_{ij}^0(s - s') \right] \\ & - r^2 W_- f_-^{\mu}(w_i) \sum_{j=1}^{n} w_j^* \int_{-\infty}^{0} ds \; e^{s/\tau} \left[1 + \int_0^{\infty} ds' \epsilon(s') \, C_{ij}^0(s - s') \right]. \end{aligned} \tag{4.17}$$

Equation 4.17 can be rewritten in terms of the window correlations c_{ij}^+ and c_{ij}^- as

$$\dot{w}_i = \tau r^2 \left[W_+ f_+^{\mu}(w_i) \sum_{j=1}^{n} w_j^* c_{ij}^+ - W_- f_-(w_i) \sum_{j=1}^{n} w_j^* c_{ij}^- \right]. \tag{4.18}$$

We find the equilibrium point $w_{\mu i}$ of synapse i by setting $\dot{w}_i = 0$ in equation 4.18. This yields

$$\frac{f_-^{\mu}(w_{\mu i})}{f_+^{\mu}(w_{\mu i})} = \left(\frac{w_{\mu i}}{1 - w_{\mu i}} \right)^{\mu} = \frac{W_+}{W_-} \frac{\sum_{j=1}^{n} w_j^* c_{ij}^+}{\sum_{j=1}^{n} w_j^* c_{ij}^-}, \tag{4.19}$$

which is defined for $\mathbf{w}^* \neq \mathbf{0}$ (note that $w_j^* \geq 0$ for $j = 1, \ldots, n$, and $c_{ij}^+, c_{ij}^- > 0$ for $i, j = 1, \ldots, n$). We denote $\frac{W_+}{W_-} \frac{\sum_{j=1}^{n} w_j^* c_{ij}^+}{\sum_{j=1}^{n} w_j^* c_{ij}^-}$ by Λ_i and find

$$w_{\mu i} = \left(1 + \frac{1}{\Lambda_i^{1/\mu}} \right)^{-1}. \tag{4.20}$$

The equilibrium points of the learning dynamics for given μ and W_+, W_- are therefore given by $\mathbf{w}_{\mu} = \langle w_{\mu 1}, \ldots, w_{\mu n} \rangle$. If we cannot find values for W_+, W_- such that these equilibrium points are stable, then the target function cannot be learned due to theorem 1. The stability analysis of the equilibrium points is based on equation 4.18. One can see that the drift is identical to zero for all W_+, W_- if $\mathbf{w}^* = \mathbf{0}$. In this case, every point in the state space is an equilibrium point, but none is stable. It follows that the target function cannot be learned if $\mathbf{w}^* = \mathbf{0}$.

In the following, we assume that $\mathbf{w}^* \neq \mathbf{0}$. We show that in this case, the equilibrium point is stable for all μ, W_+, $W_- > 0$. We consider a small perturbation δw of a single component w_i from the equilibrium point $w_{\mu i}$. This leads to some drift $\dot{w}_i$ of the perturbed system:

$$\dot{w}_i = \frac{\tau r^2}{n}\left[W_+ f_+^{\mu}(w_{\mu i} + \delta w)\sum_{j=1}^{n} w_j^* c_{ij}^+ - W_- f_-(w_{\mu i} + \delta w)\sum_{j=1}^{n} w_j^* c_{ij}^-\right]. \tag{4.21}$$

For all $\mu > 0$ and $\delta w > 0$, it holds that $f_+^{\mu}(w_i + \delta w) < f_+^{\mu}(w_i)$ and $f_-^{\mu}(w_i + \delta w) > f_-^{\mu}(w_i)$. Because $c_{ij}^+, c_{ij}^- > 0$, the synaptic drift of the perturbed system $\dot{w}_i$ is smaller than the synaptic drift of the system in equilibrium, which is 0. It follows that the synaptic drift is negative for $\delta w > 0$. A similar argument shows that the synaptic drift is positive for $\delta w < 0$. Therefore, the equilibrium point of the system is stable if and only if $\mathbf{w}^* \neq \mathbf{0}$.

To summarize, we know that there exists a function $\mathcal{W} : \mathbb{R}^+ \to (0, 1)^n$ such that for all W_+, W_-, $\mu > 0$, $\mathcal{W}(\mu)$ is a stable equilibrium point of the learning dynamics if and only if $\mathbf{w}^* \neq \mathbf{0}$. Here, we can identify $\mathcal{W}(\mu)$ with $\mathbf{w}_\mu$. If we compare this statement with theorem 1, we can deduce that the target vector $\mathbf{w}^*$ can be learned if and only if $\mathbf{w}^* \neq \mathbf{0}$ and $\lim_{\mu\to 0} w_{\mu i} = w_i^*$ for all $i \in \{1, \ldots, n\}$. In the following, we show that this criterion is indeed equivalent to the criterion given in theorem 2.

We define two sets of indices M and $\bar{M}$, where M contains all indices i with $w_i^* = 1$ and $\bar{M}$ contains all indices i with $w_i^* = 0$. More formally, we define $M = \{i \in \{1, \ldots, n\} | w_i^* = 1\}$ and $\bar{M} = \{i \in \{1, \ldots, n\} | w_i^* = 0\}$. Note that $\lim_{\mu\to 0} w_{\mu i} = 1$ if and only if $\Lambda_i > 1$. Furthermore, $\lim_{\mu\to 0} w_{\mu i} = 0$ if and only if $\Lambda_i < 1$. Therefore, $\lim_{\mu\to 0} \mathbf{w}_\mu = \mathbf{w}^*$ holds if and only if $\Lambda_i > 1$ for all $i \in M$ and $\Lambda_i < 1$ for all $i \in \bar{M}$. By the definition of Λ_i, this statement is equivalent to the following statement: $\lim_{\mu\to 0} \mathbf{w}_\mu = \mathbf{w}^*$ if and only if

$$\frac{W_-}{W_+} < \frac{\sum_{j=1}^{n} w_j^* c_{ij}^+}{\sum_{j=1}^{n} w_j^* c_{ij}^-} \quad \text{for all } i \in M, \text{ and} \tag{4.22}$$

$$\frac{W_-}{W_+} > \frac{\sum_{j=1}^{n} w_j^* c_{ij}^+}{\sum_{j=1}^{n} w_j^* c_{ij}^-} \quad \text{for all } i \in \bar{M}. \tag{4.23}$$

Equations 4.22 and 4.23 can be taken together to form a single criterion: $\lim_{\mu \to 0} \mathbf{w}_\mu = \mathbf{w}^*$ if and only if

$$\frac{\sum_{k=1}^{n} w_k^* c_{ik}^+}{\sum_{k=1}^{n} w_k^* c_{ik}^-} > \frac{\sum_{k=1}^{n} w_k^* c_{jk}^+}{\sum_{k=1}^{n} w_k^* c_{jk}^-} \quad \text{for all pairs } \langle i, j \rangle \text{ with } i \in M \text{ and } j \in \bar{M}. \tag{4.24}$$

If condition 4.24 is satisfied, we know that we can find values for W_+, $W_- > 0$ such that conditions 4.22 and 4.23 are satisfied. On the other hand, if condition 4.24 is not satisfied, there are no such values. Note that condition 4.24 is satisfied if no such pairs exist (i.e., $w_i^* = 1$ for all i). In this case, we can choose W_- / W_+ arbitrarily small to guarantee convergence. Hence, we have shown that a target vector $\mathbf{w}^*$ can be learned if and only if $\mathbf{w}^* \neq \mathbf{0}$ and condition 4.24 is satisfied. This concludes the proof of theorem 2.

For a wide class of cross-correlation functions, one can establish a relationship between learnability by STDP and the well-known concept of linear separability from linear algebra.

Definition 2. *Let $\mathbf{c}_1, \ldots, \mathbf{c}_m \in \mathbb{R}^n$ and $y_1, \ldots, y_m \in \{0, 1\}$. We say that a vector $\mathbf{w} \in \mathbb{R}^n$ linearly separates the list $\langle \langle \mathbf{c}_1, y_1 \rangle, \ldots, \langle \mathbf{c}_m, y_m \rangle \rangle$ if there exists a threshold Θ such that $y_i = sign(\mathbf{c}_i \cdot \mathbf{w} - \Theta)$ for $i = 1, \ldots, m$.*

The perceptron convergence theorem asserts that a list of training examples can be learned if a weight vector exists that separates the list (i.e., if the list is linear separable). We will show that the definition of linear separability turns out to be useful also in the context of spiking neurons if it is applied to the window correlation matrix C^+ of input spike trains. Because of synaptic delays, the response of a spiking neuron to an input spike is delayed by some time t_0. One can model such a delay in the response kernel by the restriction $\epsilon(s) = 0$ for all $s \leq t_0$.[8] The following corollary asserts that if input correlations $C_{ij}^0(s)$ vanish for time differences $s < -t_0$ (i.e., cross correlations appear only in a time window smaller than the delay), then learnability can be stated in terms of linear separability. As shown in the the proof of corollary 1, this condition implies that $c_{ij}^- = 1$ for all i, j.

Corollary 1. *If there exists a $t_0 \geq 0$ such that the response kernel $\epsilon(s) = 0$ for all $s \leq t_0$ and $C_{ij}^0(s) = 0$ for all $s < -t_0$, $i, j = 1, \ldots, n$, and the window correlation matrix C^+ is positive, then the following holds for the case of homogeneous Poisson*

[8] Different synapses have different delays. Here, we consider only a single delay t_0 for all synapses. However, this assumption is not critical for the analysis. It can easily be generalized to various different delays.

input spike trains to a linear Poisson neuron with response kernel ϵ*: A weight vector* $\mathbf{w}^*$ *can be learned if and only if* $\mathbf{w}^* \neq 0$ *and* $\mathbf{w}^*$ *linearly separates the list* $L = \langle\langle \mathbf{c}_1^+, w_1^* \rangle, \ldots, \langle \mathbf{c}_n^+, w_n^* \rangle\rangle$*, where* $\mathbf{c}_1^+, \ldots, \mathbf{c}_n^+$ *are the rows of* C^+*.*

Corollary 1 can be viewed as an analogon of the perceptron convergence theorem for the average case analysis of STDP.

Proof. The window correlations c_{ij}^- are given by

$$\begin{aligned} c_{ij}^- &= 1 + \frac{1}{\tau}\int_{-\infty}^{0} ds\; e^{s/\tau} \int_0^{\infty} ds'\; \epsilon(s') C_{ij}^0(s-s') \\ &= 1 + \frac{1}{\tau}\int_{-\infty}^{0} ds\; e^{s/\tau} \left[\int_0^{t_0} ds'\; \epsilon(s') C_{ij}^0(s-s') \right. \\ &\quad \left. + \int_{t_0}^{\infty} ds'\; \epsilon(s') C_{ij}^0(s-s')\right] \\ &= 1\,. \end{aligned}$$

The first integral in the square brackets vanishes because $\epsilon(s') = 0$ for $s' \in [0, t_0]$. The second integral in the square brackets vanishes because $C_{ij}^0(s - s') = 0$ for $s - s' < -t_0$ and $\epsilon(t_0) = 0$.

We can apply theorem 2. The inequality in theorem 2 becomes $\frac{\sum_{k=1}^n w_k^* c_{ik}^+}{\sum_{k=1}^n w_k^*} > \frac{\sum_{k=1}^n w_k^* c_{jk}^+}{\sum_{k=1}^n w_k^*}$. Let $M = \{i \in \{1, \ldots, n\} | w_i^* = 1\}$ and $\bar{M} = \{i \in \{1, \ldots, n\} | w_i^* = 0\}$. We find that the weight vector can be learned if and only if $\mathbf{w}^* \neq \mathbf{0}$ and

$$\sum_{k=1}^n w_k^* c_{ik}^+ > \sum_{k=1}^n w_k^* c_{jk}^+ \tag{4.25}$$

for all pairs $\langle i, j \rangle$ with $i \in M$ and $j \in \bar{M}$.

It remains to be shown that condition 4.25 is equivalent to the statement that $\mathbf{w}^*$ linearly separates the list $L = \langle\langle \mathbf{c}_1^+, w_1^* \rangle, \ldots, \langle \mathbf{c}_n^+, w_n^* \rangle\rangle$. Condition 4.25 is satisfied if and only if there exists some threshold Θ such that $\mathbf{w}^* \cdot \mathbf{c}_i^+ > \Theta > \mathbf{w}^* \cdot \mathbf{c}_j^+$ for all pairs $\langle i, j \rangle$ with $i \in M$ and $j \in \bar{M}$. This is equivalent to the condition that there exists some threshold Θ such that $sign(\mathbf{c}_i^+ \cdot \mathbf{w}^* - \Theta) = w_i^*$ for all $i = 1, \ldots, n$. Therefore condition 4.25 holds if and only if $\mathbf{w}^*$ linearly separates L.

The formulation of corollary 1 is tight in the sense that linear separability of the list L alone (as opposed to linear separability by the target vector $\mathbf{w}^*$) is not sufficient to imply learnability. This follows from the following fact:

Proposition 1. *There exists a window correlation matrix $C^+ = \{c_{ij}^+\}_{i,j=1,\ldots,n}$ with window correlations c_{ij}^+ and there exist vectors $\mathbf{w}, \mathbf{w}^* \in \{0, 1\}^n$, such that $\mathbf{w}$ linearly separates the list $L = \langle\langle \mathbf{c}_1^+, w_1^* \rangle, \ldots, \langle \mathbf{c}_n^+, w_n^* \rangle\rangle$ but $\mathbf{w}^*$ does not linearly separate L. Thus, the list L is linearly separable, but the target vector $\mathbf{w}^*$ cannot be learned by STDP.*

Proof. Consider homogeneous Poisson input spike trains of rate r that have normalized cross correlation functions of the form $C_{ij}^0(s) = \frac{c_{ij}}{r}\delta(s)$ with nonnegative correlation coefficients c_{ij}. Let C denote the matrix with entries c_{ij} and $\mathbf{c}_i$ denote the ith row of C. Furthermore consider some response kernel ϵ with $\epsilon(s) = 0$ for $s \leq 0$. Obviously, we can apply corollary 1 here. The positive window correlation functions are of the form $c_{ij}^+ = 1 + c_{ij}\gamma$ for some constant $\gamma > 0$. One can show that for target values $y_1, \ldots, y_n \in \{0, 1\}$, the list $L = \langle\langle \mathbf{c}_1^+, y_1 \rangle, \ldots, \langle \mathbf{c}_n^+, y_n \rangle\rangle$ is linearly separated by a vector $\mathbf{w} \in \{0, 1\}^n$ if and only if $\mathbf{w}$ linearly separates the list $L = \langle\langle \mathbf{c}_1, y_1 \rangle, \ldots, \langle \mathbf{c}_n, y_n \rangle\rangle$ (see appendix C). We will therefore consider the matrix C of correlation coefficients c_{ij} instead of C^+.

Consider the matrix and vectors

$$C = \begin{pmatrix} 1 & 0.25 & 0.1 & 0.5 & 0 \\ 0.25 & 1 & 0.1 & 0.5 & 0 \\ 0.1 & 0.1 & 1 & 0.05 & 0 \\ 0.5 & 0.5 & 0.05 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \mathbf{w}^* = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \mathbf{w} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}.$$

The list $L = \langle\langle \mathbf{c}_1, w_1^* \rangle, \ldots, \langle \mathbf{c}_n, w_n^* \rangle\rangle$ where $\mathbf{c}_i$ is the ith row vector of C is not separated by $\mathbf{w}^*$, because $C\mathbf{w}^* = (1.35, 1.35, 1.2, 1.05, 1)^T$. However, $\mathbf{w}$ separates L because $C\mathbf{w} = (0.1, 0.1, 1, 0.05, 1)^T$. One can show that there exist Poisson spike trains with correlation matrix C (see Legenstein & Maass, 2004).

For uncorrelated input spike trains of rate $r > 0$, each input spike train is correlated only with itself and only for zero time lag. Thus, the normalized cross-correlation functions are given by $C_{ij}^0(s) = \frac{\delta_{ij}}{r}\delta(s)$, where δ_{ij} is the Kronecker delta function. In this case, the condition for corollary 1 is satisfied for every response kernel ϵ with $\epsilon(s) = 0$ for $s \leq 0$. Furthermore, the positive window correlations are given by $c_{ij}^+ = 1 + \delta_{ij}\gamma$ for some constant $\gamma > 0$. For arbitrary target values $y_1, \ldots, y_n$, a weight vector $\mathbf{w}$ separates the corresponding list list $L = \langle\langle \mathbf{c}_1^+, y_1 \rangle, \ldots, \langle \mathbf{c}_n^+, y_n \rangle\rangle$ if and only if $\mathbf{w}$ separates the list $L' = \langle\langle \mathbf{e}_1, y_1 \rangle, \ldots, \langle \mathbf{e}_n, y_n \rangle\rangle$ where the vectors $\mathbf{e}_1, \ldots, \mathbf{e}_n$ are the the column vectors of the identity matrix (see appendix C). But every weight vector $\mathbf{w}^* \in \{0, 1\}^n$ with $\mathbf{w} \neq \mathbf{0}$ separates the list $\langle\langle \mathbf{e}_1, w_1^* \rangle, \ldots, \langle \mathbf{e}_n, w_n^* \rangle\rangle$. Hence, for a window correlation matrix with such entries, every weight vector $\mathbf{w}^* \in \{0, 1\}^n$ separates the corresponding list. Thus, with suitable values of

W_- and W_+, any target weight vector $\mathbf{w}^* \in \{0, 1\}^n$ with $\mathbf{w}^* \neq \mathbf{0}$ can be learned for the case of uncorrelated Poisson input spike trains:

Corollary 2. *A target weight vector* $\mathbf{w}^* \in \{0, 1\}^n$ *can be learned in the case of uncorrelated Poisson input spike trains to a linear Poisson neuron with response kernel* ϵ *such that* $\epsilon(s) = 0$ *for all* $s \leq 0$ *if and only if* $\mathbf{w}^* \neq \mathbf{0}$.

Equations 4.21 and 4.22 give necessary conditions for the relationship between long-term depression and long-term potentiation for successful learning. For uncorrelated Poisson input, equation 4.23 predicts that W_- / W_+ has to be larger than 1. By equation 4.22, this fraction is bounded from above by

$$\frac{W_-}{W_+} < 1 + \frac{w_i^*}{\tau r \sum_{j=1}^{n} w_j^*}. \tag{4.26}$$

As described in section 4.1, an optimal fraction W_- / W_+ lies halfway between 1 and the upper extreme of this inequality. For increasing n and a constant fraction of nonzero weights, the sum in the denominator of equation 4.26 becomes larger. Equation 4.26 therefore predicts that this ratio drops with increasing n (see experiment 1 in section 5).

For uncorrelated Poisson input and with different powers μ_+ and μ_-, equation 4.19, which describes the fixed point of a synapses i, reads

$$\frac{w_{\mu i}^{\mu_-}}{(1 - w_{\mu i})^{\mu_+}} = \frac{W_+}{W_-}\left(1 + \frac{1}{\tau r}\frac{w_i^*}{\sum_{j=1}^{n} w_j^*}\right). \tag{4.27}$$

Note that this equation holds not only for binary target vectors $\mathbf{w}^*$ but also for continuous target vectors $\mathbf{w}^* \in [0, 1]^n$. The learning rule therefore reflects the ordering of the target weights $\mathbf{w}^*$ in its equilibrium point (i.e., for two synapses i, j with $w_i^* > w_j^*$, we get $w_{\mu i} > w_{\mu j}$). With appropriate parameters μ_+, μ_-, W_+, and W_-, one should be able to learn a good approximation to $\mathbf{w}^*$. This is confirmed with computer simulations in section 6.2. However, this ordering breaks down for correlated inputs, because cross correlations between inputs have a strong influence on the equilibrium points of the learning dynamics.

5 Computer Simulations of Supervised Learning with STDP: Weight Modulations

We have shown through computer simulations that in spite of the negative result from section 3 for the SNCC in a worst-case input scenario, the SNCC for STDP is approximately satisfied for Poisson input spike trains, with andwithout correlations among them. This positive result is not surprising in view of the theoretical predictions of section 4, but it is not automatically implied by the preceding theory. In order to make a theoretical analysis feasible, we needed to make in section 4 a number of simplifying assumptions on the neuron model (linear Poisson neuron) and the synapse model (static synapses). In addition, a number of approximations had to be used in order to simplify the estimates; for example, we had analyzed only ensemble average and drift and had assumed that the impact of stochastic fluctuations could be ignored. As a consequence, we will see for the more realistic models of neurons and synapses that the weight vector in general does not converge to the target vector, but rather fluctuates in the neighborhood of the target vector.

We consider in this section and in sections 6 and 7 the more realistic models for neurons and synapses discussed in section 2. We also show that in some cases, a less restrictive teacher forcing suffices that tolerates undesired firing of the neuron during training. Details on the simulations can be found in appendix A.

Apart from the failure of common rules for STDP to respond appropriately (by a suitable reduction of weights of excitatory synapses) to "negative examples," where the neuron fires although it should not fire, we identified in section 3 two other structural differences between the perceptron learning rule and STDP:

i. STDP cannot change the "sign" of a synapse.

iii. STDP keeps changing synaptic parameters for inputs that are already processed in the desired way by the neuron.

In all our simulations, we apply STDP just to excitatory synapses (and they remain excitatory), whereas the parameters of inhibitory synapses remain unchanged (largely because of a lack of commonly accepted experimental data on STDP for "generic" inhibitory synapses). We show that the resulting structural difference i to the perceptron learning rule causes no problem for the convergence of learning in the computer experiments discussed in this letter (note that no inhibitory inputs were considered in the theoretical analysis of section 4).

The problem iii certainly has an impact insofar as it causes never-ending fluctuations around the target vector and does not allow a locking onto the target vector after finitely many steps as in the case of perceptron learning. The theoretical analysis of section 4 had assumed that the neuron never fires during training except when it is supposed to fire. In the subsequent computer simulations, the neuron received a strong depolarizing input when it was supposed to fire and a hyperpolarizing input, which prevented most (but not all) undesired firing, when it was not supposed to fire. It turns out that the use of such hyperpolarizing teacher input is not necessary if one instead starts the learning with small (randomly assigned) initial weights.

With large initial weights and without hyperpolarizing teacher input, learning capabilities are weak (results not shown).

5.1 Experiment 1 (Uncorrelated Input). In this experiment, a leaky integrate-and-fire neuron received inputs from $n = 100$ dynamic synapses; 90% of these synapses were excitatory and 10% were inhibitory. For each excitatory synapse, the maximal efficacy $w_{\max}$ was chosen from a gaussian distribution with mean 54 and SD 10.8, bounded by $54 \pm 3SD$[9]. Target weight vectors $\mathbf{w}^*$ were chosen as follows. We randomly selected one-half of the excitatory synapses and set their weights to the corresponding maximal efficacy $w_{\max}$. The weights of the other excitatory synapses were set to zero. The resulting target weight vector $\mathbf{w}^*$ was then used to define a transformation F, which maps 100 input spike trains to one output spike train. The threshold of the neuron was set such that the rate of the output spike train was approximately 25 Hz for an input consisting of 100 uncorrelated Poisson spike trains with a rate of 20 Hz (this input rate was used for all subsequent experiments, except for experiment 5).

We then replaced the weights of all excitatory synapses by new, randomly chosen values according to a gamma distribution with mean 9 and standard deviation 6.3. Weights of inhibitory synapses remained fixed throughout the experiment (this also holds for all other experiments discussed in this article). We then examined whether the neuron can learn with STDP to reproduce the previously defined transformation F from input spike trains to output spikes for an input consisting of 100 uncorrelated Poisson spike trains at a rate of 20 Hz. Information about the target transformation F was given to the neuron only in the form of short current injections (1 μA for 0.2 ms) at those times when this transformation F (i.e., the neuron with the weight vector $\mathbf{w}^*$) would have produced a spike. Learning was implemented as standard STDP (see rule 3.2) with parameters $\tau_+ = \tau_- = \tau = 20$ ms, $W_+ = 0.3$, and $W_- / W_+ = 1.035$.

The learning simulation was performed for 3600 seconds of simulated biological time with one long input sequence (i.e., without repetition of identical spike trains). Longer simulations (4 hours simulated biological time) were performed to test the stability of results. No significant changes in the results were observed for these runs. Results of a typical learning trial are shown in Figure 3.

Three different performance measures were used for analyzing the learning progress (see the three curves in Figure 3B). The most informative one ("spike correlation," plotted in Figure 3B with a dotted line) measures for test inputs that were not used for training (but had been generated by the same process) the deviation between the output spike train produced by the target transformation F for this input, and the output spike train pro-

[9] Values lower than 21.6 (greater than 86.4) were replaced by 21.6 (86.4).

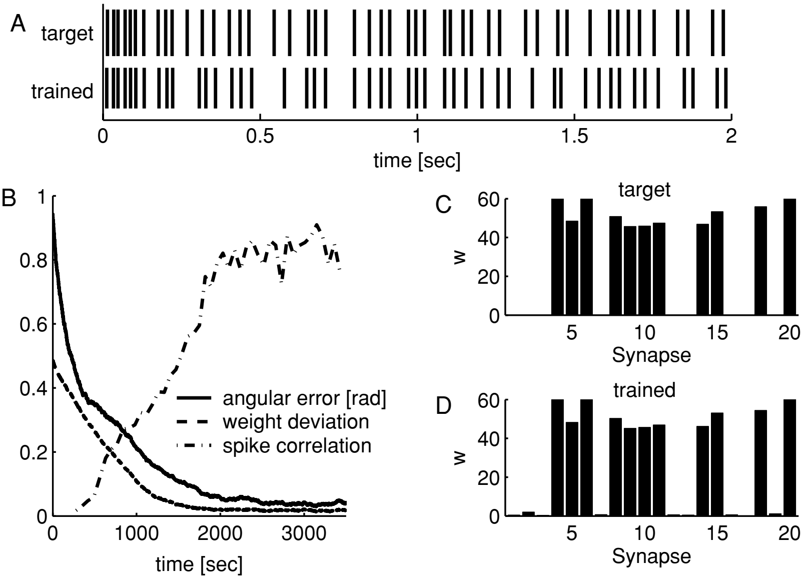

Figure 3: Learning an arbitrary transformation F on 100 uncorrelated Poisson inputs. (A) Output spike train on test data after 1 hour of training (trained) compared to the output of the target transformation F (target). (B) Evolution of the angle between weight vector $\mathbf{w}(t)$ and the vector $\mathbf{w}^*$ that implements F in radiant (angular error, solid line), the weight deviation (dashed line), and spike correlation (dotted line). (C) Twenty weights from the vector $\mathbf{w}^*$ (each weight has its maximal possible value or value 0). (D) Corresponding weights of the learned vector $\mathbf{w}(t)$ after 1 hour of training.

duced for the same input by the neuron with the current weight vector $\mathbf{w}(t)$. For that purpose, each spike in these two output spike trains was replaced by a gaussian function with an SD of 5 ms. The spike correlation between both output spike trains was defined as the correlation between the resulting smooth functions of time (for segments of length 100 s). This measure penalizes missing or superfluous spikes produced by the trained neuron, but also imprecision in timing of spikes on the scale of a few ms. The other two measures are obtained by comparing directly the current weight vector $\mathbf{w}(t)$, with the target weight vector $\mathbf{w}^*$. The angular error measures the angle between these two vectors (solid line in Figure 3B). Note that this measure does not reflect differences in the magnitude of vectors, in contrast to the third measure: weight deviation. Weight deviation is the mean absolute weight difference normalized by the mean target weight. Thus, the weight deviation can be computed as $\frac{\sum_{i=1}^{n_e} |w_i^* - w_i(t)|}{\sum_{i=1}^{n_e} w_i^*}$, with n_e being the number of excitatory weights. Note that the latter two measures are very direct, but they can be deceptive, since in general, several different weight vectors can produce good approximations to the target transformation F (especially if inputs are strongly correlated). Figures 3C and 3D show for an arbitrary subset of 20 of the 90 excitatory synapses the values of weights in $\mathbf{w}^*$ (see Figure 3C) and $\mathbf{w}(t)$ (see Figure 3D) for $t = 3600$ s. The weights in $\mathbf{w}^*$ have either value 0 or the randomly chosen maximal value $w_{\max}$ for that weight. The results shown in Figure 3 demonstrate that the spiking neuron with dynamic synapses was able to learn with STDP after about 30 minutes of training the target transformation F quite well, and further learning with STDP did not reduce the quality of the approximation. Although the chosen spike correlation measure equals zero for uncorrelated Poisson spike trains of a common rate, we tested the spike correlation of randomly chosen weight vectors (instead of the learned vector). The spike correlation produced by 20 weight vectors drawn from the same distribution as the target weight vector $\mathbf{w}^*$ was 0.24 ± 0.04 (mean ± SD). Hence, the spike correlations achieved are far above chance level.

In order to test whether this positive result is representative, we carried out 100 repetitions of the same experiment with different target vectors $\mathbf{w}^*$, different initialization $\mathbf{w}(0)$ of the weight vector before learning, and different numbers of inputs. Twenty repetitions of the experiment (always with new Poisson spike trains) were carried out for five different dimensions (i.e., for five different numbers of simultaneously injected Poisson spike trains) between 25 and 200. The quotient W_- / W_+ was set to 1.12, 1.05, 1.035, 1.025, 1.0175 for 25, 50, 100, 150, and 200 inputs respectively. Results are shown in Figure 4. Figure 4A and 4B show that randomly chosen target transformations F are learned quite well with STDP, with only slight deterioration of performance even for biologically realistic large numbers of input spike train. The required training time increases roughly linearly with the number of inputs, but stays within a reasonable range.

5.2 Experiment 2 (Noisy Teacher). In a realistic scenario of prediction learning, the predicted inputs are likely to have some timing jitter. We therefore repeated experiment 1 with the timing of "teacher spikes" jittered by gaussian noise with zero mean and SD 4 ms. In this case, learning took considerably longer (65 ± 12 minutes convergence time until an angular error of ≤ 10 degrees was achieved for the case 100 input spike trains, for 20 repetitions of the experiment; 500 minutes simulated training time), and yielded the following results: spike correlation 0.67 ± 0.1, angular error 7.5 ± 1.9 degrees, weight deviation $2.3 \pm 0.5\%$, for $W_+ = 0.045$, $W_- / W_+ = 1.0055$.[10]

[10] Somewhat better results can be achieved with additional inhibitory input that reduces non-teacher-induced firing (see experiment 3 for details). One then gets a spike correlation of 0.73 ± 0.16.

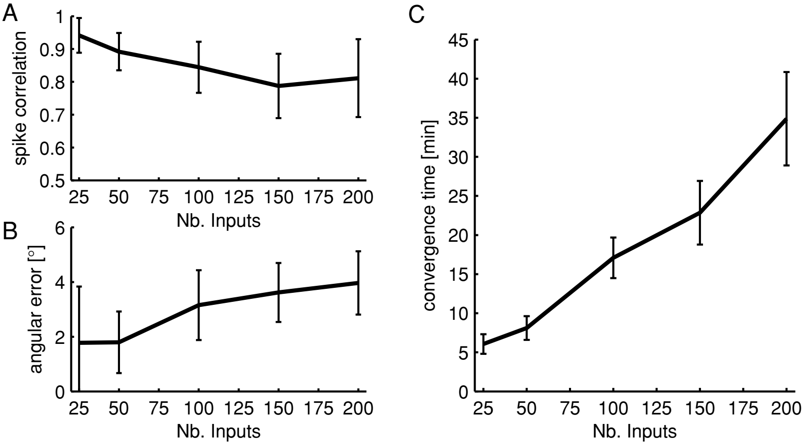

Figure 4: Results on different input sizes. For each input size, the simulation was repeated 20 times for different target transformations F, different inputs, and different initial conditions. The mean and standard deviation is shown for spike correlation (A) and angular error (B) after 1 hour of training. (C) Training time needed until an angular error of less than 10 degrees is achieved.

5.3 Experiment 3 (Correlated Input). There exist many correlations among spike trains from different neurons in a neural system, and therefore we have also carried out a variation of experiment 1 where different subgroups of input spike trains had different degrees of correlation.

In this setup, inputs with weight 0 in the generation of the target transformation are correlated with the output. The reason is that such inputs are correlated with other inputs that have a positive weight and correlations with the output. Furthermore, stronger correlated groups have a stronger influence on the output. In the extreme case, weighted inputs of input groups with small correlation within the group may be less correlated with the output than nonweighted inputs within strongly correlated groups. Again, equations 4.5 and 4.6 help to predict successful learning and determine a suitable quotient of W_- / W_+.

The experimental setup was similar to that of experiment 1. The 90 excitatory inputs were divided into 9 groups of 10 synapses per group. Spike trains were correlated within groups, whereas there were virtually no correlations between spike trains of different groups.

Correlated spike trains with given correlation coefficients cc and given decays τ_{cc} of correlations for time-shifted versions of such spike trains were generated according to the methods that were introduced and analyzed in Gütig et al. (2003). More precisely, spike trains S_i, S_j were generated such that the correlation function $C_{ij}(\Delta t) = \langle S_i(t) S_j(t + \Delta t) \rangle_t$ of S_i and S_j is exponentially decaying as a function of $|\Delta t|$, with some small time constant τ_{cc} (see appendix A). The correlation coefficient cc_i within group i consisting of 10 spike trains was set to $0.1 * (i - 1)$ for $i = 1, \ldots, 9$. The time constant of decay τ_{cc} was set to 10 ms.[11]

Target transformations F where all synapses belonging to the same group of size 10 are all assigned the weight 0 or the maximal possible value can be learned as well as the target transformation considered in experiment 1. Therefore, we have focused on the more difficult case where target transformations F have to be learned that require different weights for highly correlated input spike trains. More precisely, we have chosen the most difficult case: target transformations F that were generated by assigning within each of the 9 groups of the 10 excitatory synapses to 5 of them the weight 0 and to 5 of them their maximal weight value $w_{\max}$ (which was again chosen randomly for each synapse as in experiment 1).

Figure 5A shows a typical weight vector that results in this way. Note that learning is based not only on teacher spikes but also on non-teacher-induced firing. Therefore, in addition to the difficulties noted above, strongly correlated groups of inputs tend to cause autonomous (i.e., not teacher-induced) firing of the neuron, which results in weight increases for all weights within the corresponding group of synapses according to well-known results for STDP (Song et al., 2000; Gütig et al., 2003). Obviously this effect makes it quite hard to learn a target transformation F that requires that half of the weights for each correlated group have value 0.

However, spike correlations of 0.79 ± 0.09 could still be achieved (20 runs, angular error 14.1 ± 10 degrees, weight deviation 8.6 ± 6.3 after 1 hour of training, convergence time 716 ± 359 s until an angular error of ≤ 10 degrees is reached, for $W_+ = 0.45$, $W_- / W_+ = 1.05$).

The performance was better if additional inhibitory input was given to the neuron that reduced the occurrence of non-teacher-induced firing of the neuron. We added 30 inhibitory synapses with weights drawn from a gamma distribution with mean 25 and standard deviation 7.5 that received additional 30 uncorrelated Poisson spike trains at 20 Hz. The weight vector $\mathbf{w}(t)$ resulting after 1 hour of learning in the presence of such additional inhibitory input is shown in Figure 5B. One can see that the deviation from the target weight vector $\mathbf{w}^*$ shown in Figure 5A is very small, even for highly correlated groups of synapses with heterogeneous target weights.

On 20 trials (each with a new random distribution of maximal weights $w_{\max}$ as in experiment 1, and hence with a new target transformation F), the mean spike correlation after 1 hour of training was 0.83 ± 0.08, with an

[11] The peak correlation of the cross-correlation function is actually smaller. The correlation factor cc is obtained in the limit of $\tau_{cc} = 0$. cc_i can be interpreted as the correlation present in a large time window. Since the time constant for STDP used is 20 milliseconds, this definition is reasonable and more realistic than correlations with $\tau_{cc} = 0$ (i.e., exact coincidence of spikes).

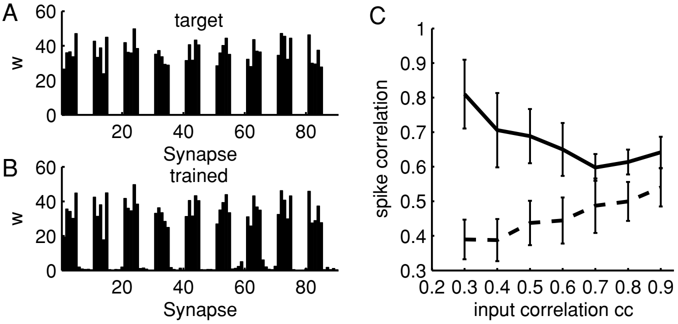

Figure 5: Effects of correlated inputs. (A) A typical target weight vector $\mathbf{w}^*$ for experiment 3 (each weight has its maximal possible value or value 0) and (B) a typical learned weight vector. No significant loss of accuracy can be seen for weights of synapses that receive highly correlated input spike trains ($cc = 0.8$ for synapses 81 to 90) in comparison with synapses that receive weakly correlated ($cc = 0.1 \cdot (i - 1)$ for the ith group) or uncorrelated inputs (e.g., synapses 1 to 10). (C) The result of experiment 4 with sharper correlation ($\tau_{cc} = 6$ ms instead of 10 ms) and 4 groups with the correlation cc plotted on the x-axis (solid line). It also shows as a dashed line the spike correlation achieved by randomly drawn weight vectors (where half of the weights were set to $w_{\max}$ and the other weights were set to 0).

angular error of 6.8 ± 4.7 degrees and a weight deviation of $4.25 \pm 2.2\%$. The spike correlation produced by 20 weight vectors drawn from the same distribution as the target weight vector $\mathbf{w}^*$ was 0.45 ± 0.05.

5.4 Experiment 4 (Dependence of Learning Performance on Input Correlation). In order to evaluate the dependence of correlation among inputs, we proceeded similarly as in experiment 3, but increased and sharpened the correlation among inputs. Now 4 groups consisting each of 10 input spike trains were constructed for which the correlations within each group had the same value cc (the input spike train to the other 50 excitatory synapses, were uncorrelated, as were the inputs to 10 inhibitory synapses; 30 extra uncorrelated inhibitory inputs were added during training as in experiment 3 to reduce undesired firing). In order to make the effects of these correlated inputs more pronounced, the time constant τ_{cc} for the temporal decay of input correlations was reduced from 10 to 6 ms. Target transformations F were chosen as in experiment 3 in the most adverse way: half of the weights of $\mathbf{w}^*$ within each correlated group were set to 0, the other half to a randomly chosen maximal value. The learning performance after 1 hour of training for 20 trials is plotted in Figure 5C for seven different values of the correlation cc that is applied in four of the input groups (solid line). The quotient W_- / W_+ was set to 1.05, 1.055, 1.06 for correlations of 0.3, 0.4, and higher correlations, respectively. Note that higher correlations induce more correlation of unweighted inputs with the output. Due to equation 4.5, this implies larger W_{-}/W_{+} for larger correlations. W_{+} was set to 0.45. One sees that highly correlated inputs do indeed reduce the performance of learning "difficult" target transformation F with STDP. The resulting correlation between the target output spike train produced by F and the output spike train produced by the neuron with weight vector $\mathbf{w}(t)$ after training is not too bad, even for highly correlated inputs (0.64 ± 0.05 for $cc = 0.9$), although the learned weight vector $\mathbf{w}(t)$ is far off the target vector $\mathbf{w}^{*}$ (angular error of 40 ± 3.4 degrees for $cc = 0.9$). In this case, many different weight vectors produce quite similar output spike trains since the majority of output spikes of F are caused by correlated activity in one of the four correlated input groups, and redistribution of weights within each correlated group causes only slight changes in the output spike trains (see the dashed line in Figure 5C). Furthermore, STDP is not well suited for selecting the right ones within these correlated groups for weight amplification.

In order to test the approximate validity of theorem 2 for leaky integrate-and-fire neurons and dynamic synapses, we repeated the above experiment for input correlations $cc = 0.1, 0.2, 0.3, 0.4$, and 0.5. For each correlation value, 20 learning trials (with different target vectors) were simulated. Sixty-five percent of the 100 learning trials were classified as being learnable. The normalized cross correlation between inputs i and j (see equation 4.13) is approximately given by $C_{ij}^{0}(s) = \frac{cc}{2\tau_{cc}r}e^{-|s|/\tau_{cc}}$ for a mean input rate of $r = 20$ Hz and a correlation decay constant of $\tau_{cc} = 6$ ms. We had to choose a response kernel ϵ such that $\epsilon(s)$ reflects the probability of spiking of the integrate-and-fire neuron as a function of time s since an input spike. This is experimentally measured with the peristimulus time histogram (PSTH). For an integrate-and-fire neuron without synaptic noise, the PSTH is proportional not to the shape of the EPSP but to its derivative (see Herrmann & Gerstner, 2001). Since the derivative of the EPSP also assumes negative values and its integral from 0 to infinity is vanishing, we could not use it for the analysis (we assumed in the analysis that the response kernel is positive and that its integral equals 1). Instead, we determined the PSTH of the neuron in simulations and fitted a double exponential to its positive part. This resulted in a response kernel of the form $\epsilon(s) = \frac{1}{\tau_1 - \tau_2}(e^{-s/\tau_1} - e^{-s/\tau_2})$ with $\tau_1 = 2$ ms and $\tau_2 = 1$ ms (least mean squares fit).

For this model, we calculated the window correlations c_{ij}^{+} and c_{ij}^{-} numerically. For each trial, we first checked whether the (randomly chosen) target vector $\mathbf{w}^{*}$ was learnable according to the condition given in theorem 2 (note that any rescaling of the target weight vector does not change the result). The actual performance of learning with STDP was evaluated after 50 minutes of training. To guarantee the best possible performance for each learning trial, training was performed on 27 different values for W_{-}/W_{+} between 1.02 and 1.15. In each trial, the best performance was chosen to evaluate the quality of convergence. The result is shown in Figure 6. Figure 6 shows

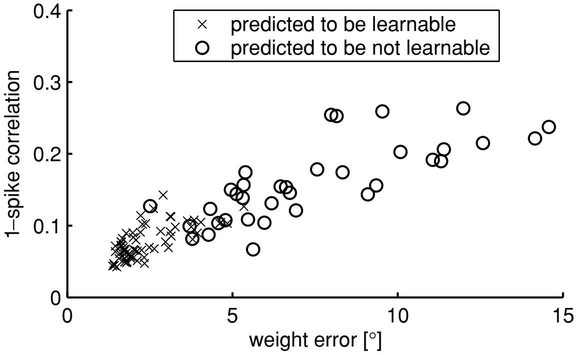

Figure 6: Comparison between theory and simulation results for a leaky integrate-and-fire neuron for input correlations between 0.1 and 0.5 ($\tau_{cc} =$ 6 ms). Each cross marks a trial where the target vector was learnable according to theorem 2. Each open circle marks a trial that is not learnable according to theorem 2. The actual learning performance of STDP is plotted for each trial in terms of the weight error (x-axis) and 1 minus the spike correlation (y-axis).

that the theoretical prediction of learnability or nonlearnability for the case of simpler neuron models and synapses from theorem 2 (which was in addition derived under some simplifying statistical assumptions) translates in a biologically more realistic scenario into a quantitative grading of the learning performance that can ultimately be achieved with STDP.

5.5 Experiment 5 (Time-Varying Input Rates). Good learning results were also obtained using spike trains with time-varying correlated firing rates as inputs. The algorithm we used to produce such inputs had been introduced in Song et al., (2000). This algorithm generates time-varying firing rates that have a cross-correlation function that decays exponentially with a time constant τ_c and an amplitude given by parameters called correlation parameters (see appendix A). Specifically, the correlation between the rates of two inputs i and j is $c_i c_j$, where c_i and c_j are the correlation parameters of these inputs. We assigned to the $n = 90$ excitatory inputs correlation parameters that varied between 0.2 and 0.9 (specifically, c_i of input i was set to $0.2 + 0.7(i - 1)/(n - 1)$. The time constant τ_c was set to 20 ms. In 20 learning trials, spike correlation was 0.89 ± 0.07, angular error was 4.7 ± 3.2 degrees, and weight deviation was $2.7 \pm 1\%$ (after 100 minutes of training, $W_+ = 0.24$, $W_-/W_+ = 1.022$). No additional inhibitory input during learning was used for this experiment.

Table 1: Comparison of Learning Performance Between the Usual STDP Rule ("Basic") and the Modification ("Modified") Suggested by Froemke & Dan (2002).

STDP Rule	Maximum Input Correlation	Spike Correlation	Angular Error (°)	Weight Deviation (%)
Basic	0.8	0.83 ± 0.08	6.8 ± 4.7	4.25 ± 2.2
Modified	0.8	0.73 ± 0.09	17.2 ± 6.1	8.4 ± 3.8
Basic	0.54	0.83 ± 0.11	4.5 ± 1.5	3.2 ± 0.6
Modified	0.54	0.91 ± 0.05	3.7 ± 1.6	2 ± 0.6
Basic	0	0.84 ± 0.08	3.2 ± 1.3	1.9 ± 0.4
Modified	0	0.9 ± 0.07	2.7 ± 2.4	0.93 ± 0.6

Notes: The last three columns show how well randomly drawn target transformations F were learned in each case. The first two lines report learning results achieved for the same input distribution as in experiment 3, with nine groups of inputs where the correlation within group i is 0.1 $(i - 1)$. Lines 3 and 4 report results for inputs with slightly weaker correlations ($0.07 \cdot (i - 1)$ in group i, $i = 1, \ldots, 9$). The last two lines report results for uncorrelated inputs. Training time was 60 minutes for the basic update and 90 for the modified update, with 20 repetitions for different target transformations F and different initial parameters. Learning parameters used for the modified update rule were $W_+ = 1.34, 1.34, 0.59$, and $W_- = 0.66, 0.625, 0.265$ for a maximum correlation of 0.8, 0.54, 0 respectively.

6 Variations of STDP Rules for Modulation of Weights

6.1 Learning Rule for Spike Trains Suggested by Froemke and Dan. In modeling studies for STDP, one usually applies the STDP rule uniformly to all pairs of pre- and postsynaptic spikes. In one recent experimental study (Froemke & Dan, 2002), plasticity was induced not by repeated parings of isolated pre- and postsynaptic spikes, but by longer pre- and postsynaptic spike trains of a type as they occur in vivo. It was found that a correction term to the STDP rule that weakens the impact of pre- and postsynaptic spikes that occur shortly after another spike within the same neuron (see appendix A) fits these experimental data better. We examined the impact of this modified STDP rule on teacher-induced learning and found that it somewhat reduces the learning accuracy in the presence of highly correlated inputs, but has no or even a slightly positive effect for other input distributions (see Table 1).

6.2 Learning Intermediate Values of Weights. The STDP rule, equation 3.2, avoids the growth of weights beyond bounds 0 and $w_{\max}$ by simple clipping. Alternatively one can also make the weight update dependent on the actual weight value, as discussed in section 4.2. With the update rule given in equation 4.7, intermediate values of weights between 0 and $w_{\max}$ become stable (as long as the input distribution does not change). However, Gütig et al. (2003) showed that this effect is highly sensitive with regard to the

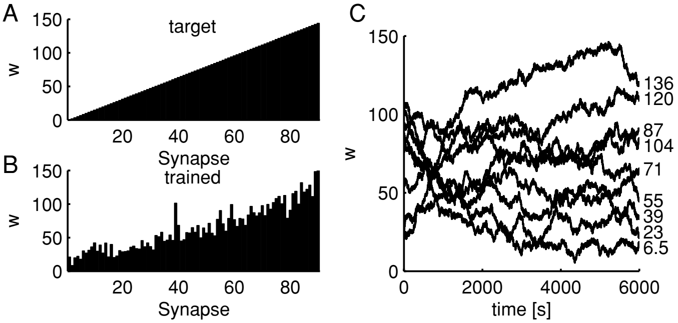

Figure 7: Learning with a multiplicative variation of STDP that is able to produce stable intermediate weight values. (A) Weight vector $\mathbf{w}^*$ of the target transformation. (B) Learned weight vector $\mathbf{w}(t)$ after 100 minutes of training. (C) Temporal evolution of weights during training (each weight can vary between 0 and $w_{\max} = 216$). The numbers on the right-hand side give the values of these weights that were used to generate the target transformation F.

values of μ_+ and μ_- and that these parameters require different values for different input distributions. In a parameter regime where stable intermediate weight values can be produced by STDP, more target transformations F from input spike trains to output spike trains can be implemented by a neuron in a stable manner, and hence can potentially be learned.

Our computer simulations show that this is in fact the case (at least for uncorrelated Poisson inputs). A typical learning result is shown in Figure 7, for a target transformation F with intermediate weights between 0 and 144 for 90 excitatory synapses, as shown in Figure 7A ($w_{\max} = 216$). The temporal evolution of nine selected weights during learning is shown in Figure 7C, and the resulting weight vector $\mathbf{w}(t)$ after 100 minutes of learning in Figure 7B. In 20 trials of 100 minutes duration (each with different initial weights drawn from a uniform distribution over [0,108], and 100 uncorrelated Poisson input spike trains at 20 Hz), a spike correlation of 0.77 ± 0.01, angular error of 20.2 ± 0.07 degrees, and a weight deviation of $8.3 \pm 0.07\%$ was reached. In this experiment, learning parameters were $W_+ = 0.12$, $W_-/W_+ = 1.03$, $\mu_+ = 0.01$, and $\mu_- = 0.03$. Results are highly sensitive to these parameters.

7 Modulation of Initial Release Probabilities by STDP

Experimental data from slice (Markram & Tsodyks, 1996) suggest that synaptic plasticity may not change the uniform scaling of the amplitudes of EPSPs resulting from a presynaptic spike train (i.e., the parameter w), but rather redistribute the sum of their amplitudes in a different way to individual EPSPs. If one assumes that STDP changes the parameter U that determines the synaptic release probability[12] for the first spike in a spike train, whereas the weight w remains unchanged (see the synapse model discussed in section 2), then the same experimental data that support rule 2.2 for STDP support the following rule for changing U:

$$U_{new} = \begin{cases} \min\{U_{\max}, U_{old} + U_+ \cdot e^{-\Delta t/\tau_+}\}, & \text{if } \Delta t > 0 \\ \max\{0, U_{old} - U_- \cdot e^{\Delta t/\tau_-}\}, & \text{if } \Delta t \le 0, \end{cases} \tag{7.1}$$

with suitable nonnegative parameters $U_{\max}, U_+, U_-, \tau_+, \tau_-$.

One can easily prove that the class of transformations F that a neuron can implement for different vectors $\mathbf{U}$ of initial release probabilities (with generic values of $\mathbf{w}$) is a different one from the class of transformations it can implement for different vectors $\mathbf{w}$. Hence, not only the learning rule changes from equation 3.2 to equation 7.1, but also the class of potential targets F for learning changes. Analogously as before, we first assigned to each excitatory synapse a value $U_{\max}$ drawn from a gaussian distribution with mean 0.25 and SD 0.02 (bounded by 0.25 ± 3 SD), as well as a value w drawn from a gamma distribution with mean 12 and standard deviation 8.4. The synaptic parameters D and F were chosen from gaussian distributions with mean 0.7, 0.021. The SD of each parameter was chosen to be 10% of its mean (with negative values replaced by values from a uniform distribution between zero and two times the mean). Then target transformations F for learning were constructed by randomly choosing for each excitatory synapse either 0 or $U_{\max}$ as the value for U (with randomly drawn w-values from a gamma distribution with mean 12 and SD 8.4). Figure 8A compares a typical target spike train used in this section to a typical target spike train used in section 5. It shows that transformations F used here typically produce other output spike trains than the corresponding assignment of values $w_{\min} = 0$ and $w_{\max}$ to these synapses (with U-values chosen as described in section 2: drawn from a gaussian distribution with mean 0.5 and SD 0.05; $w_{\max}$ randomly chosen as in experiment 1). Subsequently learning according to rule 7.1 was started with teacher-induced pulses according to F and initial values of U randomly chosen from a uniform distribution in the interval [0, 0.1] (30 extra uncorrelated inhibitory inputs were added during training as in experiment 3 to reduce undesired firing). Figure 8 shows results of repeating experiment 1 (which was for uncorrelated Poisson inputs) in this new setting. Twenty repetitions of this experiment (with different random choices of learning targets F and different initial conditions) yielded after 42 minutes of training the following results: spike correlation 0.88 ± 0.036, angular error 27.9 ± 3.7 degrees, $U-$ deviation 14.6 ± 2.6%, for $U_+ = 0.0012$,

[12] If one assumes that neurons are connected by a sufficiently large number of synaptic release sites, release probability can be approximated in a deterministic model by the amplitude of EPSPs.

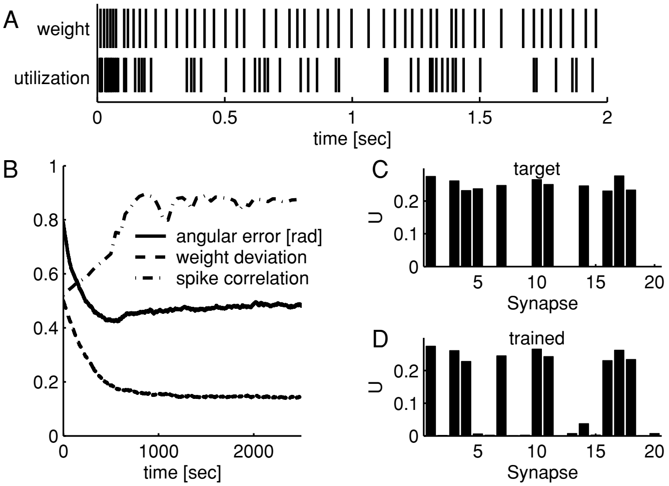

Figure 8: Results of modulation of initial release probabilities according to STDP. (A) To demonstrate typical differences between target transformations F resulting from synapse-specific values of U rather than synapse-specific values of w, we plotted the output of two such transformations F for the same input (100 uncorrelated Poisson spike trains at 20 Hz). For the upper trace, F_w was constructed by random assignments of minimal or maximal values of w to individual synapses. For the lower trace, F_U was constructed by choosing $U_{\max}$ ($U_{\min} = 0$) for a synapse whenever $w_{\max}$ ($w_{\min} = 0$) was chosen for the same synapse in the construction of F_w. (B) Performance of U-learning, analogous to Figure 3B for experiment 1. (C, D) Same plots as in Figures 3C, and 3D but for values of U (rather than w) of the target transformation F and after training (with randomly chosen initial values).

$U_-/U_+ = 1.055$. The spike correlation produced by 20 U vectors drawn from the same distribution as the target U vector (which corresponds to some baseline value of correlation) was 0.55 ± 0.09 (mean ± SD). Again, achieved spike correlations are far above chance level.

We also repeated experiment 3 with correlated Poisson inputs (more precisely, 9 groups of 10 inputs with correlation $0.1 \cdot (i - 1)$ among the inputs in group i) for the setting of U-learning. A typical result is plotted in Figure 9. Although the deviation between the vector $\mathbf{U}^*$ that was used to generate F and the last vector $\mathbf{U}(t)$ (after 35 minutes of training) is rather large (see Figures 9B and 9C), the output spike train produced by the trained neuron matches that produced for the same input by the target transformation F quite well (see Figure 9A). Apparently the output spike train is less sensitive to changes in the values of U than to changes in w. This was confirmed by testing spike correlations between output spike trains produced

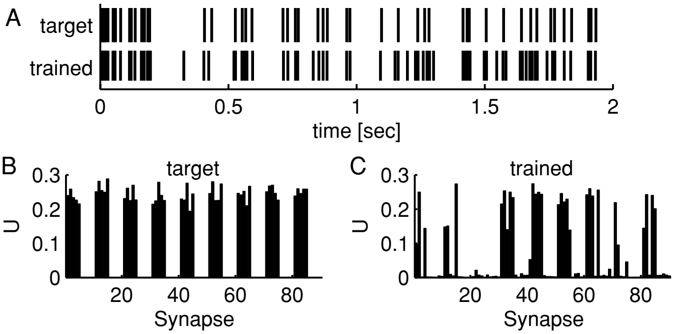

Figure 9: U-learning with correlated inputs (same input as in experiment 3; see Figure 5). (A) Typically a good fit is achieved between the output produced by the target transformation F ("target") and the output ("trained") produced by a neuron whose U-values were modulated according to STDP rule 7.1. (B) Vector $\mathbf{U}^*$ used to generate the target F. (C) Vector $\mathbf{U}(t)$ produced after 35 minutes of training.

by random U vectors and output spike trains produced by the target U vector. Such U vectors, drawn from the same distribution as the target U vectors, already achieved a spike correlation of 0.69 ± 0.6 (mean ±SD, 20 trials). Consequently, since only the "behavior of F" but not the vector $\mathbf{U}^*$ is made available to the neuron during training, the resulting correlation between target and actual output spike trains is quite high, whereas angular error between $\mathbf{U}^*$ and $\mathbf{U}(t)$, as well as the average deviation in U, remain rather large. This fact is supported by 20 repetitions of this experiment with different targets F and different initial conditions, which yielded after 35 minutes of training the following results: spike correlation 0.75 ± 0.08, angular error 39.3 ± 4.8 degrees, $U-$ deviation $25.9 \pm 4.9\%$, for $U_+ = 8 \cdot 10^{-4}$, $U_-/U_+ = 1.09$.

These positive results for U-learning with STDP are somewhat surprising, since increasing U for a synapse from a neuron that fired at some time t_1 shortly before a desired firing at time t_2 of the postsynaptic neuron in general does not increase the probability that the postsynaptic neuron will fire on its own at time t_2 if the same input spike trains would be repeated. The reason is that the presynaptic spike at time t_1 may be preceded by other spikes from the same presynaptic neuron, so that an increase of the initial release probability U of the corresponding synapse is likely to deplete synaptic resources at a faster rate and may actually result in an EPSP of a smaller amplitude in response to the presynaptic spike at time t_1. The positive results for U-learning with STDP reported in this section point to a possible benefit of relatively small values of the initial release probability U, since in this case, the previously described adverse scenario is less likely to occur for realistic presynaptic firing rates (in our simulations, U was not allowed to grow beyond a randomly chosen value $U_{\max}$ that had a mean of 0.25; increasing $U_{\max}$ reduced the effectivity of U-learning with STDP).

8 Discussion

We have examined in this letter the question, "What can a spiking neuron learn with STDP?" The answer at which we have arrived is that a spiking neuron can learn with STDP basically any map F from input to output spike trains that it could possibly implement in a stable manner. This holds at least for uncorrelated and correlated Poisson input spike trains. In other words, the spiking neurons convergence conjecture (SNCC) for STDP is approximately satisfied for such inputs in an average case sense. One could interpret this as saying that STDP endows spiking neurons with universal learning capabilities for Poisson inputs (since no neuron could possibly learn a transformation that it cannot implement with any setting of its adjustable parameters). In particular, STDP enables spiking neurons to learn to predict even very complex temporal patterns of input currents that are provided to the neuron during training.

On the other hand, we have shown that this result is quite sensitive to the distribution of inputs for which learning takes place, since we showed in section 3 that the SNCC for STDP is provably false for some worst-case input scenarios. We have highlighted in section 3 three structural differences between the perceptron learning rule for McCulloch-Pitts neurons and STDP for spiking neurons. One of these differences (failure of common rules for STDP to discourage firing for inputs for which firing is not desired) was used in section 3 to explain why the perceptron learning rule is guaranteed by the perceptron convergence theorem to converge from arbitrary initial values to an error-free solution (provided that such solution exists), whereas no corresponding guarantee can be given for STDP in the worst case. On the other hand, our theoretical average case analysis in section 4 and our computer simulations have shown that Poisson input spike trains provide a sufficiently rich set of positive examples (i.e., of input spike patterns for which the neuron is supposed to fire in order to approximate a given target map F from input spike trains to output spike trains) so that a lack of adjustment of parameters in response to negative examples is less severe. But our theoretical analysis shows that convergence of learning with STDP requires a proper choice of the relationship between the parameters W_+ and W_- (see, e.g., equation 3.1) which determine the balance between long-term synaptic facilitation and long-term synaptic depression in STDP. In the alternative interpretation of STDP where the initial release probability U is adjusted (see section 7), a suitable balance of the parameters U_+ and U_- in equation 7.1 is needed for convergence of learning. Nevertheless, our results suggest that it would be quite important to study more systematically the changes of synaptic parameters resulting from presynaptic spikes that do not cause postsynaptic spikes. Some evidence for the existence of such biological mechanisms has been provided in the previously cited work by Yves Frégnac and his collaborators, as well as in Markram et al. (1997).

It had already been shown in previous modeling studies (see, e.g., Kempter et al., 1999; Song et al., 2000) that STDP enables the most dominating one among several input sources to control the output of a neuron. In one sense, this result is closely related to the experimental results by Frégnac et al. and to the modeling results of this article, since the extra input currents induced by a "teacher" represent the dominant input source. But there is one essential difference: the control of the output spike train by the dominant input source is achieved in the latter two cases not by strengthening the synapses from this dominant input source; in fact this dominant input source disappears after training, and the neuron still fires at times when the dominant input source would have been very high.

We have shown in section 4 that a mathematical average case analysis can be carried out for supervised learning with STDP. This theoretical analysis also supports (under some simplifying assumptions) the validity of the SNCC for Poisson inputs. In addition, this theoretical analysis produces the first criterion that allows us to predict whether supervised learning with STDP will succeed (or equivalently, whether a weight vector is stable under STDP) in spite of correlations among Poisson input spike trains. For the special case of "sharp correlations" (i.e., when the cross correlations can be approximated by a δ-function), this criterion can be formulated in terms of linear separability of the rows of the correlation matrix for the input, and its mathematical form is therefore reminiscent of the well-known condition for learnability in the case of perceptron learning. In this sense, corollary 1 can be viewed as an analogon of the perceptron convergence theorem for spiking neurons with STDP. Our computer simulations show that the analytically derived criteria predict quite well whether STDP converges for correlated Poisson input spike trains even for the case of more realistic models of neurons and synapses and for the case where a number of simplifying statistical assumptions regarding the input statistics are not satisfied.

In contrast to previous modeling studies for STDP, we have based all computer simulations discussed in this article on biologically realistic models for dynamic synapses. Furthermore, we have shown in section 7 that an alternative interpretation of STDP where one assumes that it modulates the initial release probabilities U of dynamic synapses, rather than their scaling factors w, gives rise to very satisfactory convergence results for learning. This alternative interpretation of STDP is strongly suggested by data from experiments where the effect of STDP was tested with more than a single test spike (Markram & Tsodyks, 1996), but its possible impact on learning has so far been studied very little. The simulation results for modulations of initial release probabilities U by STDP (with relatively small values of U) are surprisingly positive if one takes into account that an increase of U has a quite different impact on the amplitude of an EPSP caused by a spike within a longer spike train than a corresponding increase of the synaptic efficacy w. Those positive learning results may point to functional benefits of small release probabilities for synapses that are relevant for precise timing of firing in neural circuits.

Appendix A: Details to Computer Simulations

A.1 Neuron Parameters. Membrane time constant $\tau_m = 30$ ms, absolute refractory period $T_{refract} = 3$ ms, resting potential $V_{resting} = 0$ V, reset voltage $V_{reset} = 14.2$ mV, membrane resistance $R_m = 1\,\mathrm{M}\Omega$, constant background current $I_{background}$ randomly chosen for each trial from the interval [13.5 nA, 14.5 nA]. Threshold voltage was set such that each neuron spiked at a rate of about 25 Hz. This resulted in threshold voltages slightly above 15 mV.

A.2 Synaptic Parameters. The synaptic current $x(t)$ of a synapse is increased by $A_k \cdot \frac{q}{\tau_S}$ each time a presynaptic spike arrives, with $x(0) = 0$. Here, $q = 3pC$ ($q = 6pC$) is the total charge that is injected into the postsynaptic neuron by the excitatory (inhibitory) synapse by a single spike with amplitude $A = 1$. Otherwise, the synaptic current decreases exponentially, $\tau_S \frac{dx}{dt} = -x$ with $\tau_S = 3$ ms ($\tau_S = 6$ ms) for excitatory (inhibitory) synapses (see Gerstner & Kistler, 2002).

A.3 Correlated Spike Trains. To produce n spike trains with correlation factor cc and frequency f, we proceeded, as in Gütig et al. (2003), with a time bin of size $\Delta t = 0.2$ ms bins. We constructed a Poisson spike train S_r with frequency f by assigning a spike to each bin with probability $f\,\Delta t$. The spike train S_r was used as a template for the construction of the input spike trains. Let $\theta = f\,\Delta t(1 - \sqrt{cc}) + \sqrt{cc}$ and $\phi = f\,\Delta t(1 - \sqrt{cc})$. Each input spike train was generated by assigning a spike to a bin not in S_r with probability ϕ and assigning a spike to a bin in S_r with probability θ (see Gütig et al., 2003). To model an exponential decay with time constant τ_{cc} in the cross-correlation function, we added timing jitter drawn from a Laplacian distribution with time constant $\tau_{cc}/2$ to all spikes in these spike trains.

To generate correlated rates in experiment 5, we used an algorithm that has been introduced in Song et al. (2000). The rates of two different inputs i and j with correlation parameters c_i and c_j have the cross-correlation function $\langle r_i(t) r_j(t')\rangle_t = \bar{r}^2(1 + c_i c_j \exp(-|t - t'|/\tau_c))$, where $\langle\,\rangle_t$ represents an average over the ensemble of rates, and the average firing rate $\bar{r}$ is chosen to be 20 Hz. The cross-correlation function of the rate of a given input is $\langle r(t) r(t')\rangle_t = \bar{r}^2(1 + \exp(-|t - t'|/\tau_c))$. To generate such rates for n inputs, we chose intervals of time from an exponential distribution with mean interval τ_c. For every interval, we generated $n + 1$ random numbers, y and x_a for $a = 1, 2, \ldots, n$, from gaussian distributions with zero mean and standard deviation one and σ_a respectively, where $\sigma_a^2 = 1 - c_a^2$. At the start of each interval, the firing rate for input a was set to $r_a = \bar{r}(1 + x_a + c_a y)$ and held at this value until the start of the next interval.

A.4 Modified Synaptic Update Rule. The modified update rule used in section 6 was suggested in Froemke and Dan (2002) and assigns to each pre- and postsynaptic spike an efficacy that depends on the time difference to the preceding spike in the same neuron. The efficacy of the ith spike is given by $\epsilon_i = 1 - \exp(-(t_i - t_{i-1})/\tau_s)$, where t_i and t_{i-1} are the timings of the ith and $(i-1)$th spike, respectively, and τ_s is the suppression time constant. The actual change in the amplitude of the EPSP for prespike i and postspike j is $\epsilon_i^{pre} \cdot \epsilon_j^{post} \cdot \Delta A$, where ΔA is given by equation 2.2 for $\Delta t = t_j^{post} - t_i^{pre}$. The contributions of different spike pairs were combined additively. The parameters were chosen as in Froemke & Dan (2002): $\tau_s^{pre} = 28$ ms, $\tau_s^{post} = 88$ ms, $\tau_+ = 14.8$ ms, $\tau_- = 33.8$ ms.

Appendix B: Details to the Counterexample in Section 3

The two panels of Figure 2 denote two different input scenarios with input spike trains $\langle S_1, S_2, S_3\rangle$: one where the neuron is supposed to fire at time t_3 (A) and one where the neuron is not supposed to fire at all (B). The maximal weight $w_{\max}$ of the three synapses (which is here assumed to be the same for all three synapses) should be scaled in such a way that a single spike cannot bring the neuron to its firing threshold, but two spikes at time t_2 with synaptic weights $w_{\max}$ will make it fire at time t_3 in the scenario of Figure 2A and a spike at time t_1' with weight $w_{\max}/4$ together with a spike at t_2' with weight $w_{\max}$ will make it fire at time t_3' in the scenario of Figure 2B (but no single spike on its own). Furthermore the second spike of S_2 in scenario A should be timed in such a way that postsynaptic firing at time t_3 cannot cause an increase of w_2 (because $W_+ \cdot e^{-(t_3-t_1)/\tau_+} = W_- \cdot e^{(t_3-t_4)/\tau_-}$ in rule 3.2). Then initial values $w_1 = w_3 = w_{\max}$ and $w_2 = 0$ provide a solution to both constraints of Figure 2, which is stable with regard to STDP. But if learning starts, for example, with initial values $w_1 = w_3 = w_{\max}$ and $w_2 = w_{\max}/4$, then the neuron will fire initially in both scenarios A and B. Furthermore, no application of STDP for any sequence of scenarios A and/or B (even with teacher-induced firing at time t_3 in scenario A or even with teacher-induced prevention of firing in scenario B) can decrease any of the weights. Learning with STDP also fails if one starts with small initial weights (e.g., $w_1 = w_3 = 0, w_2 = w_{\max}/4$) and teacher-induced hyperpolarization prevents all undesired firing (i.e., all firing except at time t_3 in scenario A). If sufficiently many instances of scenario A occur during training (in addition to an arbitrary number of scenarios B), then learning will in this case also converge to $w_1 = w_3 = w_{\max}$ and $w_2 = w_{\max}/4$, so that the neuron will also fire in scenario B. Hence, learning with STDP does not converge from these initial weights to a solution of this learning problem, although a stable solution exists. Note that such counterexamples can be constructed for any given positive values of the parameters W_+, W_-.

This counterexample shows that no convergence theorem can exist for STDP that holds, like the perceptron convergence theorem, for any given set of inputs. But this counterexample does not yet demonstrate failure of convergence of STDP for realistic conditions with noise, since the assumption $W_+ \cdot e^{-(t_3-t_1)/\tau_+} = W_- \cdot e^{-(t_3-t_4)/\tau_-}$ will no longer remain valid if there is jitter on the firing times.

Appendix C: A Simple Result on Linear Separability (Needed for the Proof of Proposition 1)

Consider the vectors $\mathbf{c}_1, \ldots, \mathbf{c}_m \in \mathbb{R}^n$ where $\mathbf{c}_i = (c_{i1}, \ldots, c_{in})$ and labels $y_1, \ldots, y_m \in \{0, 1\}$. Furthermore consider vectors $\mathbf{c}'_1, \ldots, \mathbf{c}'_m \in \mathbb{R}^n$ where $\mathbf{c}'_i = (c'_{i1}, \ldots, c'_{in})$ with $c'_{ij} = a + bc_{ij}$ for arbitrary constants $a \in \mathbb{R}$ and $b > 0$. We show that a vector $\mathbf{w} \in \mathbb{R}^n$ linearly separates the list $\langle\langle \mathbf{c}_1, y_1\rangle, \ldots, \langle \mathbf{c}_m, y_m\rangle\rangle$ if and only if $\mathbf{w} \in \mathbb{R}^n$ linearly separates the list $\langle\langle \mathbf{c}'_1, y_1\rangle, \ldots, \langle \mathbf{c}'_m, y_m\rangle\rangle$.

Since $c_{ij} = \frac{c'_{ij}}{b} - \frac{a}{b} = a' + b'c'_{ij}$ with $a' \in \mathbb{R}$ and $b' > 0$, we need to show only one direction. Suppose that $\mathbf{w} \in \mathbb{R}^n$ linearly separates the list $\langle\langle \mathbf{c}_1, y_1\rangle, \ldots, \langle \mathbf{c}_m, y_m\rangle\rangle$. From Definition 2, it follows that there exists a threshold $\Theta \in \mathbb{R}$ such that $y_i = sign(\sum_{j=1}^n c_{ij}w_j - \Theta)$ for $i = 1, \ldots, m$. Therefore, for the threshold $\Theta' = \Theta + \frac{a}{b}\sum_{j=1}^n w_j$, we have

$$y_i = sign\left(\sum_{j=1}^n (\frac{a}{b} + c_{ij})w_j - \Theta'\right) \text{ for all } i = 1, \ldots, m.$$

Since for every $x \in \mathbb{R}$ and $\gamma > 0$, it holds that $sign(x) = sign(\gamma x)$, we have

$$y_i = sign\left(\sum_{j=1}^n (a + bc_{ij})w_j - b\,\Theta'\right) \text{ for all } i = 1, \ldots, m.$$

Hence, there exists a threshold Θ'' such that

$$y_i = sign\left(\sum_{j=1}^n c'_{ij}w_j - \Theta''\right) \text{ for all } i = 1, \ldots, m.$$

This shows that $\mathbf{w} \in \mathbb{R}^n$ linearly separates the list $\langle\langle \mathbf{c}'_1, y_1\rangle, \ldots, \langle \mathbf{c}'_m, y_m\rangle\rangle$.

Acknowledgments

We thank Yves Frégnac, Wulfram Gerstner, and especially Henry Markram for inspiring discussions and two anonymous referees for helpful comments. The work was partially supported by the Austrian Science Fund FWF, project P15386, and PASCAL, project IST2002-506778, of the European Union.

References

Abbott, L. F., & Nelson, S. B. (2000). Synaptic plasticity: Taming the beast. *Nature Neurosci.*, *3*, 1178–1183.

Amit, D. J., Wong, K. Y. M., & Campell, C. (1989). Perceptron learning with sign-constrained weights. *J. Phys. A: Math. Gen.*, *22*, 2039–2045.

Debanne, D., Shulz, D. E., & Frégnac, Y. (1998). Activity dependent regulation of on- and off-responses in cat visual cortical receptive fields. *Journal of Physiology*, *508*, 523–548.

Duda, R. O., Hart, P. E., & Storck, D. G. (2001). *Pattern classification* (2nd ed.). New York: Wiley.

Frégnac, Y. (2002). Hebbian synaptic plasticity. In M. A. Arbib (Ed.), *The handbook of brain theory and neural networks* (pp. 515–522). Cambridge, MA: MIT Press.

Frégnac, Y., & Shulz, D. E. (1999). Activity-dependent regulation of receptive field properties of cat area 17 by supervised Hebbian learning. *Journal of Neurobiology*, *41*(1), 69–82.

Frégnac, Y., Shulz, D., Thorpe, S., & Bienenstock, E. (1988). A cellular analogue of visual cortical plasticity. *Nature*, *333*(6171), 367–370.

Frégnac, Y., Shulz, D., Thorpe, S., & Bienenstock, E. (1992). Cellular analogs of visual cortical epigenesis. I. Plasticity of orientation selectivity. *J. Neurosci.*, *12*(4), 1280–1300.

Froemke, R. C., & Dan, Y. (2002). Spike-timing-dependent synaptic modification induced by natural spike trains. *Nature*, *415*, 433–438.

Gerstner, W., & Kistler, W. M. (2002). *Spiking neuron models*. Cambridge: Cambridge University Press.

Gerstner, W., Ritz, R., & van Hemmen, J. L. (1993). Why spikes? Hebbian learning and retrieval of time-resolved excitation patterns. *Biological Cybernetics*, *69*, 503–515.

Gütig, R., Aharonov, R., Rotter, S., & Sompolinsky, H. (2003). Learning input correlations through non-linear temporally asymmetric Hebbian plasticity. *Journal of Neuroscience*, *23*, 3697–3714.

Haykin, S. (1999). *Neural networks: A comprehensive foundation* (2nd ed.). Upper Saddle River: Prentice Hall.

Herrmann, A., & Gerstner, W. (2001). Noise and the PSTH response to current transients: I. General theory and application to the integrate-and-fire neuron. *J. Comp. Neurosci.*, *11*(2), 135–151.

Kempter, R., Gerstner, W., & van Hemmen, J. L. (1999). Hebbian learning and spiking neurons. *Phys. Rev. E*, *59*(4), 4498–4514.

Kempter, R., Gerstner, W., & van Hemmen, J. L. (2001). Intrinsic stabilization of output rates by spike-based Hebbian learning. *Neural Computation*, *13*, 2709–2741.

Kistler, W. M., & van Hemmen, J. L. (2000). Modeling synaptic plasticity in conjunction with the timing of pre- and postsynaptic action potentials. *Neural Computation*, *12*, 385–405.

Legenstein, R. A., & Maass, W. (2004). *Additional material to the paper: What can a neuron learn with spike-timing-dependent plasticity?* (Tech. Rep.). Graz: Institute for Theoretical Computer Science, Graz University of Technology.

Maass, W., & Markram, H. (2002). Synapses as dynamic memory buffers. *Neural Networks*, *15*, 155–161.

Markram, H., Lübke, J., Frotscher, M., & Sakmann, B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science*, *275*, 213–215.

Markram, H., & Tsodyks, M. (1996). Redistribution of synaptic efficacy between neocortical pyramidal neurons. *Nature*, *382*, 807–810.

Markram, H., Wang, Y., & Tsodyks, M. (1998). Differential signaling via the same axon of neocortical pyramidal neurons. *PNAS*, *95*, 5323–5328.

Mehta, M. R. (2001). Neuronal dynamics of predictive coding. *Neuroscientist*, *7*, 490–495.

Rao, R. P. N., & Sejnowski, T. J. (2002). Predictive coding, cortical feedback, and spike-timing dependent plasticity. In R. P. N. Rao, B. A. Olshauser, & M. S. Lewicki, (Eds.), *Probabilistic models of the brain* (pp. 297–315). Cambridge, MA: MIT Press.

Rosenblatt, J. F. (1962). *Principles of neurodynamics*. New York: Spartan Books.

Rubin, J., Lee, D., & Sompolinsky, H. (2001). Equilibrium properties of temporal asymmetric Hebbian plasticity. *Physical Review Letters*, *86*, 364–367.

Senn, W., & Fusi, S. (in press). Learning only when necessary: Better memories of correlated patterns in networks with bounded synapses. *Neural Computation*.

Senn, W., Schneider, M., & Ruf, B. (2002). Activity-dependent selection of axonal and dendritic delays or, why synaptic transmission should be unreliable. *Neural Computation*, *14*(3), 503–619.

Song, S., Miller, K. D., & Abbott, L. F. (2000). Competitive Hebbian learning through spike-timing dependent synaptic plasticity. *Nature Neuroscience*, *3*, 919–926.

van Rossum, M. C. W., Bi, G., & Turrigiano, G. G. (2000). Stable Hebbian learning through spike-timing-dependent plasticity. *Journal of Neuroscience*, *20*, 8812–8821.

Received April 26, 2004; accepted March 22, 2005.