LETTER Communicated by Wulfram Gerstner

Learning Real-World Stimuli in a Neural Network with Spike-Driven Synaptic Dynamics

Joseph M. Brader
brader@cns.unibe.ch
Walter Senn
senn@pyl.unibe.ch
Institute of Physiology, University of Bern, Bern, Switzerland

Stefano Fusi
fusi@ini.unizh.ch
Institute of Physiology, University of Bern, Bern, Switzerland, and Institute of Neuroinformatics, ETH|UNI Zurich, 8059, Zurich, Switzerland

We present a model of spike-driven synaptic plasticity inspired by experimental observations and motivated by the desire to build an electronic hardware device that can learn to classify complex stimuli in a semisupervised fashion. During training, patterns of activity are sequentially imposed on the input neurons, and an additional instructor signal drives the output neurons toward the desired activity. The network is made of integrate-and-fire neurons with constant leak and a floor. The synapses are bistable, and they are modified by the arrival of presynaptic spikes. The sign of the change is determined by both the depolarization and the state of a variable that integrates the postsynaptic action potentials. Following the training phase, the instructor signal is removed, and the output neurons are driven purely by the activity of the input neurons weighted by the plastic synapses. In the absence of stimulation, the synapses preserve their internal state indefinitely. Memories are also very robust to the disruptive action of spontaneous activity. A network of 2000 input neurons is shown to be able to classify correctly a large number (thousands) of highly overlapping patterns (300 classes of preprocessed Latex characters, 30 patterns per class, and a subset of the NIST characters data set) and to generalize with performances that are better than or comparable to those of artificial neural networks. Finally we show that the synaptic dynamics is compatible with many of the experimental observations on the induction of long-term modifications (spike-timing-dependent plasticity and its dependence on both the postsynaptic depolarization and the frequency of pre- and postsynaptic neurons).

1 Introduction

Many recent studies of spike-driven synaptic plasticity have focused on using biophysical models to reproduce experimental data on the induction of long-term changes in single synapses (Senn, Markram, & Tsodyks, 2001; Abarbanel, Huerta, & Rabinovich, 2002; Shouval, Bear, & Cooper, 2002; Karmarkar & Buonomano, 2002; Saudargiene, Porr, & Wörgötter, 2004; Shouval & Kalantzis, 2005). The regulatory properties of synaptic plasticity based on spike timing (STDP) have been studied both in recurrent neural networks and at the level of single synapses (see Abbott & Nelson, 2000; Kempter, Gerstner, & van Hemmen, 2001; Rubin, Lee, & Sompolinsky, 2001; Burkitt, Meffin, & Grayden, 2004), in which the authors study the equilibrium distribution of the synaptic weights. These are only two of the many aspects that characterize the problem of memory encoding, consolidation, maintenance, and retrieval. In general these different aspects have been studied separately, and the computational implications have been largely neglected despite the fact that protocols to induce long-term synaptic changes based on spike timing were initially considered in the computational context of temporal coding (Gerstner, Kempter, van Hemmen, & Wagner, 1996).

More recently, spike-driven synaptic dynamics has been linked to several in vivo phenomena. For example, spike-driven plasticity has been shown to improve the performance of a visual perceptual task (Adini, Sagi, & Tsodyks, 2002), to be a good candidate mechanism for the emergence of direction-selective simple cells (Buchs & Senn, 2002; Senn & Buchs, 2003), and to shape the orientation tuning of cells in the visual cortex (Yao, Shen, & Dan, 2004). In Yao et al. (2004) and Adini et al. (2002), spike-driven models are proposed that make predictions in agreement with experimental results. Only in Buchs and Senn (2002) and Senn and Buchs (2003) did the authors consider the important problem of memory maintenance.

Other work has focused on the computational aspects of synaptic plasticity but neglects the problem of memory storage. Rao and Sejnowski (2001), for example, encode simple temporal sequences. In Legenstein, Naeger, and Maass (2005), spike-timing-dependent plasticity is used to learn an arbitrary synaptic configuration by imposing to the input and the output neurons the appropriate temporal pattern of spikes. In Gutig and Sompolinsky (2006) the principles of the perceptron learning rule are applied to the classification of temporal patterns of spikes. Notable exceptions are Hopfield and Brody (2004), in which the authors consider a self-repairing dynamic synapse, and Fusi, Annunziato, Badoni, Salamon, and Amit (2000), Giudice and Mattia (2001), Amit and Mongillo (2003), Giudice, Fusi, and Mattia (2003), and Mongillo, Curti, Romani, and Amit (2005) in which discrete plastic synapses are used to learn random uncorrelated patterns of mean firing rates as attractors of the neural dynamics. However, a limitation of all these studies is that the patterns stored by the network remain relatively simple.

Here we propose a model of spike-driven synaptic plasticity that can learn to classify complex patterns in a semisupervised fashion. The memory is robust against the passage of time, the spontaneous activity, and the presentation of other patterns. We address the fundamental problems of memory: its formation and its maintenance.

1.1 Memory Retention. New experiences continuously generate new memories that would eventually saturate the storage capacity. The interference between memories can provoke the blackout catastrophe that would prevent the network from recalling any of the previously stored memories (Hopfield, 1982; Amit, 1989). At the same time, no new experience could be stored. Instead, the main limitation on the storage capacity does not come from interference if the synapses are realistic (i.e., they do not have an arbitrarily large number of states) and hence allow only a limited amount of information to be stored in each synapse.

When subject to these constraints old, memories are forgotten (palimpsest property—Parisi, 1986; Nadal, Toulouse, Changeux, & Dehaene, 1986). In particular, the memory trace decays in a natural way as the oldest memories are replaced by more recent experiences. This is particularly relevant for any realistic model of synaptic plasticity that allows only a limited amount of information to be stored in each synapse. The variables characterizing a realistic synaptic dynamics are bounded and do not allow for long-term modifications that are arbitrarily small. When subject to these constraints, the memory trace decays exponentially fast (Amit & Fusi, 1992, 1994; Fusi, 2002), at a rate that depends on the fraction of synapses modified by every experience: fast learning inevitably leads to fast forgetting of past memories and results in uneven distribution of memory resources among the stimuli (the most recent experiences are better remembered than old ones). This result is very general and does not depend on the number of stable states of each synaptic variable or on the specific synaptic dynamics (Fusi, 2002; Fusi & Abbott, 2007). Slowing the learning process allows the maximal storage capacity to be recovered for the special case of uncorrelated random patterns (Amit & Fusi, 1994). The price to be paid is that all the memories should be experienced several times to produce a detectable mnemonic trace (Brunel, Carusi, & Fusi, 1998). The brain seems to be willing to pay this price in some cortical areas like inferotemporal cortex (see Giudice et al., 2003, for a review).

The next question is how to implement a mechanism that slows learning in an unbiased way. We assume that we are dealing with realistic synapses, so it is not possible to reduce the size of the synaptic modifications induced by each stimulus to arbitrarily small values. Fortunately each neuron sees a large number of synapses, and memory retrieval depends on only the total synaptic input. If only a small fraction of synaptic modifications is consolidated, then the change of the total synaptic input can be much smaller than $N\Delta J$, where N is the total number of synapses to be changed and ΔJ

is the minimal synaptic change. Randomly selecting the synaptic changes that are consolidated provides a simple, local, unbiased mechanism to slow learning (Tsodyks, 1990; Amit & Fusi, 1992, 1994). Such a mechanism requires an independent stochastic process for each synapse, and depending on the outcome of this process, the synaptic change is either consolidated or cancelled. The irregularity of the neural activity provides a natural source of randomness that can be exploited by the synapse (Fusi et al., 2000; Chicca & Fusi, 2001; Fusi, 2003). In this letter, we employ this approach and study a synapse that is bistable on long timescales. The bistability protects memories against the modifications induced by ongoing spontaneous activity and provides a simple way to implement the required stochastic selection mechanism. Not only is there accumulating evidence that biological single synaptic contacts undergo all-or-none modifications (Petersen, Malenka, Nicoll, & Hopfield, 1998; Wang, O'Connor, & Wittenberg, 2004), but additional synaptic states do not significantly improve the memory performance (Amit & Fusi, 1994; Fusi, 2002; Fusi & Abbott, 2007).

1.2 Memory Encoding. The second important issue is related to the way each synapse is modified to allow the network to recall a specific memory at a later time. Here we deal with supervised learning: each stimulus to be memorized imposes a characteristic pattern of activity on the input neurons, and an "instructor" generates an extra synaptic current that steers the activity of the output neurons in a desired direction. Note that the activity of the output neurons is not entirely determined by the instructor because the input neurons also contribute to determining the output (semisupervised learning). The aim of learning is to modify the synaptic connections between the input and the output neurons so that the output neurons respond as desired in both the presence and absence of the instructor. This problem can be solved with the perceptron learning rule (Rosenblatt, 1958) or with algorithms such as backpropagation (see Hertz, Krogh, & Palmer, 1991). Here we focus on a more complex and biologically inspired spike-driven synaptic dynamics that implements a learning algorithm similar to that of the perceptron. In the past, similar implementations of the synaptic dynamics have been successfully applied to learn nonoverlapping binary patterns (Giudice & Mattia, 2001; Amit & Mongillo, 2003; Giudice et al., 2003) and random uncorrelated binary patterns with a constant number of active neurons (Mongillo et al., 2005). In all these works, the authors were aiming to make the activity patterns imposed during training into stable attractors of the recurrent network dynamics. Here we consider a feedforward network, but the synaptic dynamics we develop could just as well be used in a recurrent network (see section 6.5 for more detail).

We show that in order to store more complex patterns with no restriction on the correlations or number of active neurons, the long-term synaptic dynamics should slow down when the response of the output neurons is in agreement with the one generated by the total current of the input

neurons. This is an indication that the currently presented pattern has already been learned and that it is not necessary to change the synapses further (stop-learning condition, as in the case of the perceptron learning rule: Rosenblatt, 1958; Block, 1962; Minsky & Papert, 1969). When a single output neuron is considered, arbitrary linearly separable patterns can be learned without errors (Senn & Fusi, 2005a, 2005b; Fusi & Senn, 2006) also in the extreme case of binary synapses. If more than one output neuron is read, then nonlinearly separable patterns can also be classified, which is not possible with a simple perceptron. We consider a network with multiple output neurons, realistic spike-driven dynamics implementing the stop-learning condition, and binary synapses on long timescales.

2 Abstract Learning Rule

We first describe the abstract learning rule: the schematic prescription according to which the synapses should be modified at each stimulus presentation. We then introduce the detailed spike-driven synaptic dynamics that implements this prescription in an approximate fashion.

The stochastic selection and stop-learning mechanisms we require are incorporated into a simple learning rule. We consider a single output neuron receiving a total current h, which is the weighted sum of the activities s_i of the N input neurons:

$$h = \frac{1}{N}\sum_{j=1}^{N}(J_j - g_I)s_j, \tag{2.1}$$

where the J_j are the binary plastic excitatory synaptic weights (where $J_j = 0, 1$), and g_I is a constant representing the contribution of an inhibitory population. The latter can be regarded as a group of cells uniformly connected to the input neurons and projecting their inhibitory afferents to the output neuron. Following each stimulus, the synapses are updated according to the following rule: if the instructor determines that the postsynaptic output neuron should be active and the total input h is smaller than some threshold value θ, then the efficacy of each synapse, J_j, is set equal to unity with a probability $q_+ s_j$, proportional to the presynaptic activity s_j. In general, this activity is a continuous variable, $s_j \in]0, 1[$. In this letter, we employ the binary activities $s_j = 0, 1$. On the contrary, if the output neuron should be inactive and h is larger than θ, then the synapse is depressed with probability $q_- s_j$. The threshold θ determines whether the output neuron is active in the absence of the instructor. The synapses are thus modified only when the output produced by the weighted sum of equation 2.1 is unsatisfactory, that is, it is not in agreement with the output desired by the instructor. With this prescription, learning would stop as soon as the output is satisfactory. However, in practice, it is useful to introduce a margin δ

and stop potentiation only when $h > \theta + \delta$. Analogously, depression would stop only when $h < \theta - \delta$. The margin δ guarantees a better generalization (see section 5.6). The learning rule can be summarized as follows:

$$\begin{aligned} J_i &\to 1 \text{ with probability } q_+ s_i \quad \text{if } h_i < \theta + \delta \text{ and } \xi = 1 \\ J_i &\to 0 \text{ with probability } q_- s_i \quad \text{if } h_i > \theta - \delta \text{ and } \xi = 0, \end{aligned} \tag{2.2}$$

where ξ is a binary variable indicating the desired output as specified by the instructor and the right arrow indicates how the synapse is updated.

This learning prescription allows us to learn linearly separable patterns in a finite number of iterations (Senn & Fusi, 2005a, 2005b) provided that (1) g_I is between the minimal and the maximal excitatory weights ($g_I \in]0, 1[$), (2) N is large enough, and (3) θ and δ and $q_\pm$ are small enough.

3 The Synaptic Model

The design of the synaptic dynamics has been largely dictated by the need to implement in hardware the abstract learning rule described in the previous section. The components we have selected are directly or indirectly related to some known properties of biological synapses. They are combined to produce a spike-driven synaptic dynamics that implements the desired learning rule and, at the same time, is compatible with the experimental protocols used to induce long-term modifications.

3.1 Memory Consolidation. The first aspect we consider is related to memory preservation against the effects of both spontaneous activity and the presentation of other stimuli. We assume that each synaptic update is triggered by the arrival of a presynaptic spike. However, in order to preserve existing memories, not all of these events will eventually lead to a long-term modification of the synapse. If many of these events change the synapse in the same direction and their effect accumulates, then the consolidation process might be activated. In such a case, the synapse is modified permanently, or at least until the next stimulation. Otherwise the synaptic efficacy preceding the stimulation would be restored. In the first case, a transition to a new stable state occurred. The activation of the consolidation process depends on the specific train of presynaptic spikes and on the coincidence with other events (e.g., elevated postsynaptic depolarization). Many presynaptic trains can share the same rate (which in our case encodes the stimulus), but they can produce different outcomes in terms of consolidation. In particular, if the presynaptic spikes arrive at random times, then consolidation is activated with some probability (Fusi et al., 2000; Fusi, 2003). This allows implementing the stochastic selection that chooses only a small fraction of the synapses to be changed on each stimulus presentation. Notice that the synaptic dynamics can be completely deterministic (as in

our model) and that the stochasticity of the selection is generated by the irregularity of the pre- and postsynaptic activities.

3.2 Memory Encoding. The main goal of our synaptic model is to encode patterns of mean firing rates. In order to guide our choice of model, we incorporate elements that have a counterpart in neurophysiological experiments on pairs of connected neurons. Specific experimental aspects we choose to consider are:

- **1a:** Spike-timing-dependent plasticity (STDP). If a presynaptic spike precedes a postsynaptic action potential within a given temporal window, the synapse is potentiated, and the modification is stable on long timescales in the absence of other stimulations (memory is consolidated). If the phase relation is reversed, the synapse is depressed. This behavior has been observed in vitro (Markram, Lübke, Frotscher, & Sakmann, 1997; Feldman, 2000; Sjöström, Turrigiano, & Nelson, 2001), with realistic spike trains (Froemke & Dan, 2002; Sjöström et al., 2001), and in vivo (Zhang, Tao, Holt, Harris, & Poo, 1998; Zhou, Tao, & Poo, 2003) for mean pre- and postsynaptic frequencies between 5 and 20 Hz.
- **1b:** Dependence on postsynaptic depolarization. If the STDP protocol is applied to obtain LTP but the postsynaptic neuron is hyperpolarized, the synapse remains unaltered, or it slightly depresses (Sjöström et al., 2001). More generally, the postsynaptic neuron needs to be sufficiently depolarized for LTP to occur.
- **1c:** LTP dominance at high frequencies. When both pre- and postsynaptic neurons fire at elevated frequencies, LTP always dominates LTD regardless of the phase relation between the pre- and the postsynaptic spikes (Sjöström et al., 2001).

The corresponding dynamic elements we include in our model are:

- **2a:** STDP. Two dynamical variables are needed to measure the time passed since the last pre- and postsynaptic spikes. They would be updated on the arrival or generation of a spike and then decay on the typical timescale of the temporal window of STDP (order of 10 ms). Other dynamical variables acting on longer timescales would be needed to restrict the STDP behavior in the frequency range of 5 to 20 Hz.
- **2b:** Dependence on postsynaptic depolarization. A direct reading of the depolarization is sufficient. Notice that the postsynaptic depolarization can be used to encode the instantaneous firing rate of the postsynaptic neuron (Fusi et al., 2000; Fusi, 2001): the average subthreshold depolarization of the neuron is a monotonic function of the mean firing rate of the postsynaptic neurons, in both simple

models of integrate-and-fire neurons (Fusi et al., 2000) and experiments (Sjöström et al., 2001).

2c: LTP dominance at high frequencies. We assume that a relatively slow variable acting on a timescale of 100 ms (internal calcium concentration is a good candidate—Abarbanel et al., 2002; Shouval et al., 2002) will measure the mean postsynaptic frequency. For high values of this variable, LTP should dominate, and aspects related to spike timing should be disregarded.

Ingredients 2a to 2c are sufficient to implement the abstract learning rule but without the desired stop-learning condition; that is, the condition that if the frequency of the postsynaptic neuron is too low or too high, no long-term modification should be induced. This additional regulatory mechanism could be introduced through the incorporation of a new variable or by harnessing one of the existing variables. A natural candidate to implement this mechanism is calcium concentration, ingredient 2c, as the average depolarization is not a sufficiently sensitive function of postsynaptic frequency to be exploitable (Fusi, 2003).

3.3 Model Reduction. We now introduce the minimal model that reproduces all the necessary features and implements the abstract rule. STDP can be achieved using a combination of depolarization dependence and an effective neuronal model, as in Fusi et al. (2000) and Fusi (2003). When a presynaptic spike shortly precedes a postsynaptic action potential, it is likely that the depolarization of an integrate-and-fire neuron is high, resulting in LTP. If the presynaptic spike comes shortly after the postsynaptic action potential, the postsynaptic integrate-and-fire neuron is likely to be recovering from the reset following spike emission, and it is likely to be hyperpolarized, resulting in LTD. This behavior depends on the neuronal model. In this work, we employ simple linear integrate-and-fire neurons with a constant leak and a floor (Fusi & Mattia, 1999),

$$\frac{dV}{dt} = -\lambda + I(t) \tag{3.1}$$

where λ is a positive constant and $I(t)$ is the total synaptic current. When a threshold V_θ is crossed, a spike is emitted, and the depolarization is reset to $V = H$. If at any time V becomes negative, then it is immediately reset to $V = 0$. This model can reproduce quantitatively the response function of pyramidal cells measured in experiments (Rauch, La Camera, Lüscher, Senn, & Fusi, 2003). The adoption of this neuronal model, in addition to the considerations on the temporal relations between pre- and postsynaptic spikes, allows us to reproduce STDP (point 1a of the above list) with only one dynamic variable (the depolarization of the postsynaptic neuron $V(t)$), provided that we accept modifications in the absence of postsynaptic action

potentials. Given that we never have silent neurons in realistic conditions, the last restriction should not affect the network behavior much.

These considerations allow us to eliminate the two dynamic variables of point 2a and to model the synapse with only one dynamic variable $X(t)$, which is modified on the basis of the postsynaptic depolarization $V(t)$ (ingredient 2b) and the postsynaptic calcium variable $C(t)$. We emphasize that we consider an effective model in which we attempt to subsume the description into as few parameters as possible. It is clear that different mechanisms are involved in the real biological synapses and neurons.

We now specify the details of the synaptic dynamics. The synapses are bistable with efficacies J_+ (potentiated) and J_- (depressed). Note that the efficacies J_+ and J_- can now be any two real numbers and are no longer restricted to the binary values (0,1) as in the case of the abstract learning rule. The internal state of the synapse is represented by $X(t)$, and the efficacy of the synapse is determined according to whether $X(t)$ lies above or below a threshold θ_X. The calcium variable $C(t)$ is an auxiliary variable with a long time constant and is a function of postsynaptic spiking activity,

$$\frac{dC(t)}{dt} = -\frac{1}{\tau_C}C(t) + J_C \sum_i \delta(t - t_i), \tag{3.2}$$

where the sum is over postsynaptic spikes arriving at times t_i. J_C is the contribution of a single postsynaptic spike, and τ_C is the time constant (La Camera, Rauch, Lüscher, Senn, & Fusi, 2004).

The variable $X(t)$ is restricted to the interval $0 \le X \le X_{\max}$ (in this work, we take $X_{\max} = 1$) and is a function of $C(t)$ and of both pre- and postsynaptic activity. A presynaptic spike arriving at t_{pre} reads the instantaneous values $V(t_{\text{pre}})$ and $C(t_{\text{pre}})$. The conditions for a change in X depend on these instantaneous values in the following way,

$$\begin{aligned} X &\to X + a \quad \text{if} \quad V(t_{\text{pre}}) > \theta_V \quad \text{and} \quad \theta^l_{\text{up}} < C(t_{\text{pre}}) < \theta^h_{\text{up}} \\ X &\to X - b \quad \text{if} \quad V(t_{\text{pre}}) \le \theta_V \quad \text{and} \quad \theta^l_{\text{down}} < C(t_{\text{pre}}) < \theta^h_{\text{down}}, \end{aligned} \tag{3.3}$$

where a and b are jump sizes, θ_V is a voltage threshold ($\theta_V < V_\theta$), and the θ^l_{up}, θ^h_{up}, θ^l_{down} and θ^h_{down} are thresholds on the calcium variable (see Figure 2a). In the absence of a presynaptic spike or if the conditions 3.3 are not satisfied, then $X(t)$ drifts toward one of two stable values,

$$\begin{aligned} \frac{dX}{dt} &= \alpha \quad \text{if} \quad X > \theta_X \\ \frac{dX}{dt} &= -\beta \quad \text{if} \quad X \le \theta_X, \end{aligned} \tag{3.4}$$

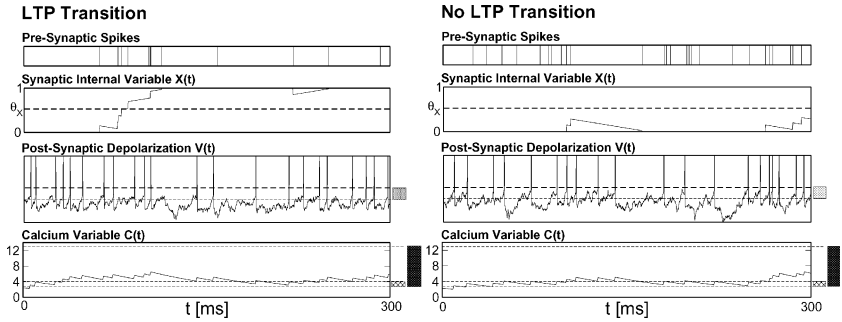

Figure 1: Stochastic synaptic transitions. (Left) A realization for which the accumulation of jumps causes X to cross the threshold θ_X (an LTP transition). (Right) A second realization in which the jumps are not consolidated and thus give no synaptic transition. The shaded bars correspond to thresholds on $C(t)$; see equation 3.3. In both cases illustrated here, the presynaptic neuron fires at a mean rate of 50 Hz, while the postsynaptic neuron fires at a rate of 40 Hz.

where α and β are positive constants and θ_X is a threshold on the internal variable. If at any point during the time course $X < 0$ or $X > 1$, then X is held at the respective boundary value. The efficacy of the synapse is determined by the value of the internal variable at t_{pre}. If $X(t_{\text{pre}}) > \theta_X$, the synapse has efficacy J_+, and if $X(t_{\text{pre}}) \leq \theta_X$, the synapse has efficacy J_-. In Figure 1 we show software simulation results for two realizations of the neural and synaptic dynamics during a 300 ms stimulation period. In both cases, the input is a Poisson train with a mean rate of 50 Hz and a mean output rate of 40 Hz. Due to the stochastic nature of pre- and postsynaptic spiking activity, one realization displays a sequence of jumps that consolidate to produce an LTP transition; X(t) crosses the threshold θ_X, whereas the other does not cross θ_X and thus gives no transition. All parameter values are in Table 1.

4 Single Synapse Behavior

4.1 Probability of Potentiating and Depressing Events. The stochastic process $X(t)$ determining the internal state of the synapse is formed by an accumulation of jumps that occur on arrival of presynaptic spikes. Before analyzing the full transition probabilities of the synapse, we present results for the probabilities that, given a presynaptic spike, X experiences either an upward or downward jump. In Figure 2a, we show the steady-state probability density function of the calcium variable $C(t)$ for different postsynaptic frequencies. At low values of the postsynaptic firing rate ν_{post}, the dynamics are dominated by the decay of the calcium concentration, resulting in a pile-up of the distribution about the origin. For larger values, $\nu_{\text{post}} > 40$ Hz, the distribution is well approximated by a gaussian. The shaded bars in

Table 1: Parameter Values Used in the Spike-Driven Network Simulations.

Neural Parameters		Teacher Population		Calcium Parameters		Inhibitory Population		Synaptic Parameters		Input Layer	
λ	$10\ V_\theta$/s	N_{ex}	20	τ_C	60 ms	N_{inh}	1000	a	$0.1\ X_{\max}$	N_{input}	2000
θ_V	$0.8\ V_\theta$	ν_{ex}	50 Hz	θ^l_{up}	$3\ J_C$	ν_{inh}	$50f$ Hz	b	$0.1\ X_{\max}$	$\nu_{stimulated}$	50 Hz
				θ^l_{down}	$3\ J_C$	J_{inh}	$-0.035\ V_\theta$	θ_x	$0.5\ X_{\max}$	$\nu_{unstimulated}$	2 Hz
				θ^h_{down}	$4\ J_C$			α	$3.5\ X_{\max}$/s	J_+	J_{ex}
				θ^h_{up}	$13\ J_C$			β	$3.5\ X_{\max}$/s	J_-	0

Notes: The parameters V_θ, $X_{\max}$, and J_C set the scale for the depolarization, synaptic internal variable and calcium variable, respectively. All three are set equal to unity. The firing frequency of the inhibitory pool is proportional to the coding level f of the presented stimulus. The teacher population projects to the output neurons, which should be active in response to the stimulus.

Figure 1 indicate the thresholds for upward and downward jumps in $X(t)$ and correspond to the bars shown alongside $C(t)$. The probability of an upward or downward jump is thus given by the product of the probabilities that both $C(t_{\text{pre}})$ and $V(t_{\text{pre}})$ fall within the defined ranges. Figure 2b shows the jump probabilities P_{up} and P_{down} as a function of ν_{post}.

4.2 Probability of Long-Term Modifications. In order to calculate the probability of long-term modification, we repeatedly simulate the time evolution of a single synapse for given pre- and postsynaptic rate. The stimulation period used is $T_{\text{stim}} = 300$ ms, and we ran $N_{\text{trials}} = 10^6$ trials for each $(\nu_{\text{pre}}, \nu_{\text{post}})$ pair to ensure good statistics. In order to calculate the probability of an LTP event, we initially set $X(0) = 0$ and simulated a realization of the time course $X(t)$. If $X(T_{\text{stim}}) > \theta_X$ at the end of the stimulation period, we registered an LTP transition. An analogous procedure was followed to calculate the LTD transition probability with the exception that the initial condition $X(0) = 1$ was used and a transition was registered if $X(T_{\text{stim}}) < \theta_X$. The postsynaptic depolarization was generated by Poisson trains from additional excitatory and inhibitory populations. It is known that a given mean firing rate does not uniquely define the mean μ and variance σ^2 of the subthreshold depolarization, and although not strongly sensitive, the transition probabilities do depend on the particular path in (μ, σ^2) space. We choose the linear path $\sigma^2 = 0.015\mu + 2$, which yields the same statistics as the full network simulations considered later and thus ensures that the transition probabilities shown in Figure 2 provide an accurate guide. The linearity of the relations between σ^2 and μ comes from the fact that for a Poisson train of input spikes emitted at a rate ν, both σ^2 and μ are linear functions of ν, and the coefficients are known functions of the connectivity and the average synaptic weights (see Fusi, 2003).

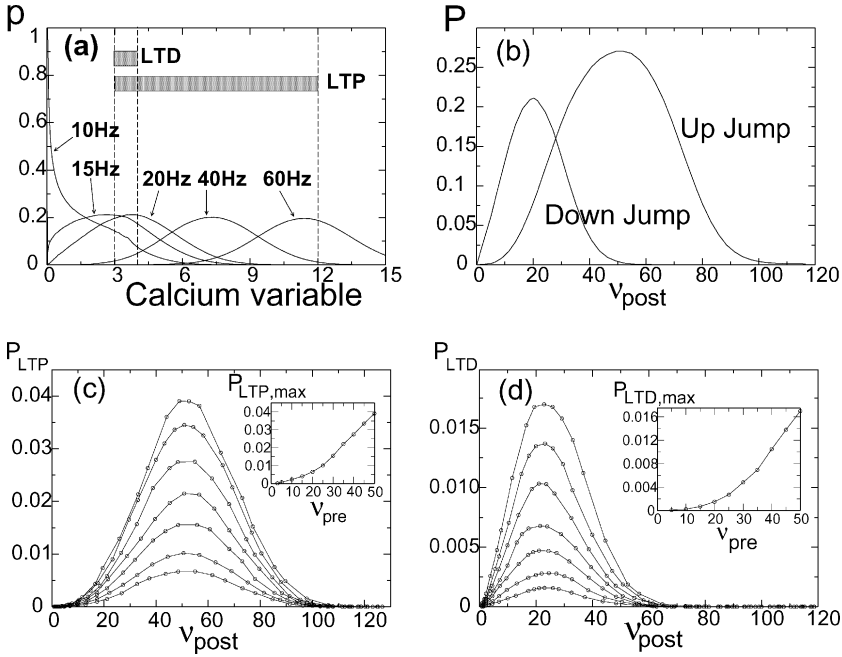

Figure 2: (a) Probability density function of the internal calcium variable for different values of ν_{post}. The bars indicate regions for which upward and downward jumps in the synaptic internal variable, X, are allowed, labeled LTP and LTD, respectively. See equation 3.3. (b) Probability for an upward or downward jump in X as a function of postsynaptic frequency. (c, d) LTP and LTD transition probabilities, respectively, as a function of ν_{post} for different values of ν_{pre}. The insets show the height of the peak in the transition probabilities as a function of the presynaptic frequency.

In Figure 2 we show the transition probabilities as a function of ν_{post} for different ν_{pre} values. There exists a strong correspondence between the jump probabilities and the synaptic transition probabilities of Figure 2. The consolidation mechanism yields a nonlinear relationship between the two sets of curves. The model synapse displays Hebbian behavior: LTP dominates at high ν_{post}, and LTD dominates at low ν_{post} when the presynaptic neuron is stimulated. When the presynaptic neuron is not stimulated, the transition probabilities become so small that the synapse remains unchanged. The decay of the transition probabilities at high and low values of ν_{post} implements the stop-learning condition and is a consequence of the thresholds on $C(t)$. The inset to Figure 2 shows the peak height of the LTP and LTD probabilities as a function of ν_{pre}. For $\nu_{\text{pre}} > 25$ Hz the maximal transition probability shows a linear dependence upon ν_{pre}. This important feature

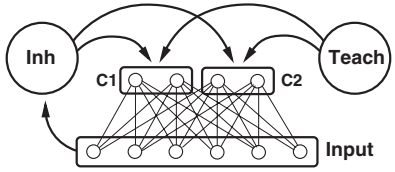

Figure 3: A schematic of the network architecture for the special case of a data set consisting of two classes. The output units are grouped into two pools, selective to stimuli C1 and C2, respectively, and are connected to the input layer by plastic synapses. The output units receive additional inputs from teacher and inhibitory populations.

implies that use of this synapse is not restricted to networks with binary inputs, as considered in this work, but would also prove useful in networks employing continuous valued input frequencies.

A useful memory device should be capable of maintaining the stored memories in the presence of spontaneous activity. In order to assess the stability of the memory to such activity, we have performed long simulation runs, averaging over 10^7 realizations, for $\nu_{\text{pre}} = 2$ Hz and a range of ν_{post} values in order to estimate the maximal LTP and LTD transition probability. We observed no synaptic transitions during any of the trials. This provides upper bounds of $P_{\text{LTP}}(\nu_{\text{pre}} = 2\,\text{Hz}) < 10^{-7}$ and $P_{\text{LTD}}(\nu_{\text{pre}} = 2\,\text{Hz}) < 10^{-7}$. With a stimulation time of 300 ms, this result implies 0.3×10^7 seconds between synaptic transitions and provides a lower bound to the memory lifetime of approximately 1 month, under the assumption of 2 Hz spontaneous activity.

5 Network Performance

5.1 The Architecture of the Network. The network architecture we consider consists of a single feedforward layer composed of N_{inp} input neurons fully connected by plastic synapses to N_{out} outputs. Neurons in the output layer have no lateral connection and are subdivided into pools of size $N_{\text{out}}^{\text{class}}$, each selective to a particular class of stimuli. In addition to the signal from the input layer, the output neurons receive additional signals from inhibitory and teacher populations. The inhibitory population provides a signal proportional to the coding level of the stimulus and serves to balance the excitation coming from the input layer (as required in the abstract rule; see section 2). A stimulus-dependent inhibitory signal is important, as it can compensate for large variations in the coding level of the stimuli. The teacher population is active during training and imposes the selectivity of the output pools with an additional excitatory signal. A schematic view of this network architecture is shown in Figure 3.

Multiple arrays of random classifiers have been the subject of considerable interest in recent studies of machine learning and can achieve results for complex classification tasks far beyond those obtainable using a single classifier ($N_{\text{out}}^{\text{class}} = 1$) (see section 6 for more on this point).

Following learning, the response of the output neurons to a given stimulus can be analyzed by selecting a single threshold frequency to determine which neurons are considered active (express a vote). The classification result is determined using a majority rule decision between the selective pools of output neurons. (For a biologically realistic model implementing a majority rule decision, see Wang, 2002.) We distinguish among three possibilities upon presentation of a pattern: (1) correctly classified—the output neurons of the correct selective pool express more votes than the other pools; (2) misclassified—an incorrect pool of output neurons wins the vote; and (3) nonclassified—no output neuron expresses a vote and the network remains silent. Nonclassifications, are preferable to misclassifications, as a null response to a difficult stimulus retains the possibility that such cases can be sent to other networks for further processing. In most cases, the majority of the errors can be made nonclassifications with an appropriate choice of threshold. The fact that a single threshold can be used for neurons across all selective pools is a direct consequence of the stop-learning condition, which keeps the output response within bounds.

5.2 Illustration of Semisupervised Learning. To illustrate our approach to supervised learning, we apply our spike-driven network to a data set of 400 uncorrelated random patterns with low coding level ($f = 0.05$) divided equally into two classes. The initial synaptic matrix is set randomly, and an external teacher signal is applied that drives the (single) output neuron to a mean rate of 50 Hz on presentation of a stimulus of class 1 and to 6 Hz on presentation of a stimulus of class 0; the mean rates of 50 Hz and 6 Hz result from the combined input of teacher plus stimulus. The LTP and LTD transition probabilities shown in Figure 2 intersect at $\nu_{\text{post}} \sim 20$ Hz. Stimuli that elicit an initial output response more than 20 Hz on average exhibit a strengthening of this response during learning, whereas stimuli that elicit an initial response less than 20 Hz show a weakening response during learning. The decay of the transition probabilities at both high and low ν_{post} eventually arrests the drift in the output response and prevents overlearning the stimuli. Patterns are presented in random order, and by definition, the presentation number increases by 1 for every 400 patterns presented to the network. In Figure 4 we show output frequency histograms across the data set as a function of presentation number. The output responses of the two classes become more strongly separated during learning and eventually begin to pile up around 90 Hz and 0 Hz due to the decay of the transition probabilities. Figure 5 shows the output frequency distributions across the data set in the absence of a teacher signal both before and after learning. Before learning, both classes have statistically identical distributions.

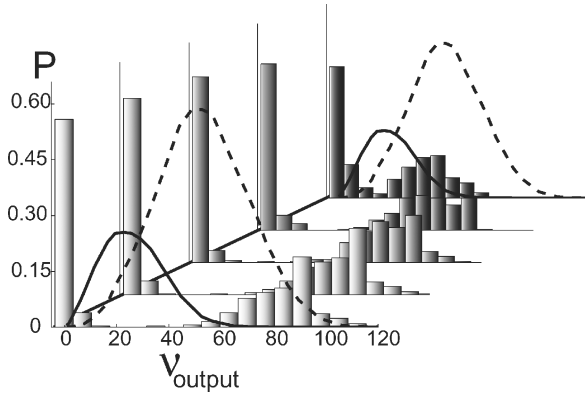

Figure 4: Frequency response histograms on presentation of the training set at different stages during learning for a simple test case consisting of a single output neuron. Throughout learning, the teacher signal is applied to enforce the correct response. The panels from back to front correspond to presentation number 1, 60, 120, 180, and 240, respectively. The solid and dashed curves correspond to the LTD and LTP transition probabilities, respectively, from Figure 2.

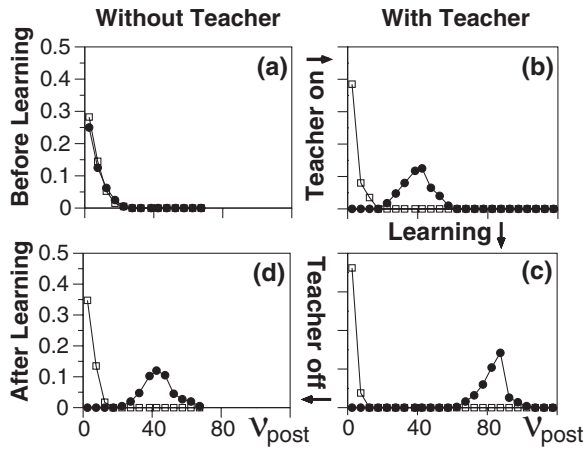

Figure 5: Frequency response histograms before and after learning. (a) Response to class 1 (filled circles) and class 0 (open squares) before learning without external teacher signal. (b) Response before learning with teacher signal. (c) Response after learning with teacher signal. (d) Situation after learning without teacher signal.

Following learning, members of class 0 have a reduced response, whereas members of class 1 have responses distributed around 45 Hz. Classification can then be made using a threshold on the output frequency.

When considering more realistic stimuli with a higher level of correlation, it becomes more difficult to achieve a good separation of the classes, and there may exist considerable overlap between the final distributions. In such cases it becomes essential to use a network with multiple output units to correctly sample the statistics of the stimuli.

5.3 The Data Sets and the Classification Problem. Real-world stimuli typically have a complex statistical structure very different from the idealized case of random patterns often considered. Such stimuli are characterized by large variability within a given class and a high level of correlation between members of different classes. We consider two separate data sets.

The first data set is a binary representation of 293 Latex characters preprocessed, as in Amit and Mascaro (2001) and presents in a simple way these generic features of complex stimuli. Each character class consists of 30 members generated by random distortion of the original character. Figure 6 shows the full set of characters. This data set has been previously studied in Amit and Geman (1997) and Amit and Mascaro (2001) and serves as a benchmark test for the performance of our network. In Amit and Mascaro (2001) a neural network was applied to the classification problem, and in Amit and Geman (1997), decision trees were employed. It should be noted that careful preprocessing of these characters is essential to obtain good classification results, and we study the same preprocessed data set used in Amit and Mascaro (2001). The feature space of each character is encoded as a 2000-element binary vector. The coding level f (the fraction of active neurons) is sparse but highly variable, spanning a range $0.01 < f < 0.04$. On presentation of a character to the network, we assign one input unit per element which is activated to fire at 50 Hz if the element is unity but remains at a spontaneous rate of 2 Hz if the element is zero. Due to random character deformations, there is a large variability within a given class, and despite the sparse nature of the stimuli, there exist large overlaps between different classes.

The second data set we consider is the MNIST data set, a subset of the larger NIST handwritten characters data set. The data set consists of 10 classes (digits $0 \rightarrow 9$) on a grid of 28×28 pixels. (The MNIST data set is available from http://yann.lecun.com, which also lists a large number of classification results.) The MNIST characters provide a good benchmark for our network performance and have been used to test numerous classification algorithms. To input the data to the network, we construct a 784-element vector from the pixel map and assign a single input neuron to each element. As each pixel has a grayscale value, we normalize each element such that the largest element has value unity. From the full MNIST data set,

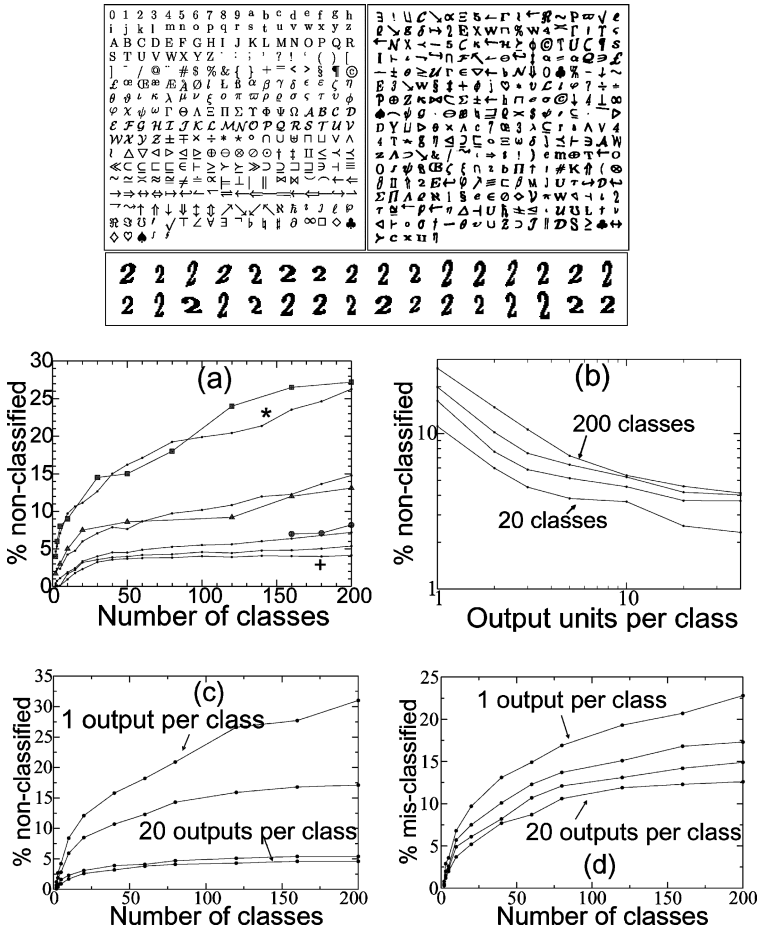

Figure 6: The full Latex data set containing 293 classes. (a) Percentage of non-classified patterns in the training set as a function of the number of classes for different numbers of output units per class. Results are shown for 1(∗), 2, 5, 10, and 40(+) outputs per class using the abstract rule (points) and for 1, 2, and 5 outputs per class using the spike-driven network (squares, triangles, and circles, respectively). In all cases, the percentage of misclassified patterns is less than 0.1%. (b) Percentage of nonclassified patterns as a function of the number of output units per class for different numbers of classes (abstract rule). Note the logarithmic scale. (c) Percentage of nonclassified patterns as a function of number of classes for generalization on a test set (abstract rule). (d) Same as for *c* but showing percentage of misclassified patterns.

we randomly select 20,000 examples for training and 10,000 examples for testing.

The Latex data set is more convenient for investigating the trends in behavior due to the large number of classes available. The MNIST data set is complementary to this; although there are only 10 classes, the large number of examples available for both training and testing makes possible a more meaningful assessment of the asymptotic (large number of output neurons) performance of the network relative to existing results for this data set. For both data sets the high level of correlation between members of different classes makes this a demanding classification task.

We now consider the application of our network to these data sets. We first present classification results on a training set obtained from simulations employing the full spike-driven network. As such simulations are computationally demanding, we supplement the results obtained from the spike-driven network with simulations performed using the abstract learning rule in order to explore more fully the trends in performance. We then perform an analysis of the stability of the spike-driven network results with respect to parameter fluctuations, an issue of practical importance when considering hardware VLSI implementation. Finally, we consider the generalization ability of the network.

5.4 Spike-Driven Network Performance. The parameters entering the neural and synaptic dynamics as well as details of the inhibitory and teacher populations can be found in Table 1. In our spike-driven simulations, the inhibitory pool sends Poisson spike trains to all output neurons at a mean rate proportional to the coding level of the presented stimulus (note that each output neuron receives an independent realization at the same mean rate). The teacher population is purely excitatory and sends additional Poisson trains to the output neurons, which should be selective to the present stimuli. It now remains to set a value for the vote expression threshold. As a simple method to set this parameter, we performed preliminary simulations for the special case $N_{\text{out}}^{\text{class}} = 1$ and adjusted the output neuron threshold to obtain the best possible performance on the full data set. The optimal choice minimizes the percentage of nonclassified patterns without allowing significant misclassification errors. This value was then used in all subsequent simulations with $N_{\text{out}}^{\text{class}} > 1$. In Figure 6a we show the percentage of nonclassified patterns as a function of number of classes for networks with different values of $N_{\text{out}}^{\text{class}}$. In order to suppress fluctuations, each data point is the average over several random subsets taken from the full data set.

Given the simplicity of the network architecture, the performance on the training set is remarkable. For $N_{\text{out}}^{\text{class}} = 1$ the percentage of nonclassified patterns increases rapidly with the number of classes; however, as $N_{\text{out}}^{\text{class}}$ is increased, the performance rapidly improves, and the network eventually enters a regime in which the percentage of nonclassified patterns remains

almost constant with increasing class number. In order to test the extent of this scaling, we performed a single simulation with 20 output units per class and the full 293-class data set. In this case we find 5.5% nonclassified (0.1% misclassified), confirming that the almost constant scaling continues across the entire data set. We can speculate that if the number of classes were further increased, the system would eventually enter a new regime in which the synaptic matrix becomes overloaded and errors increase more rapidly. The total error of 5.6% that we incur using the full data set with $N_{\text{out}}^{\text{class}} = 20$ should be contrasted with that of (Amit and Mascaro, 2001), who reported an error of 39.8% on the 293 class problem using a single network with 6000 output units, which is roughly equivalent to our network with $N_{\text{out}}^{\text{class}} = 20$. By sending all incorrectly classified patterns to subsequent networks for reanalysis ("boosting"; see section 8.6), Amit and Mascaro obtained 5.4% error on the training set using 15 boosting cycles. This compares favorably with our result. We emphasize that we use only a single network, and essentially all of our errors are nonclassifications. Figure 6b shows the percentage of nonclassified patterns as a function of the number of output units per class for fixed class number.

In order to evaluate the performance for the MNIST data set, we retain all the parameter values used for the Latex experiments. Although it is likely that the results on the MNIST data set could be optimized using specially tuned parameters, the fact that the same parameter set works adequately for both MNIST and Latex cases is a tribute to the robustness of our network (see section 5.7 for more on this issue). We find that near-optimal performance on the training set is achieved for $N_{\text{out}}^{\text{class}} = 15$, for which we obtain 2.9% nonclassifications and 0.3% misclassifications.

5.5 Abstract Rule Performance. Due to the computational demands of simulating the full spike-driven network, we have performed complementary simulations using the abstract rule in equation 2.2 to provide a fuller picture of the behavior of our network. The parameters are chosen such that the percentage of nonclassified patterns for the full data set with $N_{\text{out}}^{\text{class}} = 1$ matches those obtained from the spike-driven network. In Figure 6a we show the percentage of nonclassified patterns as a function of number of classes for different values of $N_{\text{out}}^{\text{class}}$. Although we have chosen the abstract rule parameters by matching only a single data point to the results from the spike-driven network, the level of agreement between the two approaches is excellent.

5.6 Generalization. We tested the generalization ability of the network by selecting randomly 20 patterns from each class for training and reserving the remaining 10 for testing. In Figures 6c and 6d, we show the percentage of mis- and nonclassified patterns in the test set as a function of the number of classes for different values of $N_{\text{out}}^{\text{class}}$. The monotonic increase in the percentage of nonclassified patterns in the test set is reminiscent

of the training set behavior but lies at a slightly higher value for a given number of classes. For large $N_{\text{out}}^{\text{class}}$, the percentage of nonclassifications exhibits the same slow increase with number of classes as seen in the training set. Although the percentage of misclassifications increases more rapidly than the nonclassifications, it also displays a regime of slow increase for $N_{\text{out}}^{\text{class}} > 20$.

When applied to the MNIST data set, the spiking network yields very reasonable generalization properties. We limit ourselves to a maximum value of $N_{\text{out}}^{\text{class}} = 15$ due to the heavy computational demand of simulating the spike-driven network. Using $N_{\text{out}}^{\text{class}} = 15$, we obtain 2.2% nonclassifications and 1.3% misclassifiactions. This compares favorably with existing results on this data set and clearly demonstrates the effectiveness of our spike-driven network. For comparison, k-nearest neighbor classifiers typically yield a total error in the range 2% to 3%, depending on the specific implementation, with a best result of 0.63% error obtained using shape context matching (Belongie, Malik, & Puzicha, 2002). (For a large number of results relating to the MNIST data set, see http://www.lecun.com.) Convolutional nets yield a total error around 1%, depending on the implementation, with a best performance of 0.4% error using cross-entropy techniques (Simard, Steinkraus, & Platt, 2003).

We have also investigated the effect of varying the parameter δ on the generalization performance using the abstract learning rule. Optimal performance on the training set is obtained for small values of δ, as this enables the network to make a more delicate distinction between highly correlated patterns. The price to be paid is a reduced generalization performance, which results from overlearning the training set. Conversely, a larger value of δ reduces performance on the training set but improves the generalization ability of the network. In the context of a recurrent attractor network, δ effectively controls the size of the basin of attraction. In general, an intermediate value of δ allows compromise between accurate learning of the test set and reasonable generalization.

5.7 Stability with Respect to Parameter Variations. When considering hardware implementations, it is important to ensure that any proposed model is robust with respect to variations in the parameter values. A material device provides numerous physical constraints, and so an essential prerequisite for hardware implementation is the absence of fine-tuning requirements. To test the stability of the spiking network, we investigate the change in classification performance with respect to perturbation of the parameter values for the special case of $N_{\text{out}}^{\text{class}} = 20$ with 50 classes.

In order to identify the most sensitive parameters, first consider the effect of independent variation. At each synapse, the parameter value is reselected from a gaussian distribution centered about the tuned value and with a standard deviation equal to 15% of that value. All other parameters are held at their tuned values. This approach approximates the natural

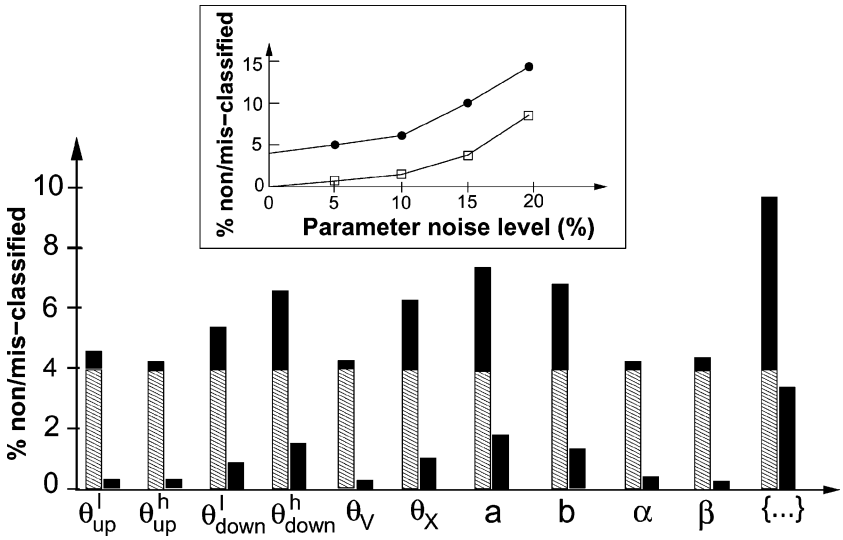

Figure 7: Stability of the network with respect to variations in the key parameters; see equations 2.2, 3.1, and 3.2. The black bars indicate the change in the percentage of nonclassified patterns, and the dark gray bar indicates the change in the percentage of misclassified patterns on random perturbation of the parameter values. With no perturbation, the network yields 4% nonclassified (light gray bars) and less than 0.1% misclassified. Simultaneous perturbation of all parameters is marked $\{\dots\}$ in the right-most column. The inset shows the change in the percentage of nonclassified (circles) and misclassified (squares) patterns as a function of the parameter noise level when all parameters are simultaneously perturbed.

variation that occurs in hardware components and can be expected to vary from synapse to synapse (Chicca, Indiveri, & Douglas, 2003). In Figure 7 we report the effect of perturbing the key parameters on the percentage of non- and misclassified patterns. The most sensitive parameters are those related to jumps in the synaptic variable $X(t)$, as these dominate the consolidation mechanism causing synaptic transitions.

To mimic a true hardware situation more closely, we also consider simultaneous perturbation of all parameters. Although the performance degrades significantly, the overall performance drop is less than might be expected from the results of independent parameter variation. It appears that there are compensation effects within the network. The inset to Figure 7 shows how the network performance changes as a function of the parameter noise level (standard deviation of the gaussian). For noise levels less than 10%, the performance is only weakly degraded.

6 Discussion

Spike-driven synaptic dynamics can implement semisupervised learning. We showed that a simple network of integrate-and-fire neurons connected by bistable synapses can learn to classify complex patterns. To our knowledge, this is the first work in which a complex classification task is solved by a network of neurons connected by biologically plausible synapses, whose dynamics is compatible with several experimental observations on long-term synaptic modifications. The network is able to acquire information from its experiences during the training phase and to preserve it against the passage of time and the presentation of a large number of other stimuli. The examples shown here are more than toy problems, which would illustrate the functioning of the network. Our network can classify correctly thousands of stimuli, with performances that are better than those of more complex, multilayer traditional neural networks.

The key to this success lies in the possibility of training different output units on different subsets or subsections of the stimuli (boosting technique—Freund & Schapire, 1999). In particular, the use of random connectivity between input and output layers would allow each output neuron to sample a different subsection of every stimulus. Previous studies (Amit & Mascaro, 2001) employed deterministic synapses and a quenched random connectivity. Here we use full connectivity but with stochastic synapses to generate the different realizations. This yields a dynamic random connectivity that changes continuously in response to incoming stimuli.

In order to gain some intuition into the network behavior, it is useful to consider an abstract space in which each pattern is represented by a point. The synaptic weights feeding into each output neuron define hyperplanes that divide the space of patterns. During learning, the hyperplanes follow stochastic trajectories through the space in order to separate (classify) the patterns. If at some point along the trajectory, the plane separates the space such that the response at the corresponding output neuron is satisfactory, then the hyperplane does not move in response to that stimulus. With a large number of output neurons, the hyperplanes can create an intricate partitioning of the space.

6.1 Biological Relevance

6.1.1 Compatibility with the Observed Phenomenology. Although the synaptic dynamics was mostly motivated by the need to learn realistic stimuli, it is interesting to note that the resulting model remains consistent with experimental findings. In order to make contact with experiments on pairs of connected neurons, we consider the application of experimentally realistic stimulation protocols to the model synapse. A typical protocol for the induction of LTP or LTD is to pair pre- and postsynaptic spikes with a given phase relation. In the simulations presented in Figure 8, we impose a

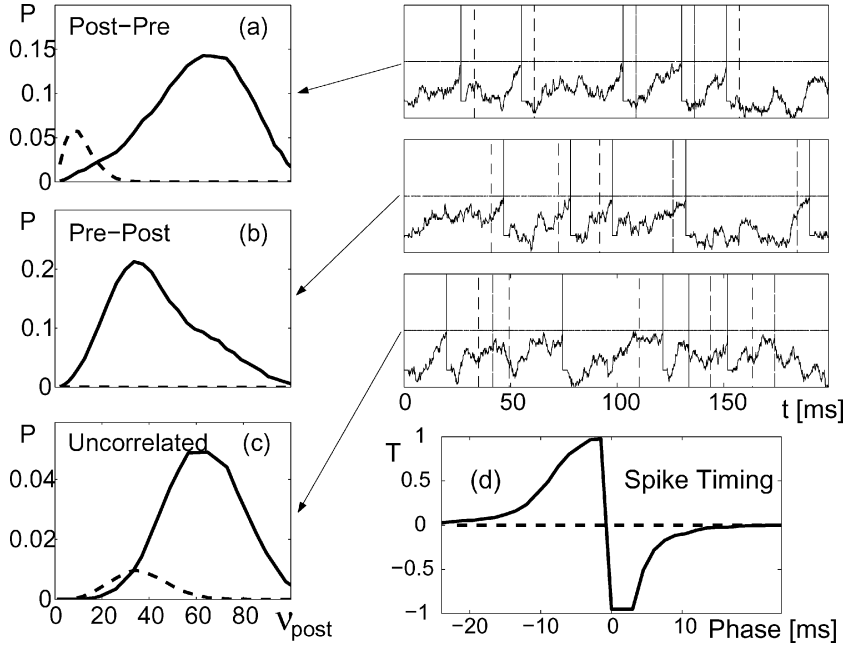

Figure 8: Synaptic transition probabilities for (a) paired post-pre, (b) paired pre-post, and (c) uncorrelated stimulation as a function of ν_{post}. The full curve is P_{LTP} and the dashed curve P_{LTD}. For the post-pre and pre-post protocols, the phase shift between pre- and postspikes is fixed at $+6$ ms and -6 ms, respectively. In all cases, $\nu_{\text{pre}} = \nu_{\text{post}}$. (d) The tendency T (see equation 6.1) as a function of the phase between pre- and postsynaptic spikes for a mean frequency $\nu_{\text{pre}} = \nu_{\text{post}} = 12$ Hz.

post-pre or pre-post pairing of spikes with a phase shift of $+6$ ms or -6 ms, respectively, and simulate the synaptic transition probabilities as a function of $\nu_{\text{post}} = \nu_{\text{pre}}$ over a stimulation period of 300 ms. The postsynaptic neuron is made to fire at the desired rate by application of a suitable teacher signal. When the prespike precedes the postspike, then LTP dominates at all frequencies; however, when the prespike falls after the postspike, there exists a frequency range ($5 < \nu_{\text{post}} < 20$) over which LTD dominates. This LTD region is primarily due to the voltage decrease following the emission of a postsynaptic spike. As the frequency increases, it becomes increasingly likely that the depolarization will recover from the postspike reset to attain a value larger than θ_V, thus favoring LTP. For completeness, we also present the LTP and LTD transition probabilities with uncorrelated pre- and postsynaptic spikes. We maintain the relation $\nu_{\text{pre}} = \nu_{\text{post}}$. We also test the spike-timing dependence of the transition probabilities by fixing $\nu_{\text{post}} = 12$ Hz and varying the phase shift between pre- and postsynaptic

spikes. In Figure 8, we plot the tendency (Fusi, 2003) as a function of phase shift. The tendency T is defined as

$$T = \left(\frac{P_{LTP}}{P_{LTD+P_{LTP}}} - \frac{1}{2} \right) \max(P_{LTP}, P_{LTD}). \tag{6.1}$$

A positive value of T implies dominance of LTP, whereas a negative value implies dominance of LTD. Although the STDP time window is rather short ($\sim$10 ms) the trend is consistent with the spike timing induction window observed in experiments.

6.1.2 Predictions. We already know that when pre- and postsynaptic frequencies are high enough, LTP is observed regardless the detailed temporal statistics (Sjöström et al., 2001). As the frequencies of pre- and postsynaptic neurons increase, we predict that the amount of LTP should progressively decrease until no long-term modification becomes possible. In the high-frequency regime, LTD can become more likely than LTP, provided that the total probability of a change decreases monotonically at a fast rate. Large LTD to compensate LTP would produce an average modification that is also small, but it would actually lead to fast forgetting as the synapses would be modified anyway. A real stop-learning condition is needed to achieve high classification performance. Although experimentalists did not study systematically what happens in the high-frequency regime, there is preliminary evidence for a nonmonotonic LTP curve (Wang & Wagner, 1999). Other regulatory mechanisms that would stop learning as soon as the neuron responds correctly might be also possible. However, the mechanism we propose is probably the best candidate in cases in which a local regulatory system is required and the instructor is not "smart" enough.

6.2 Parameter Tuning. We showed that the network is robust to heterogeneities and to parameter variations. However, the implementation of the teacher requires the tuning of one global parameter controlling the strength of the teacher signal (essentially the ratio $\nu_{ex}/\nu_{stimulated}$; see Table 1). This signal should be strong enough to steer the output activity in the desired direction and in the presence of a noisy or contradicting input. Indeed, before learning, the synaptic input is likely to be uncorrelated with the novel patterns to be learned, and the total synaptic input alone would produce a rather disordered pattern of activity of the output neurons. The teacher signal should be strong enough to dominate over this noise. At the same time, it should not be so strong that it brings the synapse into the region of very slow synaptic changes (the region above $\sim$100 Hz in Figure 4). The balance between the teacher signal and the external input is the only parameter we need to tune to make the network operate in the proper regime. In the brain, we might hypothesize the existence of other mechanisms (e.g.,

homeostasis; Turrigiano & Nelson, 2000), which would automatically create the proper balance between the synaptic inputs of the teacher (typically a top-down signal) and the synaptic inputs of the sensory stimuli.

6.3 Learning Speed. Successful learning usually requires hundreds of presentations of each stimulus. Learning must be slow to guarantee an equal distribution of the synaptic resources among all the stimuli to be stored. Faster learning would dramatically shorten the memory lifetime, making it impossible to learn a large number of patterns. This limitation is a direct consequence of the boundedness of the synapses (Amit & Fusi, 1994; Fusi, 2002; Senn & Fusi, 2005a, 2005b) and can be overcome only by introducing other internal synaptic states that would correspond to different synaptic dynamics. In particular, it has been shown that a cascade of processes, each operating with increasingly small transition probabilities, can allow for a long memory lifetime without sacrificing the amount of information acquired at each presentation (Fusi, Drew, & Abbott, 2005). In these models, when the synapse is potentiated and the conditions for LTP are met, the synapse becomes progressively more resistant to depression. These models can be easily implemented by introducing multiple levels of bistable variables, each characterized by different dynamics and each one controlling the learning rate of the previous level. For example, the jumps a_k and b_k of level k might depend on the state of level $k+1$. These models would not lead to better performance, but they would certainly guarantee a much faster convergence to a successful classification. They will be investigated in future work.

Notice also that the learning rates can be easily controlled by the statistics of the pre- and postsynaptic spikes. So far we considered a teacher signal that increases or decreases the firing rate of the output neurons. However, the higher-order statistics (e.g., the synchronization between pre- and postsynaptic neurons) can also change, and our synaptic dynamics would be rather sensitive to these alterations. Attention might modulate these statistics, and the learning rate would be immediately modified without changing any inherent parameter of the network. This idea has already been investigated in a simplified model in Chicca and Fusi (2001).

6.4 Applications. The synaptic dynamics we propose has been implemented in neuromorphic analog VLSI (very large scale integration) (Mitra, Fusi, & Indiveri, 2006; Badoni, Giulioni, Dante, & Del Giudice, 2006; Indiveri & Fusi, 2007). A similar model has been introduced in Fusi et al. (2000) and the other components required to implement massively parallel networks of integrate-and-fire neurons and synapses with the new synaptic dynamics have been previously realized in VLSI in Indiveri (2000, 2001, 2002) and Chicca and Fusi (2001). There are two main advantages of our approach. First, all the information is transmitted by events that are highly localized in time (the spikes): this leads to low power consumption and

optimal communication bandwidth usage (Douglas, Deiss, & Whatley, 1998; Boahen, 1998). Notice that each synapse in order to be updated requires the knowledge of the time of occurrence of the presynaptic spikes, its internal state, and other dynamic variables of the postsynaptic neuron (C and V). Hence each synapse needs to be physically connected to the postsynaptic neuron. The spikes from the presynaptic cell can come from other off-chip sources. Second, the memory is preserved by the bistability. In particular, it is retained indefinitely if no other presynaptic spikes arrive. All the hardware implementations require negligible power consumption to stay in one of the two stable states (Fusi et al., 2000; Indiveri, 2000, 2001, 2002; Mitra et al., 2006; Badoni, Giulioni, Dante, & Del Giudice, 2006) and the bistable circuitry does not require nonstandard technology or high voltage, as for the floating gates (Diorio, Hasler, Minch, & Mead, 1996).

Given the possibility of implementing the synapse in VLSI and the good performances obtained on the Latex data set, we believe that this synapse is a perfect candidate for low-power, compact devices with on-chip autonomous learning. Applications to the classification of real-world auditory stimuli (e.g., spoken digits) are presented in (Coath, Brader, Fusi, & Denham, 2005). They show that classification performances on classification of the letters of the alphabet are comparable to those of humans. In these applications, the input vectors are fully analog (i.e., sets of mean firing rates ranging in a given interval), showing the capability of our network to encode these kinds of patterns as well.

6.5 Multiple Layer and Recurrent Networks. The studied architecture is a single layer network. Our output neurons are essentially perceptrons, which resemble the cerebellar Purkinje cells, as already proposed by Marr (1969) and Albus (1971) (see Brunel, Hakim, Isope, Nadal, & Barbour, 2004, for a recent work). Not only they are the constituents of a single-layer network and have a large number of inputs, but they also receive a teacher signal similar to what we have in our model. However, it is natural to ask whether our synaptic dynamics can be applied also to a more complex multilayer network. If the instructor acts on all the layers, then it is likely that the same synaptic dynamics can be adopted in the multilayer case. Otherwise, if the instructor provides a bias only to the final layer, then it is unclear whether learning would converge and whether the additional synapses of the intermediate layers can be exploited to improve the performances. The absence of a theory that guarantees the convergence does not necessary imply that the learning rule would fail. Although simple counterexamples probably can be constructed for networks of a few units, it is difficult to predict what happens in more general cases.

More interesting is the case of recurrent networks in which the same neuron can be regarded as both an input and an output unit. When learning converges, then each pattern imposed by the instructor (which might be the sensory stimulus) becomes a fixed point of the network dynamics. Given the

capability of our network to generalize, it is very likely that the steady states are also stable attractors. Networks in which not all the output neurons of each class are activated would probably lead to attractors that are sparser than the sensory representations. This behavior might be an explanation of the small number of cells involved in attractors in inferotemporal cortex (see Giudice et al., 2003).

6.6 Stochastic Learning and Boosting. Boosting is an effective strategy to greatly improve classification performance by considering the "opinions" of many weak classifiers. Most of the algorithms implementing boosting start from deriving simple rules of thumb for classifying the full set of stimuli. A second classifier will concentrate more on those stimuli that were most often misclassified by the previous rules of thumb. This procedure is repeated many times, and in the end, a single classification rule is obtained by combining the opinions of all the classifiers. Each opinion is weighted by a quantity that depends on the classification performance on the training set. In our case, we have two of the necessary ingredients to implement boosting. First, each output unit can be regarded as a weak classifier that concentrates on a subset of stimuli. When the stimuli are presented, only a small fraction of randomly selected synapses changes. Our stimuli activate only tens of neurons, and the transition probabilities are small (order of 10^{-2}). The consequence is that some stimuli are actually ignored because no synaptic change is consolidated. Second, each classifier concentrates on the hardest stimuli. Indeed, only the stimuli that are misclassified induce significant changes in the synaptic structure. For the others, the transition probabilities are much smaller, and again, it is as if the stimulus is ignored. In our model, each classifier does not know how the others are performing and which stimuli are misclassified by the other output units. So the classification performances cannot be as good as in the case of boosting. However, for difficult tasks, each output neuron changes continuously its rules of thumb as the weights are stochastically updated. In general, each output neuron feels the stop-learning condition for a different weight configuration, and the aggregation of these different views often yields correct classification of the stimuli. In fact our performances are very similar to those of Amit and Mascaro (2001) when they use a boosting technique. Notice that two factors play a fundamental role: stochastic learning and a local stop-learning criterion. Indeed, each output unit should stop learning when its own output matches the one desired by the instructor, not when the stimulus is correctly classified by the majority rule. Otherwise the output units cannot concentrate on different subsets of misclassified patterns.

Acknowledgments

We thank Massimo Mascaro for many inspiring discussions and for providing the preprocessed data set of Amit and Mascaro (2001). We are grateful

to Giancarlo La Camera for careful reading of the manuscript. Giacomo Indiveri greatly contributed with many discussions in constraining the model in such a way that it could be implemented in neuromorphic hardware. Harel Shouval drew our attention to Wang and Wagner (1999). This work was supported by the EU grant ALAVLSI and partially by SNF grant PP0A-106556.

References

Abarbanel, H. D. I., Huerta, R., & Rabinovich, M. I. (2002). Dynamical model of long-term synaptic plasticity. *PNAS, 99*, 10132–10137.

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

Adini, Y., Sagi, D., & Tsodyks, M. (2002). Context enabled learning in human visual system. *Nature, 415*, 790–794.

Albus, J. (1971). A theory of cerebellar function. *Math. Biosci., 10*, 26–51.

Amit, D. (1989). *Modeling brain function*. Cambridge: Cambridge University Press.

Amit, D. J., & Fusi, S. (1992). Constraints on learning in dynamic synapses. *Network, 3*, 443.

Amit, D., & Fusi, S. (1994). Learning in neural networks with material synapses. *Neural Comput., 6*(5), 957–982.

Amit, Y., & Geman, D. (1997). Shape quantization and recognition with randomized trees. *Neural Computation, 9*, 1545–1588.

Amit, Y., & Mascaro, M. (2001). Attractor networks for shape recognition. *Neural Computation, 13*, 1415–1442.

Amit, D., & Mongillo, G. (2003). Spike-driven synaptic dynamics generating working memory states. *Neural Computation, 15*, 565–596.

Badoni, D., Giulioni, Dante, V., & Del Giudice, P. (2006). A VLSI recurrent network of spiking neurons with reconfigurable and plastic synapses. *IEEE International Symposium on Circuits and Systems ISCAS06* (pp. 1227–1230). Piscataway, NJ: IEEE.

Belongie, S., Malik, J., & Puzicha, J. (2002). Shape matching and object recognition using shape context. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(24), 509.

Block, H. (1962). The perceptron: A model for brain functioning. I. *Reviews of Modern Physics, 34*, 123–135.

Boahen, K. (1998). Communicating neuronal ensembles between neuromorphic chips. In S. Smith & A. Hamilton (Eds.), *Neuromorphic systems engineering*. Norwell, MA: Kluwer.

Brunel, N., Carusi, F., & Fusi, S. (1998). Slow stochastic Hebbian learning of classes of stimuli in a recurrent neural network. *Network, 9*(1), 123–152.

Brunel, N., Hakim, V., Isope, P., Nadal, J.-P., & Barbour, B. (2004). Optimal information storage and the distribution of synaptic weights: Perceptron versus Purkinje cells. *Neuron, 43*, 745–757.

Buchs, N., & Senn, W. (2002). Spike-based synaptic plasticity and the emergence of direction selective simple cells: Simulation results. *Journal of Computational Neuroscience, 13*, 167–186.

Burkitt, A. N., Meffin, H., & Grayden, D. B. (2004). Spike timing-dependent plasticity: The relationship to rate-based learning for models with weight dynamics determined by a stable fixed-point. *Neural Computation, 16*, 885–940.

Chicca, E., & Fusi, S. (2001). Stochastic synaptic plasticity in deterministic aVLSI networks of spiking neurons. In F. Rattay (Ed.), *Proceedings of the World Congress on Neuroinformatics* (pp. 468–477). Verlag, Vienna: ARGESIM/ASIM.

Chicca, E., Indiveri, G., & Douglas, R. (2003). An adaptive silicon synapse. *IEEE International Symposium on Circuits and Systems.* Piscataway, NJ: IEEE Press.

Coath, M., Brader, J., Fusi, S., & Denham, S. (2005). Multiple views of the response of an ensemble of spectro-temporal features support concurrent classification of utterance, prosody, sex and speaker identity. *NETWORK Computation in Neural Systems, 16*, 285–300.

Diorio, C., Hasler, P., Minch, B., & Mead, C. (1996). A single-transistor silicon synapse. *IEEE Transactions on Electron Devices, 43*, 1972–1980.

Douglas, R., Deiss, S., & Whatley, A. (1998). A pulse-coded communications infrastructure for neuromorphic systems. In W. Maass & C. Bishop (Eds.), *Pulsed neural networks* (pp. 157–178). Cambridge, MA: MIT Press.

Feldman, D. (2000). Abstract timing-based LTP and LTD at vertical inputs to layer II/III pyramidal cells in rat barrel cortex. *Neuron, 27*, 45–56.

Freund, Y., & Schapire, R. (1999). A short introduction to boosting. *Journal of the Japanese Society for Artificial intelligence, 14*, 771–780.

Froemke, R., & Dan, Y. (2002). Spike-timing-dependent synaptic modification induced by natural spike trains. *Nature, 416*, 433–438.

Fusi, S. (2001). long-term memory: Encoding and storing strategies of the brain. In J. M. Bower (Ed.), *Neurocomputing: Computational neuroscience: Trends in research 2001* (Vol. 38–40, pp. 1223–1228). Amsterdam: Elsevier Science.

Fusi, S. (2002). Hebbian spike-driven synaptic plasticity for learning patterns of mean firing rates. *Biol. Cybern., 87*, 459.

Fusi, S. (2003). Spike-driven synaptic plasticity for learning correlated patterns of mean firing rates. *Rev. Neurosci., 14*, 73–84.

Fusi, S., & Abbott, L. (2007). Limits on the memory-storage capacity of bounded synapses. *Nature Neuroscience, 10*, 485–493.

Fusi, S., Annunziato, M., Badoni, D., Salamon, A., & Amit, D. (2000). Spike-driven synaptic plasticity: Theory, simulation, VLSI implementation. *Neural Computation, 12*, 2227–2258.

Fusi, S., Drew, P., & Abbott, L. (2005). Cascade models of synaptically stored memories. *Neuron, 45*, 599–611.

Fusi, S., & Mattia, M. (1999). Collective behavior of networks with linear (VLSI) integrate-and-fire neurons. *Neural Comput., 11*(3), 633–653.

Fusi, S., & Senn, W. (2006). Eluding oblivion with smart stochastic selection of synaptic updates. *Chaos, 16*, 026112.

Gerstner, W., Kempter, R., van Hemmen, J., & Wagner, H. (1996). A neuronal learning rule for sub-millisecond temporal coding. *Nature, 383*, 76–78.

Giudice, P. D., Fusi, S., & Mattia, M. (2003). Modeling the formation of working memory with networks of integrate-and-fire neurons connected by plastic synapses. *Journal of Physiology, Paris, 97*, 659–681.

Giudice, P. D., & Mattia, M. (2001). Long and short-term synaptic plasticity and the formation of working memory: A case study. *Neurocomputing, 38–40*, 1175–1180.

Gutig, R., & Sompolinsky, H. (2006). The tempotron: A neuron that learns spike timing-based decisions. *Nat. Neurosci., 9*(3), 420–428.

Hertz, J., Krogh, A., & Palmer, R. (1991). *Introduction to the theory of neural computation*. Reading, MA: Addison-Wesley.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent selective computational abilities. *Proc. Natl. Acad. Sci. USA, 79*, 2554.

Hopfield, J., & Brody, C. D. (2004). Learning rules and network repair in spike-timing-based computation networks. *PNAS, 101*, 337–342.

Indiveri, G. (2000). Modeling selective attention using a neuromorphic AVLSI device. *Neural Comput., 12*, 2857–2880.

Indiveri, G. (2001). A neuromorphic VLSI device for implementing 2D selective attention systems. *IEEE Transactions on Neural Networks, 12*, 1455–1463.

Indiveri, G. (2002). Neuromorphic bistable VLSI synapses with spike-timing-dependent plasticity. In S. Becker, S. Thrün, & K. Obermayer (Eds.), *Advances in neural information processing systems, 15*. Cambridge, MA: MIT Press.

Indiveri, G., & Fusi, S. (2007). Spike-based learning in VLSI networks of spiking neurons. In *Proceedings of the IEEE International Symposium on Circuits and Systems* (pp. 3371–3374). Piscataway, NJ: IEEE Press.

Karmarkar, U., & Buonomano, D. (2002). A model of spike-timing dependent plasticity: one or two coincidence detectors? *J. Neurophysiology, 88*, 507–513.

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

La Camera, G., Rauch, A., Lüscher, H.-R., Senn, W., & Fusi, S. (2004). Minimal models of adapted neuronal response to in vivo–like input currents. *Neural Comput., 16*, 2101–2124.

Legenstein, R., Naeger, C., & Maass, W. (2005). What can a neuron learn with spike-timing-dependent plasticity? *Neural Comput, 17*(11), 2337–2382.

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

Marr, D. (1969). A theory of cerebellar cortex. *J. Physiol., 202*, 435–470.

Minsky, M. L., & Papert, S. A. (1969). *Perceptrons*. Cambridge, MA: MIT Press.

Mitra, S., Fusi, S., & Indiveri, G. (2006). A VLSI spike-driven dynamic synapse which learns only when necessary. *IEEE International Symposium on Circuits and Systems ISCAS06* (pp. 2777–2780). Piscataway, NJ: IEEE Press.

Mongillo, G., Curti, E., Romani, S., & Amit, D. (2005). Learning in realistic networks of spiking neurons and spike-driven plastic synapses. *European Journal of Neuroscience, 21*, 3143–3160.

Nadal, J. P., Toulouse, G., Changeux, J. P., & Dehaene, S. (1986). Networks of formal neurons and memory palimpsests. *Europhys. Lett., 1*, 535.

Parisi, G. (1986). A memory which forgets. *J. Phys. A: Math. Gen., 19*, L617.

Petersen, C. C. H., Malenka, R. C., Nicoll, R. A., & Hopfield, J. J. (1998). All-or-none potentiation at CA3-CA1 synapses. *Proc. Natl. Acad. Sci. USA, 95*, 4732–4737.

Rao, R. P. N., & Sejnowski, T. J. (2001). Spike-timing-dependent Hebbian plasticity as temporal difference learning. *Neural Computation, 13*, 2221–2237.

Rauch, A., La Camera, G., Lüscher, H.-R., Senn, W., & Fusi, S. (2003). Neocortical pyramidal cells respond as integrate-and-fire neurons to in vivo–like input currents. *J. Neurophysiology, 90*, 1598–1612.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review, 65*, 386–408.

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

Saudargiene, A., Porr, B., & Wörgötter, F. (2004). How the shape of pre- and postsynaptic signals can influence STDP: A biophysical model. *Neural Computation, 16*, 595–625.

Senn, W., & Buchs, N. (2003). Spike-based synaptic plasticity and the emergence of direction selective simple cells: Mathematical analysis. *Journal of Computational Neuroscience, 14*, 119–138.

Senn, W., & Fusi, S. (2005a). Convergence of stochastic learning in perceptrons with binary synapses. *Phys. Rev. E., 71*, 061907.

Senn, W., & Fusi, S. (2005b). Learning only when necessary: Better memories of correlated patterns in networks with bounded synapses. *Neural Computation, 17*, 2106–2138.

Senn, W., Markram, H., & Tsodyks, M. (2001). An algorithm for modifying neurotransmitter release probability based on pre- and postsynaptic spike timing. *Neural Comput., 13*(1), 35–67.

Shouval, H., & Kalantzis, G. (2005). Stochastic properties of synaptic transmission affect the shape of spike time dependent plasticity curves. *J. Neurophysiology, 93*, 643–655.

Shouval, H. Z., Bear, M. F., & Cooper, L. N. (2002). A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *PNAS, 99*, 10831–10836.

Simard, P., Steinkraus, D., & Platt, J. (2003). Best practice for convolutional neural networks applied to visual document analysis. In *Proceedings of the Seventh International Conference on Document Analysis and Recogntion* (p. 958). Washington, DC: IEEE Computer Society.

Sjöström, P. J., Turrigiano, G. G., & Nelson, S. B. (2001). Rate, timing and cooperativity jointly determine cortical synaptic plasticity. *Neuron, 32*, 1149–1164.

Tsodyks, M. (1990). Associative memory in neural networks with binary synapses. *Mod. Phys. Lett., B4*, 713–716.

Turrigiano, G., & Nelson, S. (2000). Hebb and homeostasis in neuronal plasticity. *Curr. Opin. Neurobiol., 10*, 358–364.

Wang, H., & Wagner, J. (1999). Priming-induced shift in synaptic plasticity in the rat hippocampus. *J Neurophysiol., 82*, 2024–2028.

Wang, S., O'Connor, D., & Wittenberg, G. (2004). Steplike unitary events underlying bidirectional hippocampal synaptic plasticity. *Society for Neuroscience*, p. 57.6.

Wang, X.-J. (2002). Probabilistic decision making by slow reverbaration in cortical circuits. *Neuron, 36*, 955–968.

Yao, H., Shen, Y., & Dan, Y. (2004). Intracortical mechanism of stimulus-timing-dependent plasticity in visual cortical orientation tuning. *PNAS, 101*, 5081–5086.

Zhang, L. I., Tao, H. W., Holt, C. E., Harris, W. A., & Poo, M. (1998). A critical window for cooperation and competition among developing retinotectal synapses. *Nature, 395*, 37–44.

Zhou, Q., Tao, H., & Poo, M. (2003). Reversal and stabilization of synaptic modifications in a developing visual system. *Science, 300*, 1953.

Received March 23, 2005; accepted March 19, 2007.