

A Biological Gradient Descent for Prediction Through a Combination of STDP and Homeostatic Plasticity

Mathieu N. Galtier

m.galtier@jacobs-university.de

*School of Engineering and Science, Jacobs University Bremen gGmbH,
28759 Bremen, Germany*

Gilles Wainrib

wainrib@math.univ-paris13.fr

*Laboratoire Analyse Géométrie et Applications, Université Paris 13,
Villetaneuse, France*

Identifying, formalizing, and combining biological mechanisms that implement known brain functions, such as prediction, is a main aspect of research in theoretical neuroscience. In this letter, the mechanisms of spike-timing-dependent plasticity and homeostatic plasticity, combined in an original mathematical formalism, are shown to shape recurrent neural networks into predictors. Following a rigorous mathematical treatment, we prove that they implement the online gradient descent of a distance between the network activity and its stimuli. The convergence to an equilibrium, where the network can spontaneously reproduce or predict its stimuli, does not suffer from bifurcation issues usually encountered in learning in recurrent neural networks.

1 Introduction ---

One of the main functions of the brain is prediction (Bar, 2009). This function is generally thought to rely on the idea that cortical regions learn a model of the world and simulate it to generate predictions of future events (Gilbert & Wilson, 2007; Schacter, Addis, & Buckner, 2008). Several recent experimental findings support this view, showing in particular that the spontaneous neuronal activity after presentation of a stimulus is correlated with the evoked activity (Kenet, Bibitchkov, Tsodyks, Grinvald, & Arieli, 2003), and that this similarity increases along development and learning (Berkes, Orbán, Lengyel, & Fiser, 2011). Moreover, at the scale of neuronal networks, prediction can also be seen as a general organization principle. It has been argued (Rao, & Ballard, 1999; Clark, 2012) that the brain would contain a hierarchy of predictive units able to predict their direct stimuli or inputs through the modification of the synaptic connections.

Understanding the mechanisms and principles underlying this prediction function is a key challenge, not only from a neuroscience perspective but also for machine learning, where the number of applications requiring prediction is significant.

In the field of machine learning, recurrent neural networks have been successfully proposed as candidates for these predictive units (Williams & Zipser, 1989, 1995; Pearlmutter, 1995; Jaeger & Haas, 2004; Sussillo & Abbott, 2009). In most cases, these algorithms aim at creating a neural network that autonomously and spontaneously reproduces a given time series. The Bayesian approach is also useful in designing predictors (Dayan, Hinton, Neal, & Zemel, 1995; George & Hawkins, 2009) and has been mapped to neural networks (Deneve, 2008; Friston, 2010; Bitzer & Kiebel, 2012). However, apart from a rough conceptual equivalence, this letter is devoid of Bayesian terminology and directly focuses on neural networks. In this framework, prediction is often achieved by minimizing a distance between the activity of the neural network and the target time series. Although neural networks were originally studied in a feedforward framework (Rosenblatt, 1958), the most efficient networks for prediction shall involve recurrent connections giving the network some memory properties. So-called gradient descent algorithms in recurrent neural network (Mandic & Chambers, 2001) involve the learning of the entire connectivity matrix. They minimize the distance between a target trajectory and the trajectory of the network. On the other hand, researchers in the field of reservoir computing (Lukosevicius & Jaeger, 2009) optimize only some connections in the network, whereas the others are randomly drawn and fixed. To do prediction, they minimize the one-step-ahead error corresponding to the distance between the network predictions and the next time step of the target time series. Thus, these algorithms are derived to optimize an accuracy criterion, with learning rules generally favoring prediction efficiency over biological meaning.

The field of neuroscience has seen many discoveries in the study of synaptic plasticity, in particular providing experimental evidence and possible mechanisms for two major concepts in the current biology of learning. The first is the discovery of spike-timing-dependent plasticity (STDP) (Markram, Lübke, Frotscher, & Sakmann, 1997; Bi & Poo, 1998; Caporale & Dan, 2008; Sjöström & Gerstner, 2010). It is a temporally asymmetric form of Hebbian learning induced by temporal correlations between the spikes of pre- and postsynaptic neurons. The general principle is that if a neuron fires before (resp. after) another, then the strength of the connection from the former to the latter will be increased (resp. decreased). The summation of all of these modifications leads to the strengthening of causality links among neurons. Although STDP is based on spiking networks, it has several extensions or analogs for rate-based networks (those used in machine learning) (Kempster, Gerstner, & Van Hemmen, 1999; Izhikevich & Desai, 2003; Pfister & Gerstner, 2006). The functional role of STDP is still discussed—for instance, in reducing latency (Song, Miller, & Abbott, 2000), optimizing

information transfer (Hennequin, Gerstner, & Pfister, 2010), invariant recognition (Sprekeler, Michaelis, & Wiskott, 2007), and even learning temporal patterns (Gerstner, Ritz, & Van Hemmen, 1993; Rao & Sejnowski, 2001; Yoshioka, Scarpetta, & Marinaro, 2007).

Second, the notion of homeostatic plasticity (Miller, 1996; Abbott & Nelson, 2000; Turrigiano & Nelson, 2004), including mechanisms such as synaptic scaling, has proved to be important to moderate the growth of connection strength. In contrast to previously theory-motivated normalization of the connectivity (Miller & MacKay, 1994; Oja, 1982), there is need of a biologically plausible means to prevent the connectivity from exploding under the influence of shaping mechanisms like Hebbian learning or STDP.

From a theoretical viewpoint, STDP and homeostatic plasticity are almost always studied independently. An extensive bottom-up numerical analysis of the combination of such learning mechanisms, done by Triesch and colleagues, has already led to biologically relevant behaviors (Lazar, Pipa, & Triesch, 2007, 2009; Zheng, Dimitrakakis, & Triesch, 2013). However, the mathematical understanding of their combination in terms of functionality still stands as an undocumented challenge to researchers.

This letter aims at bridging the gap between biological mechanisms and machine learning regarding the issue of predictive neural networks. We rigorously show how a biologically inspired learning rule, developed from an original combination of STDP mechanism and homeostatic plasticity, mimics the gradient descent of a distance between the activity of the neural network and its direct stimuli. This results in capturing the underlying dynamical behavior of the stimuli into a recurrent neural network and therefore in designing a biologically plausible predictive network.

The letter is organized as follows. In section 2, we construct a theoretical learning rule designed for prediction, based on an appropriate gradient descent method. In section 3, we introduce a biologically inspired learning rule that combines the concepts of STDP and homeostatic plasticity, whose purpose is to mimic the theoretical learning rule. We discuss the various biological mechanisms that may be involved in this new learning rule. Finally in section 4, we provide a mathematical justification of the link between the theoretical and the biologically inspired learning rule, based on the key idea that STDP can be seen as a differential operator.

2 Theoretical Learning Rule for Prediction

In a machine learning approach, we introduce a procedure to design a neural network that autonomously replays a target time series.

2.1 Setup. We consider a recurrent neural network made of n neurons that is exposed to a time-dependent input $\mathbf{u}(t) \in \mathbb{R}^n$ of the same dimension. Our aim is to construct a learning rule that will enable the network to reproduce the input's behavior.

Our approach is focused on learning the underlying dynamics of the input. Therefore, we assume that \mathbf{u} is generated by an arbitrary dynamical system,

$$\dot{\mathbf{u}} = \xi(\mathbf{u}), \quad (2.1)$$

with ξ a smooth vector field from \mathbb{R}^n to \mathbb{R}^n . We also assume that the trajectory of the inputs or stimuli is τ -periodic. The key mathematical assumption on the input \mathbf{u} is in fact ergodicity, but we restrict our study to periodic inputs for simplicity. In particular, periodic inputs can be constructed by the repetition of a given finite-time sample.

Although the following method virtually works with any network equations, we focus on a neural network composed of n neurons and governed by

$$\dot{\mathbf{v}} = -l\mathbf{v} + \mathbf{W}.S(\mathbf{v}), \quad (2.2)$$

where $\mathbf{v} \in \mathbb{R}^n$ is a vector representing neuronal activity, $\mathbf{W} \in \mathbb{R}^{n \times n}$ is the connectivity matrix, $l \in \mathbb{R}_+$ is a decay constant, and S is an entry-wise sigmoid function.

2.2 Gradient Descent Learning Rule. The idea behind our learning rule is to find the best connectivity matrix \mathbf{W} that will minimize a distance between the two functions $\xi(\mathbf{u})$ and $-l\mathbf{u} + \mathbf{W}.S(\mathbf{u})$. In this perspective, we define the following quantity:

$$H_{\mathbf{u}} := \frac{1}{2} \int_0^\tau \| -l\mathbf{u}(t) + \mathbf{W}.S(\mathbf{u}(t)) - \xi(\mathbf{u}(t)) \|^2 dt. \quad (2.3)$$

When $H_{\mathbf{u}} = 0$, the vector fields of systems 2.1 and 2.2 are equal on the trajectories of the inputs. This quantity may be viewed as a distance between the two vector fields defining the dynamics of the inputs and the neuronal network along the trajectories of the inputs. It is similar to classical gradient methods (Pearlmutter, 1995; Williams & Zipser, 1995; Mandic & Chambers, 2001), except that the norms are applied to the flows of inputs and neural networks instead of their activity. Thus, it focuses more specifically on the dynamical structure of the inputs. Moreover, it is possible to show, using Girsanov's theorem, that this definition coincides with the concept of relative entropy between two diffusion processes—the ones obtained by adding a standard gaussian perturbation to both equations. Therefore, we call $H_{\mathbf{u}}$ the relative entropy.

In order to capture the dynamics of the inputs the network, it is natural to look for a learning rule minimizing this quantity. To this end, we consider the gradient of this measure with respect to the connectivity matrix,

$$\nabla_{\mathbf{W}} H_{\mathbf{u}} = -[\xi(\mathbf{u}) \cdot S(\mathbf{u})' + l\mathbf{u} \cdot S(\mathbf{u})' - \mathbf{W}.S(\mathbf{u}) \cdot S(\mathbf{u})'], \quad (2.4)$$



Figure 1: Learning how to write the letter A. Time evolution of the input movie (top row) and the network activity after learning (bottom row). Each pixel corresponds to a neuron. Parameters: Number of neurons $n = 400$; $l = 1$; $S(x) = \tanh(x)$.

where the component i, j of $\mathbf{x} \cdot \mathbf{y}' \in \mathbb{R}^{n \times n}$ is $\{\mathbf{x} \cdot \mathbf{y}'\}_{ij} = \int_0^t x_i(s) y_j(s) ds$ for \mathbf{x}, \mathbf{y} functions from $[0, \tau[$ to \mathbb{R}^n . Equivalently, these functions can be seen as semicontinuous matrices in $\mathbb{R}^{n \times [0, \tau]}$, and \mathbf{x}' is the transpose of \mathbf{x} . Thus, an algorithm implementing the gradient descent $\dot{\mathbf{W}} = -\nabla_{\mathbf{W}} H_{\mathbf{u}}$ is a good candidate to minimize the relative entropy between inputs and spontaneous activity. Since $H_{\mathbf{u}}$ is quadratic in \mathbf{W} , it follows that $\mathbf{W} \rightarrow H_{\mathbf{u}}$ is a convex function, thus excluding situations with multiple local minima. Moreover, if $S(\mathbf{u}) \cdot S(\mathbf{u})'$ is invertible, one can compute directly the minimizing connectivity as

$$\mathbf{W}^* = [\xi(\mathbf{u}) \cdot S(\mathbf{u})' + l\mathbf{u} \cdot S(\mathbf{u})'] \cdot [S(\mathbf{u}) \cdot S(\mathbf{u})']^{-1}. \quad (2.5)$$

Implementing a gradient descent based on equation 2.4 does not immediately lead to a biologically relevant mechanism. First, it requires direct access to the inputs \mathbf{u} , whereas synaptic plasticity mechanisms rely on only the network activity \mathbf{v} . Second, it is a batch learning algorithm, which requires access to the entire history of the inputs. Third, it requires the ability to compute $\xi(\mathbf{u})$. We will see in section 3 how to overcome these issues combining biologically inspired synaptic plasticity mechanisms.

2.3 Example. In order to illustrate the idea that learning rule 2.4 enables the network to learn dynamical features of the input, we have constructed the following experiment. We present to the network an input movie displaying sequentially the writing of the letter A (see Figure 1, top row). To each pixel we assign one neuron, so that the input and the network share the same dimension. This input movie is repeated periodically until the connectivity matrix of the network, evolving under rule 2.4, stabilizes. Then the input is turned off, and we set the initial state of the network to a priming image showing the bottom left part of letter A. The network evolving without input strikingly reproduces the dynamical writing of letter A

as displayed in Figure 1. Thus, with this example, we have illustrated the ability of the learning rule we have derived from a theoretical principle to capture a dynamic input into a connectivity matrix.

3 A Biological Learning Rule

We now introduce a biological learning rule made of the combination of STDP and homeostatic plasticity. In section 4, we show that this learning rule minimizes $H_{\mathbf{u}}$. Here, we first give a mathematical description of this learning rule and relate the different terms to biological mechanisms.

3.1 Mathematical Description. Learning corresponds to a modification of the connectivity simultaneous to the network activity evolution. The result is a coupled system of equations. The learning rate ϵ is chosen to be small, $\epsilon \ll 1$, so that learning can be considered slow compared to the evolution of the activity. The full online learning system is

$$\begin{cases} \dot{\mathbf{v}} = -l\mathbf{v} + \mathbf{W}.S(\mathbf{v}) + \mathbf{u} \\ \frac{1}{\epsilon} \dot{\mathbf{W}}_{ij} = \underbrace{\delta[\bar{\mathbf{v}}_i, S(\bar{\mathbf{v}}_j)]}_{\text{STDP}} - \underbrace{\sum_k \mathbf{W}_{ik} S(\bar{\mathbf{v}}_k) S(\bar{\mathbf{v}}_j)}_{\text{homeostatic plasticity}} \end{cases} \quad (3.1)$$

where

$$\bar{\mathbf{v}} = l\mathbf{v} - \mathbf{W}.S(\mathbf{v}) * g_l$$

and

$$\delta[x, y] = \frac{\gamma + l}{2} x(y * g_\gamma) - \frac{\gamma - l}{2} (x * g_\gamma)y, \quad (3.2)$$

where the notation $*$ denotes the convolution operator. The function g_c is defined as $g_c : t \mapsto ce^{-ct}H(t)$, with H the Heaviside function and for any positive number c , as shown in the left panel of Figure 2. As illustrated in the right panel of Figure 2, the operator δ roughly corresponds to the classical STDP window (Bi & Poo, 1998) (taking into account a y -axis symmetry corresponding to the symmetric formalism we are using). The constant $\gamma \in \mathbb{R}_+$ is a time constant corresponding to the width of the STDP window used for learning.

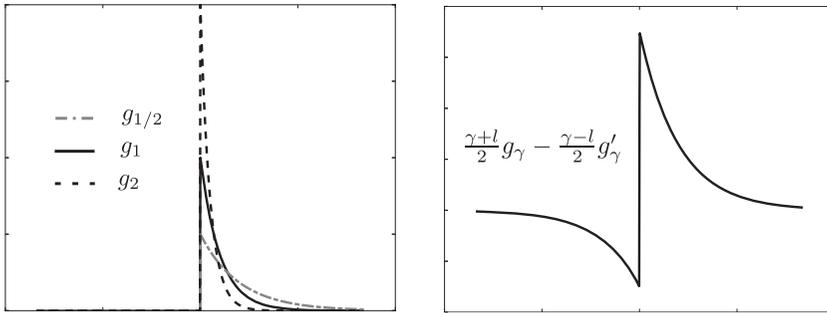


Figure 2: (Left) Plots of the function g_c for $c \in \{1/2, 1, 2\}$. (Right) Plot of the function $\Delta_\gamma(t) + l\Sigma_\gamma(t) = \frac{\gamma+l}{2}g_\gamma(t) - \frac{\gamma-l}{2}g'_\gamma(-t)$ for $\gamma = 0.5$ and $l = 0.2$. This function corresponds to the operator δ as shown in section 4.1.2.

3.2 Biological Mechanisms

3.2.1 An Input Estimate. The variable $\bar{\mathbf{v}}$ can be seen as a spatiotemporal differential variable that approximates the inputs \mathbf{u} . Although unsupervised learning rules are often algebraic combinations of element-wise functions applied to the activity of the network (Gerstner & Kistler, 2002), it is not precisely the case here. Indeed, learning is based on the variable $\bar{\mathbf{v}}$, which corresponds to the subtraction of the temporally integrated synaptic drive $\mathbf{W} \cdot \mathbf{S}(\mathbf{v}) * g_l$ from the activity of the neurons $l\mathbf{v}$. For each neuron, this variable takes into account the past of all the neurons, which are then spatially averaged through the connectivity to be subtracted from the current activity. This gives a differential flavor to this variable that is reminiscent of former learning rules (Bienenstock, Cooper, & Munro, 1982; Sejnowski, 1977) for the temporal aspect and (Miller & MacKay, 1994) for the spatial aspect. Note that this variable is not strictly speaking local (i.e., the connection \mathbf{W}_{ij} needs the values of \mathbf{v}_k to be updated), yet it is biologically plausible since the term $(\mathbf{W} \cdot \mathbf{S}(\mathbf{v}))_i$ is accessible for neuron i on its dendritic tree, a form of locality in a broader sense.

3.2.2 An STDP Mechanism. The first term $\delta[\bar{\mathbf{v}}_i, S(\bar{\mathbf{v}}_i)]$ in equation 3.1 can be related to STDP. The antisymmetric part of this term is responsible for retrieving the drift ξ in equation 2.4. The symmetric part (corresponding to Hebbian learning) is responsible for retrieving the second term in equation 2.4. Thus, it captures the causality structure of the inputs, a task generally attributed to STDP (Sjöström & Gerstner, 2010). Beyond the simple similarity of functional role, we believe a simplification of this term may shed light on the deep link it has with STDP. The main difference between our setup and STDP is that the former is based on a rate-based dynamics, whereas the latter is based on a spiking dynamics. In a pure spike framework (i.e.,

the activity is a sum of Diracs), the STDP can be seen as this simple learning rule: $\dot{W}_{ij} = \delta[\mathbf{v}_i, \mathbf{v}_j] = \frac{\gamma+1}{2}\mathbf{v}_i(\mathbf{v}_j * g_\gamma) - \frac{\gamma-1}{2}(\mathbf{v}_i * g_\gamma) \mathbf{v}_j$. Indeed, the term $\mathbf{v}_i(\mathbf{v}_j * g_\gamma)$ is nonnull only when the postsynaptic neuron i is firing and then, via the factor $\mathbf{v}_j * g_\gamma$, it counts the number of preceding presynaptic spikes that might have caused i 's excitation and weight them by the decreasing exponential g_γ . Thus, this term exactly accounts for the positive part of the STDP curve. The negative term $-(\mathbf{v}_i * g_\gamma)\mathbf{v}_j$ takes the opposite perspective and accounts for the negative part of the STDP curve. A loose extension of this rule to the case where the activity is smoothly evolving leads to identifying the function δ to the STDP mechanism for rate-based networks (Galtier, 2012; Izhikevich & Desai, 2003).

3.2.3 Homeostatic Plasticity. The second term $-\sum_k \mathbf{W}_{ik} S(\bar{\mathbf{v}}_k) S(\bar{\mathbf{v}}_j)$ in equation 3.1 accounts for what is usually presented as homeostatic plasticity mechanisms. The previous STDP term seems to be a powerful mechanism to shape the response of the network. However, there is a need for a regulatory process to prevent uncontrolled growth of the network connectivity (Abbott & Nelson, 2000; Turrigiano & Nelson, 2004; Miller, 1996). It has been argued that STDP could be self-regulatory (Van Rossum, Bi, & Turrigiano, 2000; Song et al., 2000), but it is not the case in our framework, and an explicit balancing mechanism is necessary to avoid the divergence of the system. This last term is the only one with a negative sign and is multiplicative with respect to the connectivity. Thus, according to Abbott and Nelson (2000), it is a reasonable candidate for homeostasis. It has been argued (Turrigiano, Leslie, Desai, Rutherford, & Nelson, 1998; Kim, Tsien, & Alger, 2012) that homeostatic plasticity might keep the relative synaptic weights by dividing the connectivity with a common scaling factor, theoretically preventing a possible information loss. In contrast to these ad hoc renormalizations often introduced in other learning rules (Miller & MacKay, 1994; Oja, 1982), our relative entropy-minimizing learning rule thus naturally introduces an original form of homeostatic plasticity.

Although we have separated the description of the various terms in equation 3.1, our approach suggests that homeostasis may be seen not necessarily as a scaling term but as a constitutive part of a learning principle, deeply entangled (Turrigiano, 1999) with the other learning mechanisms.

3.3 Numerical Application. Although the focus of the letter is on theory, we introduce a simple numerical example to illustrate the predictive properties of the biological learning rule. More precisely, we investigate the question of retrieving the connectivity of a neural network based on the observation of the time series of its activity. This is an inverse problem, usually a challenging topic in computational neuroscience (Friston, Harrison, & Penny, 2003; Galán, 2008; Potthast & beim Graben, 2009) since it may give access to large-scale effective connectivities simply from observing a

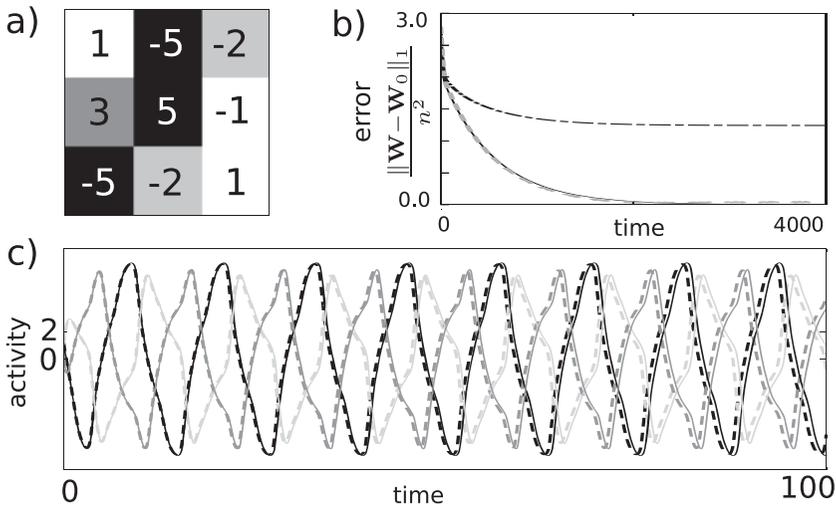


Figure 3: Retrieving the connectivity. (a) Connectivity matrix W_0 of the input network. (b) Evolution of the difference between current connectivity and input connectivity through learning. The black dot-dashed curve corresponds to the online learning rule, equation 3.1, in the homogeneous case— $l = l_{net} = l_0$ in both equations in equation 3.1. The gray dashed curve corresponds to the online learning system 3.1 in the hybrid framework: $l = l_0$ for the learning equation and $l = l_{net} \gg l_0$ for the network equation. For this simulation, we chose $l_{net} = 50$. The solid black curve (superposed on the gray dashed curve) corresponds to the batch relative entropy minimization, equation 2.4. (c) The dashed curves correspond to the activity of the inputs. It is a three-dimensional input ($n = 3$), and the three different gray levels correspond to the different dimensions. The solid curves correspond to the simulation of the network (top equation in equation 3.1) after learning, in the hybrid framework, and without inputs. The parameters for these simulations are $l_0 = 1$, $\epsilon = 0.01$, and $\gamma = 100$.

neuronal activity. Here we address it in an elementary framework. The network generating the activity patterns, referred to as an input network, evolves according to $\dot{\mathbf{u}} = -l_0 \mathbf{u} + W_0 \cdot S(\mathbf{u})$. For this example, the network is made of $n = 3$ neurons, and its connectivity W_0 is shown in Figure 3a. These parameters were chosen so that the activity is periodic, as shown by the dashed curves in Figure 3c. Then we simulate the entire system, equation 3.1, with a decay constant l_{net} and observe that its connectivity W_{net} converges to W_0 .

As we show in the next section, it is necessary that $\mathbf{u} * g_l * g_\gamma \simeq \mathbf{u}$ in order to approximate accurately the input's activity with the online learning rule, equation 3.1. Given that the time constant of the inputs is governed by l_0 , the previous approximation holds only if $\gamma, l_{net} \gg l_0$. If this assumption is

broken (e.g., $l_{net} = l_0$), then the final connectivity matrix is different from \mathbf{W}_0 (see the black dot-dashed curve in Figure 3b). Indeed, in this case, the network learns only to replay the slow variation of the inputs.

A method to recover the precise time course of the inputs consists in artificially changing the time constants at different steps of an algorithm. First, simulate the network equation (top equation in equation 3.1) with a constant $l_{net} \gg l_0$. Yet the time constant in the learning equation (bottom equation in equation 3.1) is to be kept at l_0 , thus introducing a hybrid model. In this framework, the connectivity converges exactly to \mathbf{W}_0 as shown in the gray dashed curve in Figure 3b. After learning, simulate the network equation with the learned connectivity and with a time constant switched back to l_0 . This gives an activity as shown in the solid curves in Figure 3c.

4 Link Between Theoretical and Biological Learning Rules _____

In this section, we show how the biological learning rule 3.1 implements the gradient descent based on equation 2.4. First, we introduce three technical results, which are the mathematical cornerstones of the letter, and then combine them to obtain the desired result.

4.1 Three Technical Results. Equation 2.4 is not biologically plausible for three main reasons: (1) it requires a direct access to the inputs \mathbf{u} , whereas synaptic plasticity mechanisms rely on only the network activity \mathbf{v} ; (2) it is a batch learning algorithm, which requires access to the entire history of the inputs; and (3) it requires the ability to compute $\xi(\mathbf{u})$ (equal to the time derivative of the inputs according to equation 2.1).

On the contrary, the biological learning rule, equation 3.1, does not have these problems. First, it uses an estimate of the inputs, noted $\bar{\mathbf{v}}$, based on the activity variable only. Second, it is an online learning rule; it takes input on the fly and relies on a slow-fast mechanism to temporally average the variables. Third, it approximates the computation of $\xi(\mathbf{u})$ with a function δ inspired from STDP convolution. In the following, we mathematically address these three points successively.

4.1.1 $\bar{\mathbf{v}}$ Is an Input Estimate. The online learning rule, 3.1, is expressed by means of the activity of the neural network \mathbf{v} governed by the top part of equation 3.1. However, to be comparable to the gradient of the relative entropy, equation 2.4, we first need to make explicit the dependence on the inputs \mathbf{u} . Therefore, we show that the network dynamics induces a simple relation between $\bar{\mathbf{v}}$ and the inputs \mathbf{u} . A simple computation in the Fourier domain shows that the convolution between the temporal operator $\frac{d}{dt} + \mathbb{I}_d$ and g_l results in \mathbb{I}_d . Applying this result to the neural dynamics leads to reformulating the top part of equation 3.1 as $l\mathbf{v} - \mathbf{W} \cdot S(\mathbf{v}) * g_l = \mathbf{u} * g_l$. We recognize the definition of the variable $\bar{\mathbf{v}}$ (originally defined according

to this relation) such that the network's dynamics of the fast equation in equation 3.1 is equivalent to

$$\bar{\mathbf{v}} = \mathbf{u} * g_l. \quad (4.1)$$

4.1.2 Temporal Averaging of the Learning Rule. To prove that equation 3.1 implements the gradient descent of the relative entropy, we need to use a timescale separation assumption, enabling the input to reveal its dynamical structure through ergodicity. Indeed, the fact that learning is very slow compared to activity \mathbf{v} corresponds to the assumption $\epsilon \ll 1$. In this case, we can apply classic results of periodic averaging (Sanders & Verhulst, 1985, see also Galtier & Wainrib, 2012, in the context of neural networks) to show that the evolution of \mathbf{W} is well approximated by

$$\dot{\mathbf{W}} = [\bar{\mathbf{v}} * \Delta_\gamma + l\bar{\mathbf{v}} * \Sigma_\gamma - \mathbf{W} \cdot S(\bar{\mathbf{v}})] \cdot S(\bar{\mathbf{v}})', \quad (4.2)$$

where $\Delta_\gamma : t \mapsto \frac{\gamma}{2}(g_\gamma(-t) - g_\gamma(t))$ and $\Sigma_\gamma : t \mapsto \frac{g_\gamma(-t) + g_\gamma(t)}{2}$. The right panel of Figure 2 shows a plot of the function $\Delta_\gamma + l\Sigma_\gamma$.

4.1.3 STDP as a Differential Operator. Here, we prove the two following equalities

$$[\mathbf{x} * \Sigma_\gamma] \cdot \mathbf{y}' = (\mathbf{x} * g_\gamma) \cdot (\mathbf{y} * g_\gamma)' \quad (4.3)$$

and

$$[\mathbf{x} * \Delta_\gamma] \cdot \mathbf{y}' = (\hat{\mathbf{x}} * g_\gamma) \cdot (\mathbf{y} * g_\gamma)' \quad (4.4)$$

This second equation is the key mathematical mechanism that makes STDP a good approximation of the temporal derivative of the inputs.

Both proofs consist in going in the Fourier domain, where convolutions are turned into multiplications. We use the unitary, ordinary frequency convention for the Fourier transform. Observe that the Fourier transform of g_γ is $\hat{g}_\gamma(\xi) = \frac{\gamma}{\gamma + 2i\pi\xi}$. And we define $g'_\gamma : t \mapsto g_\gamma(-t)$ such that $\Delta_\gamma = \frac{\gamma}{2}(g'_\gamma - g_\gamma)$ and $\hat{g}'_\gamma(\xi) = \hat{g}_\gamma(-\xi)$.¹

¹This notation is motivated by the following: the convolution with g_γ can be written as a matrix multiplication with a continuous Toeplitz matrix $\mathbb{R}^{\mathbb{R} \times \mathbb{R}}$ whose component ts is $g_\gamma(t - s)$. In this framework, the convolution with g'_γ corresponds to the multiplication with the transpose of the previous Toeplitz operator.

Let us show that $[\mathbf{x} * \Sigma_\gamma] \cdot \mathbf{y}' = (\mathbf{x} * g_\gamma) \cdot (\mathbf{y} * g_\gamma)'$. We proceed in two steps:

1. We show that $\frac{g_\gamma + g'_\gamma}{2} = g_\gamma * g'_\gamma$. The Fourier transform of the convolution $g_\gamma * g'_\gamma$ is the product $\frac{\gamma}{\gamma + 2i\pi\xi} \frac{\gamma}{\gamma - 2i\pi\xi} = \frac{\gamma^2}{\gamma^2 + 4\pi^2\xi^2}$. From the usual Fourier tables, we observe that the right-hand side is the Fourier transform of $t \mapsto \frac{\gamma}{2} e^{-\gamma|t|}$. This immediately leads to the result.
2. We show that $(\mathbf{x} * g'_\gamma) \cdot \mathbf{y}' = \mathbf{x} \cdot (\mathbf{y} * g_\gamma)'$ Compute

$$\begin{aligned} \{(\mathbf{x} * g'_\gamma) \cdot \mathbf{y}'\}_{ij} &= \int_{-\infty}^{\infty} (\mathbf{x}_i * g'_\gamma)(s) \mathbf{y}_j(s) ds \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{+\infty} \mathbf{x}_i(t) g'_\gamma(s - t) \mathbf{y}_j(s) ds dt \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{+\infty} \mathbf{x}_i(t) g_\gamma(t - s) \mathbf{y}_j(s) ds dt \\ &= \{\mathbf{x} \cdot (\mathbf{y} * g_\gamma)'\}_{ij}. \end{aligned}$$

Using result (1) and applying result (2) to $\mathbf{x} * g_\gamma$ and \mathbf{y} leads to the result $[\mathbf{x} * \Sigma_\gamma] \cdot \mathbf{y}' = (\mathbf{x} * g_\gamma) \cdot (\mathbf{y} * g_\gamma)'$.

To prove equation 4.4, we first show that $\mathbf{x} * \Delta_\gamma = \dot{\mathbf{x}} * \frac{g_\gamma + g'_\gamma}{2}$. The Fourier transform of the convolution $\mathbf{x} * \Delta_\gamma$ is the product $\hat{\mathbf{x}} \hat{\Delta}_\gamma$. Besides,

$$\begin{aligned} \frac{2}{\gamma} \hat{\Delta}_\gamma &= \widehat{g'_\gamma - g_\gamma}(\xi) = \frac{\gamma}{\gamma - 2i\pi\xi} - \frac{\gamma}{\gamma + 2i\pi\xi} = 2i\pi\xi \frac{2\gamma}{\gamma^2 + 4\pi^2\xi^2} \\ &= \frac{2i\pi\xi}{\gamma} \left(\frac{\gamma}{\gamma + 2i\pi\xi} + \frac{\gamma}{\gamma - 2i\pi\xi} \right) = \frac{2i\pi\xi}{\gamma} \widehat{(g'_\gamma + g_\gamma)}(\xi). \end{aligned}$$

Because $\frac{d\hat{\mathbf{x}}}{dt}(\xi) = \hat{\mathbf{x}} 2i\pi\xi$, taking the inverse Fourier transform of $\hat{\mathbf{x}} 2i\pi\xi \widehat{\frac{g'_\gamma + g_\gamma}{2}}(\xi)$ gives the intermediary result $\mathbf{x} * \Delta_\gamma = \dot{\mathbf{x}} * \frac{g_\gamma + g'_\gamma}{2}$. Using it and applying the first result to $\dot{\mathbf{x}}$ and \mathbf{y} leads to equation 4.4.

4.2 Main Result. We prove here that the biological learning rule, 3.1, implements the gradient descent based on equation 2.4.

The temporally averaged version of the biological learning rule is given by equation 4.2. We simultaneously use equations 4.1 and 4.4 on this formula to show that the solution of the biological learning rule is well approximated by

$$\begin{aligned} \dot{\mathbf{W}} &= [\xi(\mathbf{u} * g_l) * g_\gamma] \cdot [S(\mathbf{u} * g_l) * g_\gamma]' + l(\mathbf{u} * g_l * g_\gamma) \cdot (S(\mathbf{u} * g_l) * g_\gamma)' \\ &\quad - \mathbf{W} \cdot S(\mathbf{u} * g_l) \cdot S(\mathbf{u} * g_l)' \end{aligned}$$

If \mathbf{u} is slow enough (i.e., $\mathbf{u} * g_t * g_y \simeq \mathbf{u}$), then this equation is precisely the gradient descent of $H_{\mathbf{u}}$. If \mathbf{u} is too fast, the network will learn only to predict the slow variations of the inputs. Some fast-varying information is lost in the averaging process, mainly because the network equation in equation 3.1 acts as a relaxation equation, which filters the activity. Note that this loss of information is not necessarily a problem for the brain, since it may be extracting and treating information at different timescales (Kiebel, Daunizeau, & Friston, 2008). Actually, the choice of the parameter l specifies the timescale under which the inputs are observed.

4.3 Stability. Both theoretical and biological learning rules are assured to make the connectivity converge to an equilibrium point whatever the initial condition. In particular, this method does not suffer from the bifurcation issues often encountered in recurrent neural network learning (Doya, 1992). In most cases, problems arise when learning leads the network activity to a bifurcation. The bifurcation suddenly changes the value of the quantity being minimized, and this intricate coupling leads to very slow convergence rates. The reason that this method has such an unproblematic converging property is that the relative entropy (as opposed to other energy functions traditionally used) is independent of the network activity \mathbf{v} . Besides, for any network pattern \mathbf{v} , equation 4.1 ensures that $\bar{\mathbf{v}}$ is always equal to the convolved inputs. These two arguments break the pathologic coupling preventing from getting interesting convergence results.

From a mathematical perspective, the Krasovskii-Lasalle invariance principle (Khalil & Grizzle, 1992) ensures that the theoretical learning rule $\dot{\mathbf{W}} = -\nabla_{\mathbf{W}} H_{\mathbf{u}}$ converges to an equilibrium point. Indeed, the relative entropy acts as an energy or Lyapunov function. If $S(\mathbf{u}) \cdot S(\mathbf{u})'$ is invertible, then the equilibrium point is unique and defined by equation 2.5. If it is not invertible, then the equilibrium depends on the initial condition.

If the inputs are slow enough, it has been shown that biological learning is well approximated by the theoretical gradient descent. Therefore, the former exhibits the same stable converging behavior as the latter. Thus, the biological learning rule, equation 3.1, is stable. In practice, we have never encountered a diverging situation even when the inputs are not slow enough.

5 Discussion and Conclusion

We have shown that a biological learning rule shapes a network into a predictor of its stimuli. This learning rule is made of a combination of an STDP mechanism and homeostatic plasticity. After learning, the network would spontaneously predict and replay the stimuli it has previously been exposed to. This was achieved by showing that this learning rule minimizes a quantity analogous to the relative entropy (or Kullback-Leibler

divergence) between the stimuli and the network activity as for other well-known algorithms (Ackley, Hinton, & Sejnowski, 1985; Hinton, 2009).

We believe this letter brings interesting arguments in the debate concerning the functional role of STDP. We have shown that the antisymmetric part of STDP can be seen as a differential operator. When its effect is moderated by an appropriate scaling term, we argue that it could correspond to a generic mechanism shaping neural networks into predictive units. This idea is not new, but this letter may give it a stronger theoretical basis.

This study also gives central importance to the time constants characterizing the network. Indeed, the fact the biological learning rule, equation 3.1, implements a gradient descent is rigorously true for slow inputs. Inputs are slow if they are significantly slower than both the time window of the STDP and the decay constant of the network. This means that subnetworks in the brain could select the frequency of the information they are predicting. Given that the brain may process information at different timescales (Kiebel et al., 2008), this may be an interesting feature. Besides, note that the proposed biological learning rule, equation 3.1, is partly characterized by the activity decay parameter l . This is surprising because it links the intrinsic dynamics of the neurons to the learning processes corresponding to the synapse. Therefore, it may provide a direction to experimentally check this theory. Is the symmetric part of STDP (Hebbian learning) stronger between faster neurons?

One of the characteristic features of this research is the combination of rate-based networks and the concept of spike-timing-dependent plasticity. Obviously this made it impossible to consider the rigorous definition of STDP. However, we have argued that the function δ in equation 3.2 is an alternative formulation equivalent to the others in the spiking framework, and it can be trivially extended to analog networks. Besides, it does capture the functional mechanism of the STDP in the rate-based framework. When the activation of a neuron precedes (resp. follows) the activation of another, the strength of the synapse from the former to the latter is increased (resp. decreased). Finally, we believe the theory presented in this letter gives support to this rate-based STDP since it appears to fill a gap between machine learning and biology by implementing a differential operator.

This approach can be applied to forms of network equations other than equation 2.2, such as the Wilson-Cowan or Kuramoto models, leading to different learning rules. In this perspective, learning can be seen as a projection of a given arbitrary dynamical system to a versatile neuronal network model, and the learning rule will depend on the chosen model. However, any network equation with an additive structure—intrinsic dynamics plus communication with other neurons—as in equation 2.2 will lead to a similar structure for the learning rule, with terms that may share similar biological interpretations as developed above. A special case is the linear network $\dot{\mathbf{v}} = -l\mathbf{v} + \mathbf{W}\cdot\mathbf{v}$ for which various statistical methods to estimate the connectivity matrix have been applied in, for example, climate

modeling (Penland & Sardeshmukh, 1995), gene regulatory networks (Yeung, Tegnér, & Collins, 2002) and spontaneous neuronal activity (Galán, 2008). In this simpler case, the biological learning rule in equation 3.1 would be $\frac{1}{\epsilon} \mathbf{W}_{ij} = \delta[\tilde{\mathbf{v}}_i, \tilde{\mathbf{v}}_j] - \sum_k \mathbf{W}_{ik} \tilde{\mathbf{v}}_k \tilde{\mathbf{v}}_j$ with $\tilde{\mathbf{v}} = l\mathbf{v} - \mathbf{W} \cdot \mathbf{v} * g_i$. The method developed in this letter can be used to extend the inverse modeling approach previously developed in the linear case to models with nonlinear interactions.

One of the main restrictions of this approach is that the dimensionality of the stimuli and that of the network have to be identical. Therefore, the accuracy of this biological mechanism does not match the state-of-the-art machine learning algorithms. This is not necessarily a big issue since a high-dimensional projection of the inputs may be used as preprocessing. From a biological viewpoint and taking the example of vision, this would correspond to the retino-cortical pathway, which is not one-to-one and very redundant. But this limitation also puts forward the need to study a network containing hidden neurons in a similar fashion. Ongoing research is revealing that the mathematical formalism is well suited to extend this approach to the field of reservoir computing (Jaeger & Haas, 2004).

Acknowledgments

We thank Herbert Jaeger for his helpful comments on the manuscript. M.N.G. was partially funded by the Amarsi project (FP7-ICT 24833), ERC advanced grant NerVi 227747, the région PACA, France, and the IP project BrainScaleS 269921.

References

- Abbott, L., & Nelson, S. (2000). Synaptic plasticity: Taming the beast. *Nature Neuroscience*, 3, 1178–1183.
- Ackley, D., Hinton, G., & Sejnowski, T. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, 9(1), 147–169.
- Bar, M. (2009). Predictions: A universal principle in the operation of the human brain. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1521), 1181–1182.
- Berkes, P., Orbán, G., Lengyel, M., & Fiser, J. (2011). Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment. *Science*, 331(6013), 83–87.
- Bi, G., & Poo, M. (1998). Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience*, 18(24), 10464–10472.
- Bienenstock, E., Cooper, L., & Munro, P. (1982). Theory for the development of neuron selectivity: Orientation specificity and binocular interaction in visual cortex. *Journal of Neuroscience*, 2(1), 32–48.

- Bitzer, S., & Kiebel, S. (2012). Recognizing recurrent neural networks (RRNN): Bayesian inference for recurrent neural networks. *Biological Cybernetics*, *106*, 201–217.
- Caporale, N., & Dan, Y. (2008). Spike timing-dependent plasticity: A Hebbian learning rule. *Annu. Rev. Neurosci.*, *31*, 25–46.
- Clark, A. (2012). Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behav. Brain Sci.*, *36*, 181–204.
- Dayan, P., Hinton, G., Neal, R., & Zemel, R. (1995). The Helmholtz machine. *Neural Computation*, *7*(5), 889–904.
- Deneve, S. (2008). Bayesian spiking neurons I: Inference. *Neural Computation*, *20*(1), 91–117.
- Doya, K. (1992). Bifurcations in the learning of recurrent neural networks. In *Proceedings of the IEEE International Symposium on Circuits and Systems* (Vol. 6, pp. 2777–2780). Piscataway, NJ: IEEE.
- Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, *11*(2), 127–138.
- Friston, K., Harrison, L., & Penny, W. (2003). Dynamic causal modelling. *Neuroimage*, *19*(4), 1273–1302.
- Galán, R. (2008). On how network architecture determines the dominant patterns of spontaneous neural activity. *PLoS One*, *3*(5), e2148.
- Galtier, M. (2012). *A mathematical approach to unsupervised learning in recurrent neural networks*. Doctoral dissertation, Mines Paristech/INRIA.
- Galtier, M., & Wainrib, G. (2012). Multiscale analysis of slow-fast neuronal learning models with noise. *Journal of Mathematical Neuroscience*, *2*(1), 13.
- George, D., & Hawkins, J. (2009). Towards a mathematical theory of cortical microcircuits. *PLoS Computational Biology*, *5*(10), e1000532.
- Gerstner, W., & Kistler, W. (2002). *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge: Cambridge University Press.
- Gerstner, W., Ritz, R., & Van Hemmen, J. (1993). Why spikes? Hebbian learning and retrieval of time-resolved excitation patterns. *Biological Cybernetics*, *69*(5), 503–515.
- Gilbert, D., & Wilson, T. (2007). Propection: Experiencing the future. *Science*, *317*(5843), 1351–1354.
- Hennequin, G., Gerstner, W., & Pfister, J. (2010). Stdp in adaptive neurons gives close-to-optimal information transmission. *Frontiers in Computational Neuroscience*, *4*.
- Hinton, G. E. (2009). Deep belief networks. *Scholarpedia*, *4*(4), 5947.
- Izhikevich, E., & Desai, N. (2003). Relating STDP to BCM. *Neural Computation*, *15*(7), 1511–1523.
- Jaeger, H., & Haas, H. (2004). Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, *304*(5667), 78–80.
- Kempter, R., Gerstner, W., & Van Hemmen, J. (1999). Hebbian learning and spiking neurons. *Physical Review E*, *59*(4), 4498.
- Kenet, T., Bibitchkov, D., Tsodyks, M., Grinvald, A., & Arieli, A. (2003). Spontaneously emerging cortical representations of visual attributes. *Nature*, *425*(6961), 954–956.
- Khalil, H., & Grizzle, J. (1992). *Nonlinear systems*. New York: Macmillan.
- Kiebel, S., Daunizeau, J., & Friston, K. (2008). A hierarchy of time-scales and the brain. *PLoS Computational Biology*, *4*(11), e1000209.

- Kim, J., Tsien, R., & Alger, B. (2012). An improved test for detecting multiplicative homeostatic synaptic scaling. *PLoS One*, 7(5), e37364.
- Lazar, A., Pipa, G., & Triesch, J. (2007). Fading memory and time series prediction in recurrent networks with different forms of plasticity. *Neural Networks*, 20(3), 312–322.
- Lazar, A., Pipa, G., & Triesch, J. (2009). SORN: A self-organizing recurrent neural network. *Frontiers in Computational Neuroscience*, 3.
- Lukosevicius, M., & Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3), 127–149.
- Mandic, D., & Chambers, J. (2001). *Recurrent neural networks for prediction: Learning algorithms, architectures and stability*. New York: Wiley.
- Markram, H., Lübke, J., Frotscher, M., & Sakmann, B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic APS and EPSPS. *Science*, 275(5297), 213–215.
- Miller, K. (1996). Synaptic economics: Competition & cooperation in correlation-based synaptic plasticity. *Neuron*, 17, 371–374.
- Miller, K., & MacKay, D. (1994). The role of constraints in Hebbian learning. *Neural Computation*, 6(1), 100–126.
- Oja, E. (1982). Simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15(3), 267–273.
- Pearlmutter, B. (1995). Gradient calculations for dynamic recurrent neural networks: A survey. *IEEE Transactions on Neural Networks*, 6(5), 1212–1228.
- Penland, C., & Sardeshmukh, P. (1995). The optimal growth of tropical sea surface temperature anomalies. *Journal of Climate*, 8(8), 1999–2024.
- Pfister, J., & Gerstner, W. (2006). Triplets of spikes in a model of spike timing-dependent plasticity. *Journal of Neuroscience*, 26(38), 9673–9682.
- Potthast, R., & beim Graben, P. (2009). Inverse problems in neural field theory. *SIAM Journal on Applied Dynamical Systems*, 8, 1405–1433.
- Rao, R., & Ballard, D. (1999). Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2, 79–87.
- Rao, R., & Sejnowski, T. (2001). Spike-timing-dependent Hebbian plasticity as temporal difference learning. *Neural Computation*, 13(10), 2221–2237.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408.
- Sanders, J., & Verhulst, F. (1985). *Averaging Methods in Nonlinear Dynamical Systems*. New York: Springer.
- Schacter, D., Addis, D., & Buckner, R. (2008). Episodic simulation of future events. *Annals of the New York Academy of Sciences*, 1124(1), 39–60.
- Sejnowski, T. (1977). Statistical constraints on synaptic plasticity. *Journal of Theoretical Biology*, 69(2), 385–389.
- Sjöström, J., & Gerstner, W. (2010). Spike-timing dependent plasticity. *Scholarpedia*, 5(2), 1362.
- Song, S., Miller, K., & Abbott, L. (2000). Competitive Hebbian learning through spike-timing-dependent synaptic plasticity. *Nature Neuroscience*, 3, 919–926.
- Sprekeler, H., Michaelis, C., & Wiskott, L. (2007). Slowness: An objective for spike-timing-dependent plasticity? *PLoS Computational Biology*, 3(6), e112.

- Sussillo, D., & Abbott, L. (2009). Generating coherent patterns of activity from chaotic neural networks. *Neuron*, 63(4), 544–557.
- Turrigiano, G. (1999). Homeostatic plasticity in neuronal networks: The more things change, the more they stay the same. *Trends in Neurosciences*, 22(5), 221–227.
- Turrigiano, G., Leslie, K., Desai, N., Rutherford, L., & Nelson, S. (1998). Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature*, 391(6670), 892–896.
- Turrigiano, G., & Nelson, S. (2004). Homeostatic plasticity in the developing nervous system. *Nature Reviews Neuroscience*, 5(2), 97–107.
- Van Rossum, M., Bi, G., & Turrigiano, G. (2000). Stable Hebbian learning from spike timing-dependent plasticity. *Journal of Neuroscience*, 20(23), 8812–8821.
- Williams, R., & Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2), 270–280.
- Williams, R., & Zipser, D. (1995). Gradient-based learning algorithms for recurrent networks and their computational complexity. In Y. Chauvin & D. E. Rumelhart (Eds.), *Back-Propagation: Theory, Architectures and Applications* (pp. 433–486). Hillsdale, NJ: Psychology Press.
- Yeung, M., Tegnér, J., & Collins, J. (2002). Reverse engineering gene networks using singular value decomposition and robust regression. *Proceedings of the National Academy of Sciences*, 99(9), 6163–6168.
- Yoshioka, M., Scarpetta, S., & Marinaro, M. (2007). Spatiotemporal learning in analog neural networks using spike-timing-dependent synaptic plasticity. *Physical Review E*, 75(5), 051917.
- Zheng, P., Dimitrakakis, C., & Triesch, J. (2013). Network self-organization explains the statistics and dynamics of synaptic connection strengths in cortex. *PLoS Computational Biology*, 9(1).

Received February 5, 2013; accepted May 23, 2013.