ARTICLE Communicated by Naftali Tishby

Spiking Neurons Can Learn to Solve Information Bottleneck Problems and Extract Independent Components

Stefan Klampfl
klampfl@igi.tugraz.at
Robert Legenstein
legi@igi.tugraz.at
Wolfgang Maass
maass@igi.tugraz.at
Institute for Theoretical Computer Science, Graz University of Technology, A-8010 Graz, Austria

Independent component analysis (or blind source separation) is assumed to be an essential component of sensory processing in the brain and could provide a less redundant representation about the external world. Another powerful processing strategy is the optimization of internal representations according to the information bottleneck method. This method would allow extracting preferentially those components from high-dimensional sensory input streams that are related to other information sources, such as internal predictions or proprioceptive feedback. However, there exists a lack of models that could explain how spiking neurons could learn to execute either of these two processing strategies. We show in this article how stochastically spiking neurons with refractoriness could in principle learn in an unsupervised manner to carry out both information bottleneck optimization and the extraction of independent components. We derive suitable learning rules, which extend the well-known BCM rule, from abstract information optimization principles. These rules will simultaneously keep the firing rate of the neuron within a biologically realistic range.

1 Introduction

The information bottleneck (IB) approach and independent component analysis (ICA) have both attracted substantial interest as general principles for unsupervised learning (Tishby, Pereira, & Bialek, 1999; Hyvärinen, Karhunen, & Oja, 2001). A hope has been that they might also help us to understand strategies for unsupervised learning in biological systems. However, it has turned out to be quite difficult to establish links between known learning algorithms that have been derived from these general principles and learning rules that could possibly be implemented by synaptic plasticity of a spiking neuron. Fortunately, in a simpler context, a direct

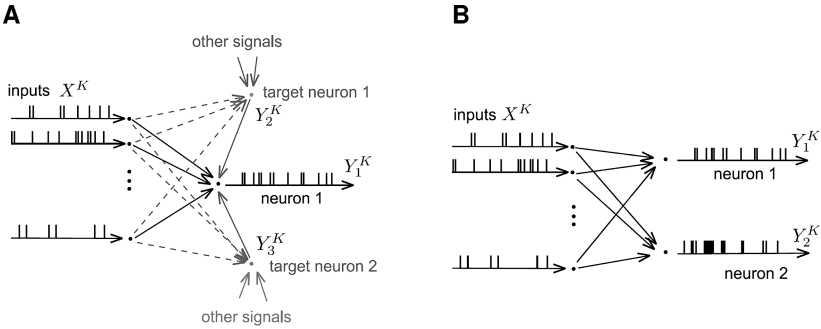

Figure 1: Learning situations analyzed in this article. (A) In an information bottleneck task, the learning neuron (neuron 1) wants to maximize the mutual information between its output Y_1^K and the activity of one or several target neurons $Y_2^K, Y_3^K, \ldots$ (which can be functions of the inputs X^K or other external signals), while at the same time keeping the mutual information between the inputs X^K and the output Y_1^K as low as possible (and its firing rate within a desired range). Thus, the neuron should learn to extract from its high-dimensional input those aspects that are related to these target signals. This setup is discussed in sections 3 to 5. (B) Two neurons receiving the same inputs X^K from a common set of presynaptic neurons both learn to maximize information transmission and simultaneously keep their outputs Y_1^K and Y_2^K statistically independent. Such extraction of independent components from the input is described in section 6.

link between an abstract information-theoretic optimization goal and a rule for synaptic plasticity has recently been established (Toyoizumi, Pfister, Aihara, & Gerstner, 2005). The resulting rule for the change of synaptic weights in Toyoizumi et al. maximizes the mutual information between pre- and postsynaptic spike trains, under the constraint that the postsynaptic firing rate stays close to some target firing rate. We show in this article that this approach can be extended to situations where simultaneously, the mutual information between the postsynaptic spike train of the neuron and other signals (such as for example the spike trains of other neurons) has to be minimized (see Figure 1). This opens the door to the exploration of learning rules for IB analysis and independent component extraction with spiking neurons that would be optimal from a theoretical perspective.

The IB method (Tishby et al., 1999) is a recently developed information-theoretic approach that tries to compress information about a data variable X, while at the same time preserving as much information as possible about a relevant (target) variable Y; that is, it aims at selecting a compact representation $\tilde{X}$ of the data X. Information that X provides about Y is squeezed through a "bottleneck" of the compressed variable $\tilde{X}$. There is a trade-off between compression (low mutual information between $\tilde{X}$ and X) and preserving relevant information (high mutual information between

$\tilde{X}$ and Y). That is, one usually maximizes $-I(\tilde{\mathbf{X}}; \mathbf{X}) + \beta I(\tilde{\mathbf{X}}; \mathbf{Y})$ with some trade-off parameter β, where $I(\mathbf{U}; \mathbf{V})$ denotes the mutual information between random variables U and V. In this approach, we interpret the input spike trains X^K to a neuron as the data X, the output spike train Y_1^K as the compact representation $\tilde{X}$ of X, and the relevant variable Y as a 'target' spike train Y_2^K (or several target spike trains $Y_2^K, Y_3^K, \ldots$) (see Figure 1A).

Independent component analysis (Hyvärinen et al., 2001) is another well-known statistical technique for decomposing complex data into statistically independent parts, thereby providing a less redundant representation. In our approach, we minimize the mutual information between the output spike trains Y_1^K and Y_2^K of two neurons receiving the same input X^K. Simultaneously we want both neurons to extract meaningful information by maximizing the mutual information between the inputs X^K and the output spike train Y_i^K of both neurons $i = 1, 2$ (see Figure 1B).

We review in section 2 the neuron model and learning rule from Toyoizumi et al. (2005). We show in section 3 how this learning rule can be extended so that it not only maximizes mutual information with some given spike trains and keeps the output firing rate within a desired range, but simultaneously minimizes mutual information with other spike trains or other time-varying signals. In section 4 we analyze the learning strategies of the resulting learning rules and relate them to the classical (Bienenstock, Cooper, & Munro, 1982) and generalized (Toyoizumi et al., 2005) Bienenstock-Cooper-Munro (BCM) rule. Applications to concrete IB tasks are discussed in section 5. Because of the many different types of target signals that might be relevant in a biological system, we do not model the way that such target signals might affect the synapse or neuron under consideration, but rather use it as an abstract signal in the learning rule. In section 6, we show that a modification of this learning rule allows a spiking neuron to extract information from its input spike trains that is independent from the information extracted by another neuron. Moreover, we present an approximation of the learning rule that indicates how the rule might possibly be implemented in a biologically realistic circuit.

2 Neuron Model and a Basic Learning Rule

We use the neuron model from Toyoizumi et al. (2005), which is a stochastically spiking neuron model with refractoriness, where the probability of firing in each time step depends on the current membrane potential and the time since the last output spike. It is convenient to formulate the model in discrete time with step size Δt. The total membrane potential of a neuron i at time step $t^k = k\Delta t$ is given by

$$u_i(t^k) = u_r + \sum_{j=1}^{N} \sum_{n=1}^{k} w_{ij} \epsilon(t^k - t^n) x_j^n, \tag{2.1}$$

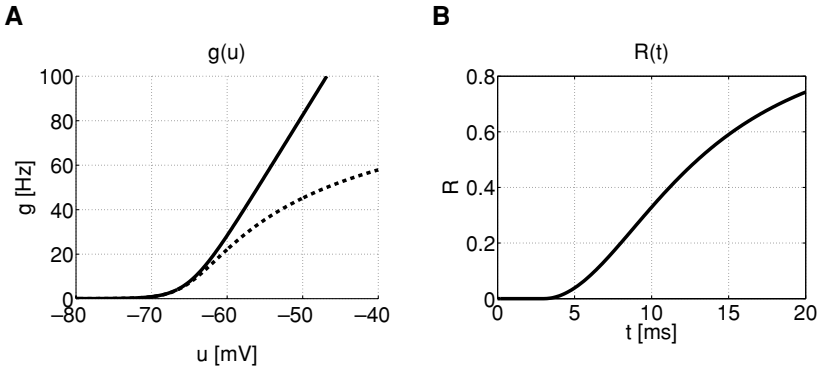

Figure 2: Characteristic functions of the neuron model (see equation 2.2). (A) Gain function $g(u)$ (solid; see equation 2.4) and $g_{alt}(u)$ (dashed; see equation 2.5) as a function of the membrane potential u (plotted for $u_0 = -65$ mV, $\Delta u = 2$ mV, $r_0 = 11$ Hz, $g_{\max} = 100$ Hz). (B) Refractory variable $R(t)$ as a function of the time $t - \hat{t}$ since the last postsynaptic spike (plotted for $\hat{t} = 0$, and for an absolute refractory period $\tau_{abs} = 3$ ms, relative refractory time $\tau_{refr} = 10$ ms).

where $u_r = -70$ mV is the resting potential and w_{ij} is the weight of the synapse from the presynaptic neuron j ($j = 1, \ldots, N$). An input spike train at synapse j is described up to the kth time step by a sequence $X_j^k = (x_j^1, x_j^2, \ldots, x_j^k)$ of zeros (no spike) and ones (spike). Each presynaptic spike at time t^n ($x_j^n = 1$) evokes a postsynaptic potential (PSP) with exponential by decaying time course $\epsilon(t - t^n) = U_{PSP} e^{-(t-t_n)/\tau_m}$ for $t \geq t_n$ with time constant $\tau_m = 10$ ms and PSP amplitude $U_{PSP} = 1$ mV. The probability ρ_i^k of the firing of neuron i at time step t^k is then given by

$$\rho_i^k = 1 - \exp[-g(u_i(t^k)) R_i(t^k) \Delta t] \approx g(u_i(t^k)) R_i(t^k) \Delta t, \tag{2.2}$$

where the refractory variable,

$$R_i(t) = \frac{(t - \hat{t}_i - \tau_{abs})^2}{\tau_{refr}^2 + (t - \hat{t}_i - \tau_{abs})^2} \Theta(t - \hat{t}_i - \tau_{abs}), \tag{2.3}$$

assumes values in $[0, 1]$ and depends on the last firing time $\hat{t}_i$ of neuron i (see Figure 2B). The absolute refractory period $\tau_{abs} = 3$ ms is the time period after a firing during which no spike can occur; in the relative refractory time $\tau_{refr} = 10$ ms, it is hard, but not impossible, to emit an action potential. The

Heaviside step function Θ takes a value of 1 for nonnegative arguments and 0 otherwise. The gain function,

$$g(u) = r_0 \log\left\{1 + \exp\left[\frac{u - u_0}{\Delta u}\right]\right\}, \tag{2.4}$$

is a smooth increasing function of the membrane potential u (see Figure 2A; $u_0 = -65$ mV, $\Delta u = 2$ mV, $r_0 = 11$ Hz). The approximation in equation 2.2 is valid for sufficiently small Δt ($\rho_i^k \ll 1$). The function $g(u)$ implements a stochastic threshold around u_0; below u_0, it goes to 0, and above u_0, it increases linearly with the membrane potential (with slope $r_0/\Delta u$). Note that due to refractoriness, the output firing rate of the neuron cannot be made arbitrarily high. For a neuron model without refractoriness (see section 3.2), one has to formalize an upper bound on the firing rate of the neuron in a different way. For that we choose, as in Toyoizumi et al. (2005) an alternative gain function,

$$g_{alt}(u) = \left[\frac{1}{g_{\max}} + \frac{1}{g(u)}\right]^{-1}, \tag{2.5}$$

with a maximum rate of $g_{\max} = 100$ Hz (see Figure 2A).

This model from Toyoizumi et al. (2005) is a special case of the spike response model, and with a refractory variable $R(t)$ that depends on only the time since the last postsynaptic event, it has renewal properties (Gerstner & Kistler, 2002). The output of neuron i at the kth time step is denoted by a variable y_i^k, which assumes the value of 1 if a postsynaptic spike occurs and 0 otherwise. A specific spike train up to the kth time step is written as $Y_i^k = (y_i^1, y_i^2, \ldots, y_i^k)$.

The information transmission between an ensemble of input spike trains $\mathbf{X}^K$ and the output spike train $\mathbf{Y}_i^K$ of total duration $K\Delta t$ can be quantified by the mutual information[1] (Cover & Thomas, 1991)

$$I\left(\mathbf{X}^K; \mathbf{Y}_i^K\right) = \sum_{X^K, Y_i^K} P\left(X^K, Y_i^K\right) \log \frac{P\left(Y_i^K \mid X^K\right)}{P\left(Y_i^K\right)}. \tag{2.6}$$

The idea in Toyoizumi et al. (2005) was to maximize the quantity $I(\mathbf{X}^K; \mathbf{Y}_i^K) - \gamma D_{KL}(P(Y_i^K)||\tilde{P}(Y_i^K))$, where

$$D_{KL}\left(P\left(Y_i^K\right)||\tilde{P}\left(Y_i^K\right)\right) = \sum_{Y_i^K} P\left(Y_i^K\right) \log \frac{P\left(Y_i^K\right)}{\tilde{P}\left(Y_i^K\right)} \tag{2.7}$$

[1] We use boldface letters ($\mathbf{X}^k$) to distinguish random variables from specific realizations (X^k).

denotes the Kullback-Leibler divergence (Cover & Thomas, 1991) between the actual distribution $P(Y_i^K)$ and a given target distribution $\tilde{P}(Y_i^K)$. The inclusion of this second term imposes the additional constraint that the firing statistics $P(Y_i)$ of the neuron i should stay as close as possible to a target distribution $\tilde{P}(Y_i)$. This distribution was chosen in Toyoizumi et al. (2005) to yield a constant target firing rate $\tilde{g}$. An online learning rule performing gradient ascent on this quantity was derived in Toyoizumi et al. for the weight w_{ij} of neuron i,

$$\frac{dw_{ij}(t)}{dt} = \alpha C_{ij}(t) B_i^{post}(t, \gamma), \tag{2.8}$$

which consists of the correlation term C_{ij} and the postsynaptic term B_i^{post} (Toyoizumi et al., 2005). The term C_{ij} measures coincidences between postsynaptic spikes at neuron i and PSPs generated by presynaptic action potentials arriving at synapse j,

$$\frac{dC_{ij}(t)}{dt} = -\frac{C_{ij}(t)}{\tau_C} + \sum_l \epsilon\left(t - t_j^{(l)}\right) \frac{g'(u_i(t))}{g(u_i(t))} [\delta(t - \hat{t}_i) - g(u_i(t)) R_i(t)], \tag{2.9}$$

with time constant $\tau_C = 1$ s, $\delta(t)$ being the Dirac-δ function and $g'(u_i(t))$ denoting the derivative of g with respect to u. The term

$$\begin{aligned} B_i^{post}(t, \gamma) = \delta(t - \hat{t}_i) \log \left[\frac{g(u_i(t))}{\bar{g}_i(t)} \left(\frac{\tilde{g}}{\bar{g}_i(t)} \right)^{\gamma} \right] \\ - R_i(t)[g(u_i(t)) - (1 + \gamma)\bar{g}_i(t) + \gamma \tilde{g}] \end{aligned} \tag{2.10}$$

compares the current firing rate $g(u_i(t))$ with its average firing rate[2] $\bar{g}_i(t)$, and simultaneously the running average $\bar{g}_i(t)$ with the constant target rate $\tilde{g}$. The second argument indicates that this term also depends on the optimization parameter γ.

3 Information-Theoretic Principles Provide Learning Rules for More Complex Learning Goals

We extend the learning rule presented in the previous section to a more complex scenario, where the mutual information between the output spike train Y_1^K of the learning neuron (neuron 1) and some target spike trains Y_l^K ($l > 1$) has to be maximized, while simultaneously minimizing the

[2]The rate $\bar{g}_i(t) = \langle g(u_i(t)) \rangle_{\mathbf{x}|Y_i}$ denotes an expectation of the firing rate over the input distribution given the postsynaptic history and is implemented as a running average with an exponential time window (with a time constant of 10 s).

mutual information between the inputs X^K and Y_1^K. Obviously this is the generic information bottleneck (IB) scenario applied to spiking neurons (see Figure 1A). A learning rule for extracting independent components with spiking neurons (see section 6) can be derived in a similar manner by switching the signs of the first two terms in the objective function, 3.1. In this section, we derive two online learning rules, a spike-based and a simplified rate-based learning rule, for this IB task.

3.1 Spike-Based Learning Rule. For simplicity, we consider the case of an IB optimization for only one target spike train Y_2^K and derive an update rule for the synaptic weights w_{1j} of neuron 1. The quantity to maximize is therefore

$$L = -I\big(\mathbf{X}^K; \mathbf{Y}_1^K\big) + \beta I\big(\mathbf{Y}_1^K; \mathbf{Y}_2^K\big) - \gamma D_{KL}\big(P\big(Y_1^K\big) \| \tilde{P}\big(Y_1^K\big)\big), \tag{3.1}$$

where β and γ are optimization constants. To maximize this objective function, we derive the weight change Δw_{1j}^k during the kth time step by gradient ascent on equation 3.1, assuming that the weights w_{1j} can change between some bounds $0 \le w_{1j} \le w_{\max}$ (we assume $w_{\max} = 1$ throughout this article).

Now we have to calculate the gradient of L with respect to the weights of the learning neuron, w_{1j}. Note that all three terms of equation 3.1 implicitly depend on w_{1j} because the output distribution $P(Y_1^K)$ changes if we modify the weights w_{1j}. Since the first and the last term of the equation have already been considered (up to the sign) in Toyoizumi et al. (2005), we will concentrate here on the middle term,

$$L_{12} := \beta I\big(\mathbf{Y}_1^K; \mathbf{Y}_2^K\big) = \beta \sum_{Y_1^K, Y_2^K} P\big(Y_1^K, Y_2^K\big) \log \frac{P\big(Y_1^K, Y_2^K\big)}{P\big(Y_1^K\big) P\big(Y_2^K\big)}, \tag{3.2}$$

and denote the contribution of the gradient of L_{12} to the total weight change Δw_{1j}^k in the kth time step by $\Delta \tilde{w}_{1j}^k$. One can proceed here also similarly as in Toyoizumi et al. (2005), but some additional aspects have to be taken into account.

Up to now, we have considered only spike trains of length $K\Delta t$ in equations 3.1 and 3.2. In order to get an expression for the weight change in a specific time step k, $\Delta \tilde{w}_{1j}^k$, we have to calculate the contribution of this time bin to the objective function L_{12}. According to the chain rule of information theory (Cover & Thomas, 1991), we can write the probabilities $P(Y_i^K)$ and $P(Y_1^K, Y_2^K)$ occurring in equation 3.2 as products over the probability distributions of individual time bins given the corresponding postsynaptic histories: $P(Y_i^K) = \prod_{k=1}^K P(y_i^k | Y_i^{k-1})$ and $P(Y_1^K, Y_2^K) = \prod_{k=1}^K P(y_1^k, y_2^k | Y_1^{k-1}, Y_2^{k-1})$. As a consequence, we can express the middle term L_{12} in equation 3.1 as

a sum over the contributions of individual time bins, $L_{12} = \sum_{k=1}^{K} \Delta L_{12}^k$, with

$$\Delta L_{12}^k = \left\langle \beta \log \frac{P\left(y_1^k, y_2^k | Y_1^{k-1}, Y_2^{k-1}\right)}{P\left(y_1^k | Y_1^{k-1}\right) P\left(y_2^k | Y_2^{k-1}\right)} \right\rangle_{\mathbf{X}^k, \mathbf{Y}_1^k, \mathbf{Y}_2^k} . \tag{3.3}$$

Hence, ΔL_{12}^k reflects the statistical dependence between the binary variables y_1^k and y_2^k, given the postsynaptic histories Y_1^{k-1} and Y_2^{k-1}. An evaluation of the probabilities used in equation 3.3 can be found in section A.1.

The weight change $\Delta \tilde{w}_{1j}^k$ in each time step k is then proportional to the gradient of this expression ΔL_{12}^k with respect to the weights w_{1j},

$$\Delta \tilde{w}_{1j}^k = \alpha \frac{\partial \Delta L_{12}^k}{\partial w_{1j}}, \tag{3.4}$$

where $\alpha > 0$ denotes the learning rate. Under the assumption of small Δt (we choose $\Delta t = 1$ ms throughout the simulations), evaluation of the gradient 3.4 yields (for a detailed derivation, see section A.2)

$$\Delta \tilde{w}_{1j}^k = \alpha \left\langle C_{1j}^k \beta F_{12}^k \right\rangle_{\mathbf{X}^k, \mathbf{Y}_1^k, \mathbf{Y}_2^k} . \tag{3.5}$$

The term in the angled brackets in equation 3.5 consists of two factors. The first factor is a correlation term C_{1j}^k as in Toyoizumi et al. (2005),

$$C_{1j}^k = C_{1j}^{k-1} \left(1 - \frac{\Delta t}{\tau_C}\right) + \sum_{n=1}^{k} \epsilon(t^k - t^n) x_j^n \frac{g'(u_1(t^k))}{g(u_1(t^k))} \left[y_1^k - \rho_1^k\right], \tag{3.6}$$

which counts the coincidences between postsynaptic spikes ($y_1^k = 1$) and the time course of PSPs generated by presynaptic spikes ($x_j^n = 1$) in an exponential time window with time constant $\tau_C = 1$ s. The term $g'(u_i(t))$ denotes the derivative of $g(u)$ with respect to u and measures the sensitivity of the neuron for changes in the membrane potential.

The second factor measures the momentary statistical dependence between the outputs y_1^k and y_2^k,

$$\begin{aligned} F_{12}^k = {} & y_1^k y_2^k \log \frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)\bar{g}_2(t^k)} - y_1^k \left(1 - y_2^k\right) R_2(t^k) \Delta t \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k)\right] - \\ & - \left(1 - y_1^k\right) y_2^k R_1(t^k) \Delta t \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k)\right] + \\ & + \left(1 - y_1^k\right)\left(1 - y_2^k\right) R_1(t^k) R_2(t^k) (\Delta t)^2 \left[\bar{g}_{12}(t^k) - \bar{g}_1(t^k)\bar{g}_2(t^k)\right]. \end{aligned} \tag{3.7}$$

Here, $\bar{g}_i(t^k) = \langle g(u_i(t^k)) \rangle_{\mathbf{X}^k | Y_i^{k-1}}$ denotes the average firing rate of neuron i, and $\bar{g}_{12}(t^k) = \langle g(u_1(t^k)) g(u_2(t^k)) \rangle_{\mathbf{X}^k | Y_1^{k-1}, Y_2^{k-1}}$ denotes the average product of firing rates of both neurons. Both quantities are implemented online as

running exponential averages with a time constant of 10 s. Note that F_{12}^k depends directly on the relationship between the joint probability of firing, which is represented by $\bar{g}_{12}(t^k)$, and the product of the individual firing probabilities given by $\bar{g}_1(t^k)\bar{g}_2(t^k)$.

Yet the weight change (see equation 3.5) is still given by an average over the distributions of spike trains X^k, Y_1^k, Y_2^k up to time step k and cannot be implemented as an online rule in this way. However, under the assumption of a small learning rate α, we can approximate the expectation $\langle\cdot\rangle_{\mathbf{X}^k,\mathbf{Y}_1^k,\mathbf{Y}_2^k}$ in equation 3.3 by averaging over a single long trial. Considering now all three terms in equation 3.1, we finally arrive at an online rule for maximizing L:

$$\frac{\Delta w_{1j}^k}{\Delta t} = -\alpha C_{1j}^k \left[B_1^k(-\gamma) - \beta \Delta t B_{12}^k \right]. \tag{3.8}$$

The term C_{1j}^k, equation 3.6, is sensitive to correlations between the output of the neuron and its presynaptic input at synapse j (correlation term) and the terms B_1^k and B_{12}^k characterize the postsynaptic state of the neuron (postsynaptic terms). Typical time courses of these terms are shown in Figure 3.

This learning rule is thus an extension to the generalized BCM rule for spiking neurons (Toyoizumi et al., 2005). The term $B_1^k(-\gamma)$ is given by

$$\begin{aligned} B_1^k(-\gamma) = {} & \frac{y_1^k}{\Delta t} \log \left[\frac{g(u_1(t^k))}{\bar{g}_1(t^k)} \left(\frac{\bar{g}_1(t^k)}{\tilde{g}} \right)^{\gamma} \right] \\ & - \left(1 - y_1^k\right) R_1(t^k) \left[g(u_1(t^k)) - (1-\gamma)\bar{g}_1(t^k) - \gamma \tilde{g} \right] \end{aligned} \tag{3.9}$$

and has been described together with C_{1j}^k in the previous section (these terms are discrete time versions of $C_{1j}(t)$ in equation 2.9 and $B_1^{post}(t,\gamma)$ in equation 2.10, respectively).[3] Our learning rule contains an extra term $B_{12}^k = F_{12}^k/(\Delta t)^2$ that is sensitive to the statistical dependence between the output spike train of the neuron and the target signal. It is given by

$$\begin{aligned} B_{12}^k = {} & \frac{y_1^k y_2^k}{(\Delta t)^2} \log \frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)\bar{g}_2(t^k)} - \frac{y_1^k}{\Delta t}\left(1 - y_2^k\right) R_2(t^k) \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k) \right] \\ & - \frac{y_2^k}{\Delta t}\left(1 - y_1^k\right) R_1(t^k) \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k) \right] \\ & + \left(1 - y_1^k\right)\left(1 - y_2^k\right) R_1(t^k) R_2(t^k) \left[\bar{g}_{12}(t^k) - \bar{g}_1(t^k)\bar{g}_2(t^k) \right]. \end{aligned} \tag{3.10}$$

[3] The argument of B_1^k, $-\gamma$, is different from the second argument in equation 2.8, γ, because the term $I(\mathbf{X}^K;\mathbf{Y}_1^K)$ enters the objective function, equation 3.1, with a different sign, whereas the constraint with the KL divergence enters with the same sign.

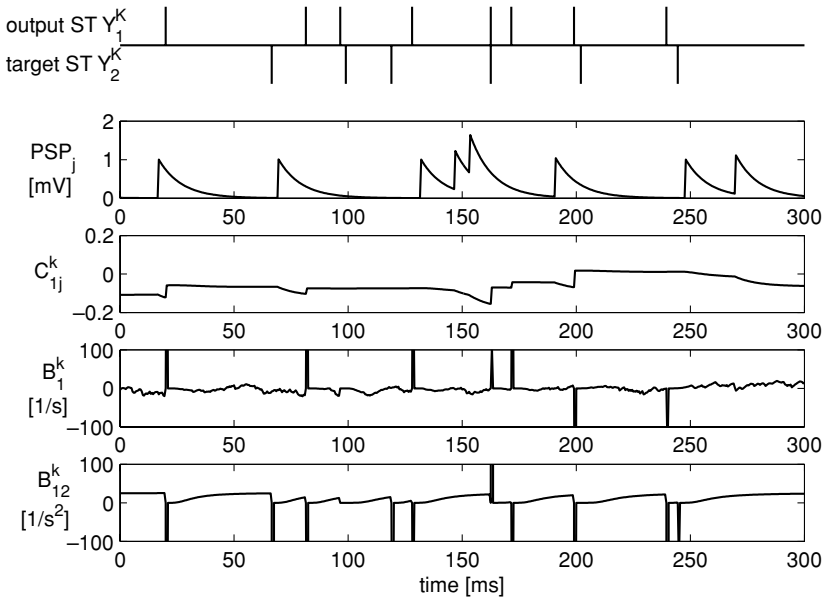

Figure 3: Visualization of the impact of the three terms in learning rule 3.8. From top to bottom: instances of an output spike train Y_1^K and a target spike train Y_2^K of length 300 ms, the time course of the PSP $\sum_n \epsilon(t^k - t^n)x_j^n$ during that time at a single synapse j, of the correlation term C_{1j}^k, equation 3.6, for the input at this synapse j, and of the postsynaptic terms B_1^k, equation 3.9, and B_{12}^k, equation 3.10. While the term B_1^k has peaks only for spikes in the output spike train Y_1^K, the term B_{12}^k has additional peaks at times of action potentials in the target spike train Y_2^K. Their amplitude and sign depend on the momentary statistical dependence of the recent histories of both spike trains.

This term basically compares the average product of firing rates $\bar{g}_{12}$ (which corresponds to the joint probability of spiking) with the product of average firing rates $\bar{g}_1\bar{g}_2$ (representing the probability of independent spiking). In this way, it measures the momentary mutual information between the output of the neuron and the target spike train. B_{12}^k consists of four terms—one for each firing state of the two neurons. The first term produces a peak when both neurons fire in the same time step (see the positive peak in the bottom trace of Figure 3). The second and third terms result in peaks when only one neuron is active (see the negative peaks in Figure 3). Note that these terms additionally depend on the refractory state of the other neuron: in case of almost coincident spikes, the second event has no influence due to the refractoriness of the other neuron, which has spiked just before. In other words, the learning rule distinguishes between the two cases whether a neuron does not spike because of refractoriness or because of a low firing

rate; only in the latter case does this have an influence. Finally, the fourth term of equation 3.10 results in small fluctuations of B_{12}^k in between firing events and depends on the refractoriness of both neurons and the difference between $\bar{g}_{12}$ and $\bar{g}_1\bar{g}_2$. Note, however, that the actual sign of B_{12}^k depends on the recent firing histories of the two neurons; for example, if the two spike trains have recently been correlated, $\bar{g}_{12}$ is larger than $\bar{g}_1\bar{g}_2$ (as is the case in Figure 3). Furthermore, the actual weight change depends according to equation 3.8 on an interplay between both the postsynaptic terms B_1^k and B_{12}^k and the correlation term C_{1j}^k.

Note that during the duration of an excitatory postsynaptic potential (EPSP) caused by an input spike there is an increased probability of generating an output spike (Kempter, Gerstner, & van Hemmen, 1999). If two neurons share the same input, they will then have a correlated spiking probability. This effect of the EPSP is captured by the correlation term C_{ij}^k, equation 3.6, which is sensitive to correlations between input and output spikes. It is increased if an input spike is accompanied by an output spike during the duration of the EPSP caused by that input spike. Note that the term B_{12}^k in equation C.3 is multiplied with the term C_{ij}^k in the actual learning rule, 3.8. The term B_{12}^k on its own is sensitive only to the mutual information between the binary variables y_1^k and y_2^k given their histories (estimated by the running averages of firing rates), regardless of how they have been generated.

In equation 3.8, in order to compensate the effect of a small Δt, the constant β has to be large enough for the term B_{12}^k to have an influence on the weight change. In the limit $\Delta t \to 0$, the value of β approaches infinity. One can overcome this problem by using instead of equation 3.1 an alternative objective function that includes the information rate $I(\mathbf{Y}_1^K; \mathbf{Y}_2^K)/\Delta t$ instead of the mutual information $I(\mathbf{Y}_1^K; \mathbf{Y}_2^K)$. In this case, the Δt on the right-hand side of equation 3.1 would cancel out, and the trade-off parameter β would become a constant of dimension s (time). However, in the following, we use our original objective function, equation 3.1, and analyze weight changes in discrete time with a fixed Δt.

3.2 Simplified Rate-Based Learning Rule. To gain more insight into the learning rule, equation 3.8, we consider a simplified neuron model without refractoriness. The dynamics of this model are governed by equations 2.1 and 2.2 with $R_i(t) = 1$ (i.e., $\tau_{abs} = \tau_{refr} = 0$ ms). As in Toyoizumi et al. (2005), we use $g_{alt}(u)$, equation 2.5, for the gain function in order to pose an upper limit on the postsynaptic firing rate in the absence of refractoriness. In this rate model, the probability of spiking is independent of the postsynaptic history. Since there is no refractoriness, the postsynaptic rate ν_1^k at time t^k is given directly by the current value of $g_{alt}(u_1(t^k))$. Toyoizumi et al. (2005) showed that the update rule, equation 3.8, resembles the BCM rule (Bienenstock et al., 1982). Since we want to maximize here a different objective

function, equation 3.1, we expect an "anti-Hebbian BCM" rule with an additional term accounting for statistical dependencies between Y_1^K and Y_2^K.

With these simplifying assumptions above, the learning rule, equation 3.8, reduces to the following learning rule for a rate model (see section A.4 for a detailed derivation):

$$\frac{\Delta w_{1j}^k}{\Delta t} = -\alpha \nu_j^{pre,k} f(\nu_1^k) \left\{ \log \left[\frac{\nu_1^k}{\bar{\nu}_1^k} \left(\frac{\bar{\nu}_1^k}{\tilde{g}} \right)^\gamma \right] \right.$$
$$\left. -\beta \Delta t \left(\nu_2^k \log \left[\frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k \bar{\nu}_2^k} \right] - \bar{\nu}_2^k \left[\frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k \bar{\nu}_2^k} - 1 \right] \right) \right\}, \tag{3.11}$$

where the presynaptic rate at synapse j at time t^k is denoted by $\nu_j^{pre,k} = a \sum_{n=1}^k \epsilon(t^k - t^n) x_j^n$ with a in units $(\mathrm{Vs})^{-1}$. The values $\bar{\nu}_1^k$, $\bar{\nu}_2^k$, and $\bar{\nu}_{12}^k$ are running averages of the output rate ν_1^k, the rate of the target signal ν_2^k, and of the product of these values, $\nu_1^k \nu_2^k$, respectively. The function $f(\nu_1^k) = g'_{alt}(g_{alt}^{-1}(\nu_1^k))/a$ is proportional to the derivative of g_{alt} with respect to u, evaluated at the current membrane potential. It measures the momentary sensitivity of the output rate for changes of the membrane potential (see Figure 4A). This weight change approximates a gradient ascent for the objective function 3.1. The approximation is valid for small Δt (we choose $\Delta t = 1$ ms in the simulations). Note that the factor β has to compensate a small Δt so that the second term has influence on the weight change. A detailed discussion of this rule is given in section 4.

4 Analysis of the Resulting Learning Rules

In the previous section, we derived learning rules that minimize the information transmission of a neuron while simultaneously keeping the mutual information between the output and target spike trains as high as possible. Additionally we have imposed the constraint that the firing rate of the learning neuron should stay close to a constant target firing rate. These rules are summarized in Tables 1 and 2. The spike-based rule has been derived for a stochastically spiking neuron model with refractoriness; for the rate-based rule, we considered a simplified neuron model without refractoriness, as in Toyoizumi et al. (2005). In this section, we interpret these rules and show how they relate to the classical BCM rule and to the generalized rule presented in Toyoizumi et al.

4.1 Comparison of the Simplified Rule with the Spike-Based Rule. Comparing the spike-based and rate-based learning rules (respectively, equations 3.8 and 3.11), we find that for both rules, the weight change depends on the correlation of pre- and postsynaptic activity, via either the correlation term C_{ij}^k or the Hebbian term $\nu_j^{pre,k} f(\nu_1^k)$. In both cases, the influence

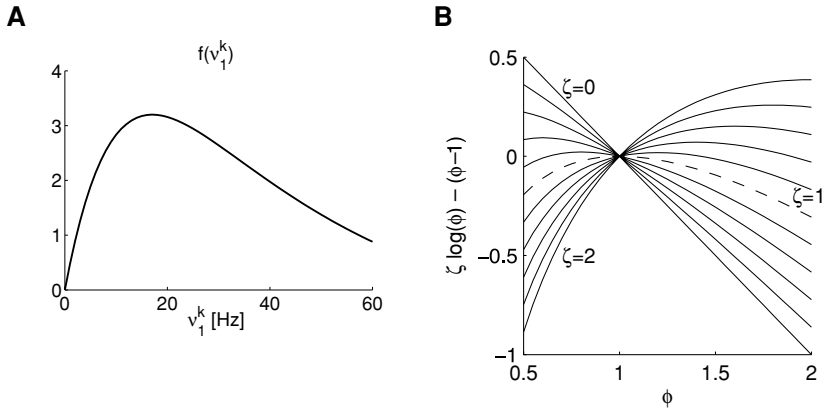

Figure 4: Influence of specific terms of the rate-based rule, equation 3.11. (A) Sensitivity function $f(\nu_1^k) = g'_{alt}(g_{alt}^{-1}(\nu_1^k))/a$ as a function of the postsynaptic firing rate ν_1^k with $a = 10^3$ $(\mathrm{Vs})^{-1}$. (B) The influence of correlations between ν_1^k and ν_2^k (measured by $\phi = \bar{\nu}_{12}^k/(\bar{\nu}_1^k \bar{\nu}_2^k)$. See equation 4.1 on the simplified rule for different ratios $\zeta = \nu_2^k/\bar{\nu}_2^k$. The plotted function captures the weight changes induced by the second line of equation 3.11. This function is zero for uncorrelated signals ($\phi = 1$). For correlated signals ($\phi > 1$), firing rates ν_2^k sufficiently above mean induce LTP. For anticorrelated signals ($\phi < 1$), firing rates ν_2^k sufficiently below mean induce LTP.

of the postsynaptic activity on the weight change depends also on the current sensitivity of the neuron, which is expressed through the derivative of g with respect to u (see the plot of $f(\nu_1^k)$ in Figure 4A). Furthermore, the first term in the curly brackets of equation 3.11 corresponds to the first term of $B_1^k(-\gamma)$, equation 2.10. This classical BCM term is responsible for regulating the information transmission of the neuron and for the homeostatic process that tries to maintain a constant target firing rate via a sliding threshold of the postsynaptic activity, $\bar{\nu}_1^k$ (Toyoizumi et al., 2005). However, this term is augmented by an expression sensitive to the statistical dependence between the output of the neuron and the target signal—second line in equation 3.11 and B_{12}^k (equation 3.10). Here, the second line in equation 3.11 corresponds to the first two terms in equation 3.10. All the other terms in B_1^k and B_{12}^k can be neglected in the rate-based rule for small Δt (see the derivation in section A.4 and analogous derivation in Toyoizumi et al., 2005).

4.2 Interpretation of the Simplified Rule. To gain a better understanding of the derived learning rule, we analyze the rate-based rule, equation 3.11, in more detail. The prefactor of this rule, $\nu_j^{pre,k} f(\nu_1^k)$, is a nonlinear Hebbian term because the weight change does not depend on the postsynaptic activity ν_1^k directly, but only via the nonlinear function f. It is

Table 1: Summary of the Spike-Based Learning Rule for the Information Bottleneck Task Derived in Section 3.1.

Performing gradient ascent on L, equation 3.1, yields an online learning rule for the weights of neuron 1, w_{1j}. The weight change Δw_{1j}^k at time $t^k = k\Delta t$ is given by

$$\frac{\Delta w_{1j}^k}{\Delta t} = -\alpha C_{1j}^k \left[B_1^k(-\gamma) - \beta \Delta t B_{12}^k \right] \tag{3.8}$$

with a learning rate $\alpha > 0$ and optimization parameters β and γ with values > 0.

The correlation term C_{1j}^k measures coincidences between postsynaptic spikes at neuron 1 and PSPs generated by presynaptic action potentials arriving at synapse j:

$$C_{1j}^k = C_{1j}^{k-1}\left(1 - \frac{\Delta t}{\tau_C}\right) + \sum_{n=1}^{k} \epsilon(t^k - t^n) x_j^n \frac{g'(u_1(t^k))}{g(u_1(t^k))} \left[y_1^k - \rho_1^k \right] \tag{3.6}$$

τ_C	time constant of exponential correlation window
x_j^n	binary variable indicating a presynaptic spike at synapse j in the nth time step
y_1^k	binary variable indicating an output spike of neuron 1 in the kth time step
ρ_1^k	firing probability of neuron 1 in the kth time step, equation 2.2
$\epsilon(s)$	time course of PSP in response to a presynaptic spike at time $s = 0$
$g(u_1(t))$	gain function 2.4 evaluated at the value of the membrane potential $u_1(t)$ of neuron 1
$g'(u)$	derivative of $g(u)$ with respect to u

The term B_1^k is responsible for regulating the mutual information between input and output and maintaining the constant target firing rate for neuron 1:

$$B_1^k(\gamma) = \frac{y_1^k}{\Delta t} \log \left[\frac{g(u_1(t^k))}{\bar{g}_1(t^k)} \left(\frac{\tilde{g}}{\bar{g}_1(t^k)} \right)^{\gamma} \right] - (1 - y_1^k) R_1(t^k) \left[g(u_1(t^k)) - (1+\gamma)\bar{g}_1(t^k) + \gamma \tilde{g} \right]. \tag{3.9}$$

$R_1(t^k)$	refractory variable 2.3 of neuron 1 at time t^k
$\bar{g}_1(t^k)$	running average of the postsynaptic firing rate $g(u_1(t^k))$ of neuron 1
$\tilde{g}$	constant target firing rate

(Continued on next page)

Table 1: Continued

The term B_{12}^k measures the mutual information between the output spike train Y_1^k of neuron 1 and and the target spike train Y_2^k:

$$\begin{aligned} B_{12}^k = & \frac{y_1^k y_2^k}{(\Delta t)^2} \log \frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)\bar{g}_2(t^k)} - \frac{y_1^k}{\Delta t}(1-y_2^k) R_2(t^k) \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k) \right] \\ & - \frac{y_2^k}{\Delta t}(1-y_1^k) R_1(t^k) \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k) \right] \\ & + (1-y_1^k)(1-y_2^k) R_1(t^k) R_2(t^k) \left[\bar{g}_{12}(t^k) - \bar{g}_1(t^k)\bar{g}_2(t^k) \right]. \end{aligned} \tag{3.10}$$

y_2^k	binary variable indicating a spike in the target spike train in the kth time step
$R_2(t^k)$	refractory state of target spike train
$\bar{g}_2(t^k)$	running average of firing rate of target spike train
$\bar{g}_{12}(t^k)$	running average of the product between firing rates of the output and target spike train

Table 2: Summary of the Simplified (Rate-Based) Learning Rule for the Information Bottleneck Task Derived in Section 3.2.

For a simplified neuron model without refractoriness the spike-based rule, equation 3.8, reduces to the following rate-based rule:

$$\begin{aligned} \frac{\Delta w_{1j}^k}{\Delta t} = & -\alpha \nu_j^{pre,k} f(\nu_1^k) \left\{ \log \left[\frac{\nu_1^k}{\bar{\nu}_1^k} \left(\frac{\bar{\nu}_1^k}{\bar{g}} \right)^\gamma \right] \right. \\ & \left. - \beta \Delta t \left(\nu_2^k \log \left[\frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k \bar{\nu}_2^k} \right] - \bar{\nu}_2^k \left[\frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k \bar{\nu}_2^k} - 1 \right] \right) \right\}. \end{aligned} \tag{3.11}$$

α	learning rate
β, γ	optimization parameters
$\nu_j^{pre,k}$	presynaptic firing rate at synapse j at time t^k
$f(\nu_1^k)$	sensitivity of neuron 1 at its current firing state ν_1^k
ν_1^k	output firing rate of neuron 1 at time t^k
ν_2^k	firing rate of the target signal at time t^k
$\bar{\nu}_1^k$, $\bar{\nu}_2^k$	running averages of ν_1^k and ν_2^k
$\bar{\nu}_{12}^k$	running average of the product $\nu_1^k \nu_2^k$

proportional to the impact of synapse j onto the membrane potential at time t^k times the sensitivity of the output rate on changes of the membrane potential at time t^k. This prefactor distributes the weight changes given by the terms in the curly brackets to the individual synapse j. Changes of strongly active synapses are larger than those of relatively silent ones. We can divide the term in the curly brackets into three functionally different parts. Each of these parts corresponds to the optimization of one of the terms in equation 3.1. The first part, $\log(\nu_1^k/\bar{\nu}_1^k)$, together with the prefactor $\nu_j^{pre,k} f(\nu_1^k)$, drives the optimization of mutual information between inputs and outputs (note that this part is combined with the second part in equation 3.11, which is discussed below). The second part, $\log(\bar{\nu}_1^k/\tilde{g})^\gamma$, accounts for homeostatic processes to stabilize the output rate. These two parts together with the prefactor introduce competition between the synapses, and, as already noted in Toyoizumi et al. (2005), they implement a BCM-like learning rule. The third part is given by the two terms of the second line of equation 3.11. These terms drive the maximization of mutual information between the output of the neuron Y_1^K and the target signal Y_2^K. We investigate this part of the update rule in more detail. The correlation between ν_1^k and ν_2^k is measured by

$$\phi := \frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k \bar{\nu}_2^k}, \tag{4.1}$$

which appears in both terms of the second line of equation 3.11. It has value 1 for uncorrelated firing rates: values over 1 for positive correlations and values less than 1 for negative correlations (anticorrelations). To see how the second line of equation 3.11 depends on the ratio between ν_2^k and $\bar{\nu}_2^k$, we assume that $\bar{\nu}_2^k$ is constant and introduce $\zeta := \nu_2^k/\bar{\nu}_2^k$. Then the second line of the equation is proportional to

$$\zeta \log(\phi) - (\phi - 1). \tag{4.2}$$

For $\nu_2^k = \bar{\nu}_2^k$, this function is negative if $\phi \neq 1$ and zero if $\phi = 1$ (dashed line in Figure 4B).

Suppose that the output of the neuron is positively correlated with the target signal ($\phi > 1$; see Figure 4B). Then a firing rate ν_2^k of this target signal sufficiently above mean (e.g., $\zeta = 2$) induces long-term potentiation (LTP) in active synapses (i.e., synapses j with large $\nu_j^{pre,k}$). This will further increase the correlation between ν_1^k and ν_2^k for the encountered input. A firing rate ν_2^k of the target signal below mean ($\zeta < 1$) will induce long-term depression (LTD) in active synapses. Again, this increases the correlation between ν_1^k and ν_2^k.

For anticorrelated signals ($\phi < 1$; see Figure 4B), firing rates ν_2^k sufficiently below mean (e.g., $\zeta = 0$) induce LTP in active synapses. This will increase the anticorrelation between ν_1^k and ν_2^k for the encountered input.

Similarly, anticorrelation is increased for ν_2^k above mean, when LTD is induced in active synapses. Note that correlation and anticorrelation both contribute to the increase of mutual information.

4.3 Comparison with the BCM Learning Rule. To elucidate the relation to the classical Bienenstock-Cooper-Munro (BCM) learning rule (Bienenstock et al., 1982) we rewrite the simplified rule, equation 3.11, as

$$\frac{\Delta w_{1j}^k}{\Delta t} = -\alpha \nu_j^{pre,k} \Phi\left(\nu_1^k, \nu_2^k\right), \tag{4.3}$$

where Φ is a two-dimensional function of the firing rates ν_1^k and ν_2^k,

$$\Phi\left(\nu_1^k, \nu_2^k\right) = f\left(\nu_1^k\right) \left\{ \log\left[\frac{\nu_1^k}{\bar{\nu}_1^k} \left(\frac{\bar{\nu}_1^k}{\tilde{g}} \right)^{\gamma} \right] - \beta \Delta t \left[\nu_2^k \log \phi - \bar{\nu}_2^k (\phi - 1) \right] \right\}, \tag{4.4}$$

with $\phi = \bar{\nu}_{12}^k / (\bar{\nu}_1^k \bar{\nu}_2^k)$. This function $\Phi(\nu_1^k, \nu_2^k)$ can be seen as an extension of the classical BCM synaptic modification function (Bienenstock et al., 1982; Toyoizumi et al., 2005) and is plotted in Figures 5A to 5D for the special case that both average firing rates are equal to the constant target firing rate (i.e., $\bar{\nu}_1^k = \bar{\nu}_2^k = \tilde{g} = 20$ Hz) for four different values of the quotient ϕ.

Because of the anti-Hebbian nature of equation 4.3, values of Φ above 0 produce LTD. An analogous Hebbian learning rule for the extraction of independent components is derived in section 6. For such Hebbian learning rules, values of Φ above 0 produce LTP. One sees that for $\phi = 1$ (see Figure 5B), the second term in equation 4.4 vanishes, in which case, Φ does not depend on ν_2^k and reduces to the classical BCM function in Toyoizumi et al. (2005), where regimes of LTP and LTD are separated by a sliding threshold that depends in a nonlinear way on the running average of the postsynaptic rate $\bar{\nu}_1^k$. On the other hand, if $\phi \neq 1$, the value of Φ additionally depends on the current firing rate ν_2^k, which results in shifted versions of the BCM function where the balance between positive and negative domains varies as ν_2^k is changed from small to large values.

If $\phi < 1$, the signals are anticorrelated (see Figure 5A). In this case, Φ is more negative for small values of ν_2^k and more positive for large values of ν_2^k. This means that for the anti-Hebbian learning rule, equation 4.3, weights (and therefore also the firing rate ν_1^k) tend to increase for small ν_2^k and decrease for large ν_2^k. Therefore, the output of the neuron and the target signal become even more anticorrelated. Similarly, for correlated signals ($\phi > 1$; see Figures 5C and 5D), their correlation increases even further, since for small values of ν_2^k, the output firing rate ν_1^k tends to decrease as well (due to positive values of Φ), whereas it grows for large ν_2^k (because of

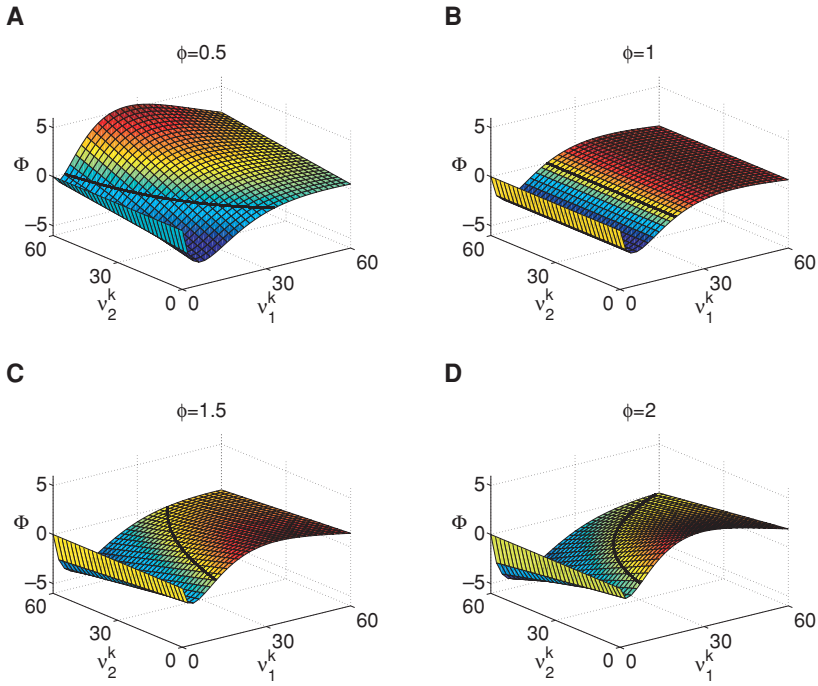

Figure 5.

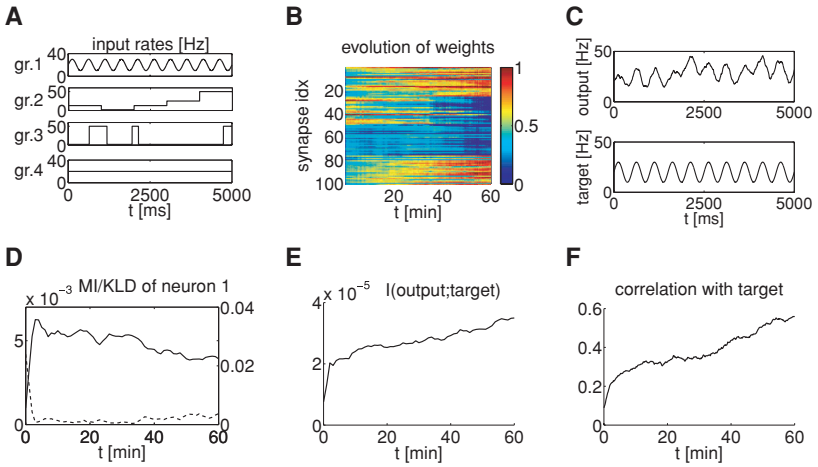

Figure 6.

negative values of Φ). In both cases (correlated or anticorrelated signals), the statistical dependence between the output and the target signal increases, as should be the case for an IB task.

4.4 Comparison with a Previously Proposed Method for Information Bottleneck Optimization. In the original formulation of the information bottleneck method (Tishby et al., 1999), the data variable X should be compressed as much as possible by a quantization or representation $\tilde{X}$. At the same time, however, this compressed variable should capture as much information as possible about a relevance variable Y. There is a trade-off between compression and preserving meaningful information, leading to the following objective function to minimize,

$$L = I(\tilde{\mathbf{X}}; \mathbf{X}) - \beta I(\tilde{\mathbf{X}}; \mathbf{Y}), \tag{4.5}$$

Figure 5: Two-dimensional synaptic modification function $\Phi(\nu_1^k, \nu_2^k)$, equation 4.4, of the rate-based learning rule, equation 3.11, as an extension of the classical BCM rule for $\bar{\nu}_1^k = \bar{\nu}_2^k = \tilde{g} = 20$ Hz, $\beta = 50$, $\gamma = 1$, and different values of the quotient $\phi = \bar{\nu}_{12}^k/(\bar{\nu}_1^k \bar{\nu}_2^k)$, which measures the correlation between the output of the neuron and the target signal. The sliding threshold between LTP and LTD depends not only on the postsynaptic firing rate r_1^k, but also on the target signal r_2^k if both signals are correlated ($\phi > 1$) or anticorrelated ($\phi < 1$). (A $\phi = 0.5$. B $\phi = 1$. C $\phi = 1.5$. D $\phi = 2$). Note that Φ is reduced to a one-dimensional function (like in the classical BCM-rule) for $\phi = 1$ (see panel B). In each plot, the solid black line indicates the transition from depression to potentiation ($\Phi = 0$).

Figure 6: Extracting a single rate modulation with the spike-based rule, equation 3.8. (A) Modulation of input rates for each of the four groups. (B) Evolution of weights during 60 minutes of learning (red: strong synapses, $w_{ij} \approx 1$; blue: depressed synapses, $w_{ij} \approx 0$). Weights were initialized randomly between 0.10 and 0.12, $\alpha = 5 \cdot 10^{-4}$, $\beta = 10^3$, $\gamma = 10$. Each group receives Poisson input with a different rate modulation $r_i(t)$; the rate modulation of the target signal is the same as for input group 1, $r_T(t) = r_1(t)$. (C) Output rate and rate of the target signal during 5 s after learning. (D) Evolution of the average mutual information per time bin (solid line, left scale) between input and output, and the Kullback-Leibler divergence per time bin (dashed line, right scale) as a function of time. Averages are calculated over segments of 1 minute. (E) Evolution of the average mutual information per time bin between output and the target signal as a function of time. (F) Trace of the time-varying correlation between output rate and rate of the target signal during learning. Correlation coefficients are calculated every 10 s.

where $\beta > 0$ is a trade-off parameter. If the joint distribution $P(X, Y)$ is given, the value of L depends only on the stochastic mapping[4] $P(\tilde{X} \mid X)$, because $\tilde{X}$ is independent of Y given X. For a given β, the optimal solution that minimizes equation 4.5 is given by Tishby et al. (1999),

$$P(\tilde{X} \mid X) = \frac{P(\tilde{X})}{Z(X, \beta)} \exp\left[-\beta D_{KL}(P(Y \mid X) \| P(Y \mid \tilde{X}))\right], \tag{4.6}$$

where $Z(X, \beta)$ is a normalization function. Note that equation 4.6 is implicit because both $P(\tilde{X})$ and $P(Y \mid \tilde{X})$ depend on $P(\tilde{X} \mid X)$ through

$$P(\tilde{X}) = \sum_X P(X) P(\tilde{X} \mid X) \tag{4.7}$$

and

$$P(Y \mid \tilde{X}) = \frac{1}{P(\tilde{X})} \sum_X P(X, Y) P(\tilde{X} \mid X). \tag{4.8}$$

Equations 4.6 to 4.8 can be solved iteratively with an extension of the Blahut-Arimoto (BA) algorithm, which is well known from applications to problems from rate distortion theory and channel capacity calculations (Tishby et al., 1999; Cover & Thomas, 1991). This generalized BA algorithm performs alternating iterations over the distributions $P(\tilde{X} \mid X)$, $P(\tilde{X})$, and $P(Y \mid \tilde{X})$ and can be shown to converge to the optimal solution of equations 4.6 to 4.8 (Tishby et al., 1999). In the following we briefly discuss the relationship between this traditional IB algorithm and our learning rule for spiking neurons.

The traditional IB approach has so far mainly been applied to discrete variables X, $\tilde{X}$, and Y in a wide range of applications (see Slonim, 2002, for a review and references). However, in the general theory, there is no restriction on the type of these variables. In this article, we apply the IB principle to spike trains (see Figure 1A): the input spike trains X^K to the learning neuron correspond to the data variable X, the output spike train Y_1^K of this neuron represents the compressed variable $\tilde{X}$, and the target spike train Y_2^K specifies the relevant variable Y. This yields the following correspondence to the notation of Tishby et al. (1999):

$$\begin{aligned} P(\tilde{X} \mid X) &\triangleq P\left(Y_1^K \mid X^K\right), \\ P(\tilde{X}) &\triangleq P\left(Y_1^K\right), \\ P(Y \mid \tilde{X}) &\triangleq P\left(Y_2^K \mid Y_1^K\right). \end{aligned}$$

[4] This means that the objective function L, equation 4.5, can be written as a functional $\mathcal{L}[P(\tilde{X}|X)] = I(\tilde{\mathbf{X}}; \mathbf{X}) - \beta I(\tilde{\mathbf{X}}; \mathbf{Y})$ and minimizing the equation, is equivalent to minimizing the functional $\mathcal{L}[P(\tilde{X}|X)]$ with respect to the conditional distribution $P(\tilde{X}|X)$ (Tishby et al., 1999).

In the traditional IB algorithms, one usually specifies the trade-off parameter β and the joint distribution $P(X, Y)$ in advance. This is also the case for our experiments (see section 5) where we choose a particular statistics for the input and target spike trains, X^K and Y_2^K. Furthermore, both the BA algorithm and our learning rule search for the optimal distributions $P(\tilde{X} \mid X)$ and $P(Y_1^K \mid X^K)$, respectively. However, the compression achieved by the mapping $P(Y_1^K \mid X^K)$ is modeled not explicitly but implicitly through the weights w_{1j} of the learning neuron. By updating these weights, we successively adapt the stochastic input-output relationship given by $P(Y_1^K \mid X^K)$. Due to this modification of $P(Y_1^K \mid X^K)$, the distributions $P(Y_1^K) = \langle P(Y_1^K \mid X^K) \rangle_{X^K}$ and $P(Y_2^K \mid Y_1^K) = P(Y_1^K, Y_2^K)/P(Y_1^K)$ change implicitly, whereas in the traditional IB algorithm, the corresponding distributions $P(\tilde{X})$ and $P(Y \mid \tilde{X})$ are updated in a separate step. However, as in the BA algorithm, where the new value of $P(\tilde{X} \mid X)$ depends on the values of $P(\tilde{X})$ and $P(Y \mid \tilde{X})$, in our learning rule the adaptation of $P(Y_1^K \mid X^K)$, that is, the change of the weights w_{1j}, depends on $P(Y_1^K)$ and $P(Y_2^K \mid Y_1^K)$ through the terms C_{1j}^k, B_1^k, and B_{12}^k of the learning rule, equation 3.8.

More precisely, by comparing the current firing rate with its running average, the term B_1^k, equation 3.9, depends on both the output distribution $P(Y_1^k)$ and the probability of the output given the input spike trains, $P(Y_1^k \mid X^k)$ (see also Toyoizumi et al., 2005). The distribution $P(Y_2^k \mid Y_1^k)$ influences the term B_{12}^k, equation 3.10, since this term compares the joint probability $P(Y_1^k, Y_2^k)$ with the independent distribution $P(Y_1^k)P(Y_2^k)$, or equivalently, $P(Y_2^k \mid Y_1^k)$ with $P(Y_2^k)$. Finally, both terms B_1^k and B_{12}^k are multiplied in the learning rule, equation 3.8, with the correlation term C_{1j}^k, which can be written as the derivative of the logarithm of $P(Y_1^K \mid X^K)$ with respect to the weights w_{1j}, $C_{1j}^k = \frac{\partial}{\partial w_{1j}} \log P(Y_1^k \mid X^k)$ (see equation 3.6 and section A.2).

Summarizing, the main difference between the previous IB approach from Tishby et al. (1999) and our special application to spiking neurons is that in our case, the distribution under consideration, $P(Y_1^K \mid X^K)$, is parameterized by the weights w_{1j} of a spiking neuron (whereas in most IB algorithms, no special assumptions are made about the probability distributions to be optimized), and our learning rule is an online learning rule performing gradient ascent on the objective function. In the generalized BA algorithm, the probability distributions are changed directly (e.g., by maintaining probability tables) and always converge to the optimal solution, whereas our learning rule can change them only implicitly by adapting the weights w_{1j}, and there is no guarantee that the global optimum is found. Another difference is that the IB algorithm from Tishby et al. (1999) is an offline algorithm that performs the optimization over the whole range of the random variables, whereas our algorithm is an online algorithm where the weights are adapted sequentially as the input and the target spike trains are presented to the neuron. In this sense, our update rule can be viewed as a novel online learning approach to IB optimization for a concrete parameterized instance of the problem.

5 Application to Information Bottleneck Optimization

We use a setup as in Figure 1A where we want to maximize the information that the output Y_1^K of a learning neuron conveys about one or more target signals Y_2^K, Y_3^K, ... In the following simulations, we let the neuron receive inputs X^K at $N = 100$ synapses, with weights randomly initialized at small values (from 0.10 to 0.12). Unless stated otherwise, we choose $\tilde{g} = 30$ Hz for the target firing rate, and we use discrete time with $\Delta t = 1$ ms.

5.1 Extracting a Single Rate Modulation. In a first experiment, we investigate how the spike-based learning rule, equation 3.8, performs in a simple rate coding paradigm: the information is encoded in the firing rates of the spike trains. We divide the inputs into four groups of 25 synapses each. In the following, let $r_i(t)$ and $r_T(t)$ denote the firing rate of group i ($i = 1, \ldots, 4$) and of the target signal, respectively, at time t. Each input spike train is generated by an inhomogeneous Poisson process with common rate modulation within each group; however, the rate modulations for different groups are statistically independent (see Figure 6A). More precisely, for input group 1 (synapses 1 to 25), we choose a periodic rate modulation $r_1(t) = r_0 + A\sin(2\pi t/T)$ with $r_0 = 20$ Hz, $A = 10$ Hz, and $T = 500$ ms. The rate of group 2 (synapses 26 to 50) is constant during intervals of 1 s, each second a firing rate is chosen randomly out of the values 2 Hz, 13 Hz, 25 Hz, 40 Hz, and 50 Hz. Synapses 51 to 75 (input group 3) receive a rate that has a constant value of 2 Hz, except that a burst is initiated at each time step with a probability of 0.0005. Thus, there is a burst on average every 2 s. The duration of a burst is chosen from a gaussian distribution with mean 0.5 s and SD 0.2 s; the minimum duration is chosen to be 0.1 s. During a burst, the rate is set to 50 Hz. Finally the remaining synapses (76 to 100; group 4) receive constant rate Poisson spike trains at 20 Hz.

We generate the target spike train by an inhomogeneous Poisson process with the same rate modulation as the inputs of group 1, $r_1(t)$. In this case, we expect that weights will grow only for the first group and remain depressed for the other inputs, since these are the only inputs that are not statistically independent from the target signal. However, Figure 6 shows that besides for group 1, strong weights are also developed for group 4, the uncorrelated constant rate Poisson input. This is because the neuron has to achieve a mean postsynaptic firing rate close to the constant target firing rate of 30 Hz, and uncorrelated Poisson spike trains with a constant rate are always statistically independent from any other spike train. Therefore, developing strong weights for this group of inputs does not increase the mutual information between input and output, which should be kept as low as possible. All other synapses are depressed because their inputs are statistically independent from the target signal. Moreover, Figure 6 shows that after learning, the time course of the output rate modulation is similar to that of the target signal; therefore, the neuron has learned to "represent"

the target signal. Furthermore, the mutual information between input and output decreases, whereas the information as well as the correlation between the output and the target signal increases.

Further experiments show that one can also extract the rates of input groups 2 and 3, $r_2(t)$ and $r_3(t)$, if a correlated spike train is chosen as the target signal. However, it is not reasonable to take an uncorrelated fixed-rate Poisson spike train as the target spike train, since it does not contain mutual information with any of the inputs. Using such a target has the same effect as removing the target signal (see the next experiment).

5.2 Extracting a Time-Varying Combination of Rate Modulations. In the second experiment, we consider a target signal that is only indirectly related to some of the inputs, and in addition this relationship varies over time. Again, the input is divided into four groups of 25 synapses, each with different rate modulations. This time we use rates that are constant during random intervals and can take five different values: 2 Hz, 13 Hz, 25 Hz, 40 Hz, and 50 Hz. The time during which the rate remains constant is drawn uniformly from the interval [0 s, 1 s], and the value of the rate is also chosen uniformly among the five available values. The spike trains are generated from an inhomogeneous Poisson process with a rate modulation created with this method for each of the four input groups, independently from each other (see Figure 7A).

The rate of the target signal $r_T(t)$ is chosen to be a linear combination of the input rates. At the beginning, we set it to the mean between the rates of group 1 and 2, that is, $r_T(t) = (r_1(t) + r_2(t))/2$, in order to test whether this suffices for triggering the increase of weights from input groups 1 and 2. To make the experiment more interesting, we change the rate of the target signal to the mean of rates of group 1 and 3, $r_T(t) = (r_1(t) + r_3(t))/2$, after 15 minutes . Furthermore, to investigate the effect of removing the target signal after some time, we switch it off after 45 minutes ($r_T(t) = 0$).

Figure 7 shows the performance of the simplified learning rule, equation 3.11, for this task. In Figure 7B, we see that weights grow initially for input groups 1 and 2 and remain depressed for the other inputs, as expected. After 15 minutes, as the firing rate combination of the target signal changes, the weights of group 2 are weakened, whereas the efficacies of the third group now start to grow. This means that the learning rule is able to adapt to new situations where the relevant target signal changes. However, the final distribution of synaptic efficacies persists when the target signal is removed after 45 minutes.

5.3 Extracting Spike-Spike Correlations. So far we have considered only rate coding—the information was encoded in the firing rates of the spike trains. But can the proposed learning rule also take into account information that is contained in the spike timings rather than in the firing rates? In the next experiment, we investigate the effect of spike-spike correlations

between the target spike train and parts of the input for the spike-based learning rule, equation 3.8. All input spike trains and the target spike train are now generated by a Poisson process at a constant rate of 20 Hz. However, different correlation groups are established within the inputs in the following way. The first 25 inputs are strongly correlated with the target spike train (with a coefficient of 0.5), and the second 25 synapses have weaker correlations with the target spike train (coefficient 0.2). The remaining 50 inputs

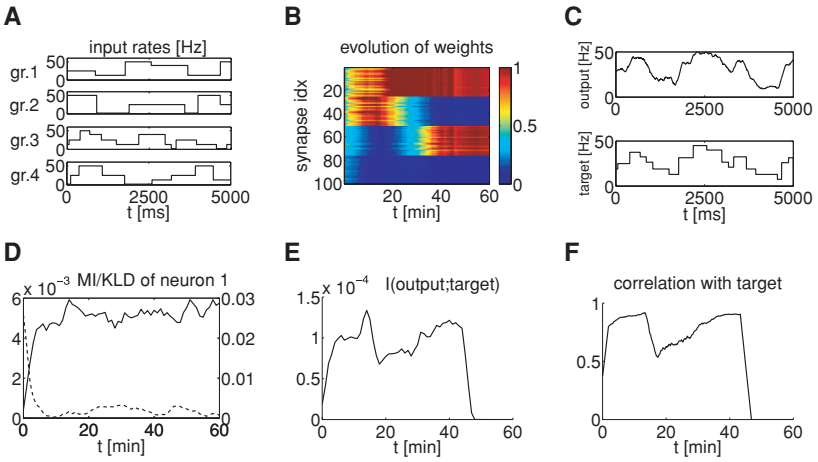

Figure 7.

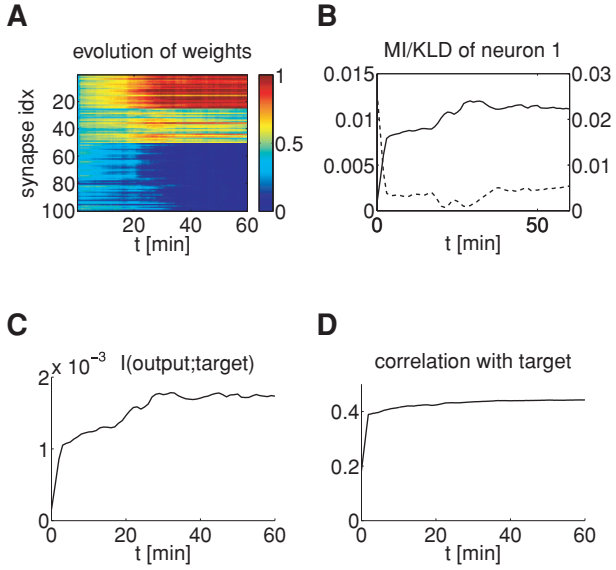

Figure 8.

are uncorrelated with the target spike train; however, inputs 51 to 75 are pairwise correlated with a coefficient of 0.5, and inputs 76 to 100 are uncorrelated. Inputs belonging to different groups are also uncorrelated. Correlated spike trains are generated by the procedure described in Gütig, Aharonov, Rotter, and Sompolinsky (2003) and Legenstein, Näger, and Maass (2005).

Figure 8 shows that strong weights grow for those synapses where the input has spike-spike correlations with the target spike train. Because the first group of inputs is correlated more strongly than the second group,

Figure 7: Extracting input components that are indirectly and just during certain time points related to the target signal with the rate-based rule, 3.11. (A) Modulation of input rates for each of the four groups. Each group i receives Poisson input with a different rate modulation $r_i(t)$. (B) Evolution of weights during 60 minutes of learning (red: strong synapses, $w_{ij} \approx 1$; blue: depressed synapses, $w_{ij} \approx 0$). Weights were initialized randomly between 0.10 and 0.12, $\alpha = 10^{-3}$, $\beta = 5 \cdot 10^3$, $\gamma = 10$. Initially, $r_T(t) = 1/2(r_1(t) + r_2(t))$; after 15 minutes it changes to $r_T(t) = 1/2(r_1(t) + r_3(t))$. After 45 minutes, $r_T(t) = 0$. (C) Output rate and rate of the target signal during 5 s just before the target signal is removed. (D) Evolution of the average mutual information between input and output per time bin (solid line, left scale) and the Kullback-Leibler divergence per time bin (dashed line, right scale) as a function of time. Averages are calculated over segments of 1 minute. (E) Evolution of the average mutual information per time bin between output and the target signal as a function of time. (F) Trace of the correlation between output rate and rate of the target signal during learning. Note that the target signal is changed after 15 minutes and set to 0 after 45 minutes. Correlation coefficients are calculated every 10 seconds.

Figure 8: Extracting spike-spike correlations with the spike-based learning rule, equation 3.8. (A) Evolution of weights during 60 minutes of learning (red: strong synapses, $w_{ij} \approx 1$; blue: depressed synapses, $w_{ij} \approx 0$.) Weights were initialized randomly between 0.10 and 0.12, $\alpha = 10^{-4}$, $\beta = 10^2$, $\gamma = 50$. All inputs and the target spike train are Poisson spike trains at a constant rate of 20 Hz. Input group 1 and the target spike train, are correlated with a coefficient of 0.5; between input group 2 and the target spike train, a correlation coefficient of 0.2 is established. Group 3 is also correlated with 0.5, but uncorrelated to the target spike train, and group 4 is uncorrelated at all. Spike trains from different groups are uncorrelated. (B) Evolution of the average mutual information per time bin (solid line, left scale) between input and output, and the Kullback-Leibler divergence per time bin (dashed line, right scale) as a function of time. Averages are calculated over segments of 1 minute. (C) Evolution of the average mutual information per time bin between output and the target signal as a function of time. (D) Trace of the current spike-spike correlation between the output spike train and the target spike train during learning. Correlation coefficients are calculated every 10 s. This experiment shows that the neuron learns with the IB learning rule to extract information from high-dimensional input streams that is contained in the spike times.

weights from the first group reach their maximum value of 1, whereas those for the second group grow up only to a value of about 0.5. That is, the learning rule is sensitive to different levels of correlation. Note that the information conveyed by spike-spike correlations is about one order of magnitude larger than in the previous experiments with rate coding. In Figure 8D, the correlation between the output and the target spike train is bounded from above by the maximum correlation of inputs with the target spike train (0.5).

5.4 Extracting Information That Is Relevant for two Different Target Signals. We use a setup as in Figure 1A where we want to maximize the information that the output Y_1^K of a learning neuron conveys about two target signals, Y_2^K and Y_3^K. If the target signals are statistically independent from each other, we can optimize the mutual information to each target signal separately; we include the term $\beta(I(\mathbf{Y}_1^K; \mathbf{Y}_2^K) + I(\mathbf{Y}_1^K; \mathbf{Y}_3^K))$ in the objective function 3.1. This leads to an update rule,

$$\frac{\Delta w_{1j}^k}{\Delta t} = -\alpha C_{1j}^k \left[B_1^k(-\gamma) - \beta \Delta t \left(B_{12}^k + B_{13}^k \right) \right], \tag{5.1}$$

where B_{12}^k and B_{13}^k are the postsynaptic terms, equation 3.10, sensitive to the statistical dependence between the output and target signals 1 and 2, respectively.

In this experiment, we demonstrate that it is possible to consider two very different kinds of target signals: one target spike train has a similar rate modulation as one part of the input, while the other target spike train has a high spike-spike correlation with another part of the input. The first two of the four input groups consist of rate-modulated Poisson spike trains, where the rate of the first 25 inputs is modulated by a gaussian white noise signal with mean 20 Hz that has been low-pass-filtered with a cut-off frequency of 5 Hz. Synapses 26 to 50 receive the burst signal described in section 5.1, which was used there for input group 3 (see Figure 9A). Spike trains from the remaining groups 3 and 4 are Poisson spike trains at a constant rate of 20 Hz, but have spike-spike correlations with a coefficient of 0.5 within each group. However, spike trains from different groups are uncorrelated. The first target spike train is chosen to have a similar rate modulation as the inputs from group 1; gaussian random noise is superimposed on the rate with a standard deviation of 2 Hz. The second target spike train is correlated with inputs from group 3 (with a coefficient of 0.5) but uncorrelated to inputs from group 4. Furthermore, both target signals are silent during random intervals: at each time step, the rate of each target signal is independently set to 0 with a certain probability (10^{-5}) and remains silent for a duration chosen from a gaussian distribution with mean 5 s and SD 1 s (minimum duration is 1 s). Hence this experiment tests whether learning works even if the target signals are not available all of the time.

Figure 9 shows that strong weights evolve for the first and third groups of synapses, whereas the efficacies for the remaining inputs are depressed. Both groups with growing weights are correlated with one of the target signals; therefore, the mutual information between output and target spike trains increases. Since spike-spike correlations convey more information than rate modulations, synaptic efficacies develop more strongly to group 3 (the group with spike-spike correlations). This results in an initial decrease in correlation with the rate-modulated target signal to the benefit of higher correlation with the second target spike train. However, after about 30 minutes when the weights become stable, the correlations as well as the mutual information quantities stay roughly constant.

5.5 Extracting Information Uncorrelated with the Target Signal But with Higher-Order Statistical Dependencies. So far we have analyzed the IB setup only for situations where the target signal is correlated to parts of the input (via either rate correlations or spike-spike correlations). To show that the learning rule is also able to extract statistical dependencies of higher order, we try to extract uncorrelated but still statistically dependent information. In this experiment, we use again rate coding and choose the firing rate of the target signal to be a function of one of the input rate modulations as to induce strong statistical dependence between the target spike train and this input group. In order to decorrelate the target signal from this input, a whitening transformation is applied (see appendix B).

We generate the rate modulations for the four input groups $r_1(t), \ldots, r_4(t)$ in the same way as in the experiment described in section 5.3: piecewise constant rates chosen randomly out of the set {2 Hz, 13 Hz, 25 Hz, 40 Hz, 50 Hz}, and the duration during which the rate is constant is drawn uniformly from the interval [0 s, 1 s]. Inputs from the same group share the same rate modulation, and inputs from different groups are statistically independent, since the rates are drawn independently for each group. The rate of the target signal is chosen to be a function of the first input rate: $r_T(t) = f(r_1(t))$, where $f(2) = 13$ Hz, $f(13) = 25$ Hz, $f(25) = 40$ Hz, $f(40) = 50$ Hz, and $f(50) = 2$ Hz. In this way, statistical dependence has been established between the first input group and the target spike train. Now, the whitening transformation is applied to decorrelate the rate modulation of the first input group, $r_1(t)$, and the target signal, $r_T(t)$, yielding $\tilde{r}_1(t)$ and $\tilde{r}_T(t)$. Finally, the input spike trains are generated by inhomogeneous Poisson processes with the rates $\tilde{r}_1(t)$, $r_2(t)$, $r_3(t)$, and $r_4(t)$, and the target spike train is drawn from $\tilde{r}_T(t)$. For this experiment, we choose $\tilde{g} = 20$ Hz.

The performance of the rate-based rule on this task is shown in Figure 10. It can be seen that weights reach values close to maximal efficacy for the statistically dependent group (group 1) and finally get depressed for the remaining inputs. The output is now uncorrelated to, but still statistically dependent on, the target signal. Note that the mutual information between output and target signal increases, whereas the correlation stays around 0.

This means that the learning rule is also sensitive to higher-order statistical dependencies.

6 Extracting Independent Components

With a slight modification in the objective function 3.1, the learning rule allows us to extract statistically independent components from an ensemble

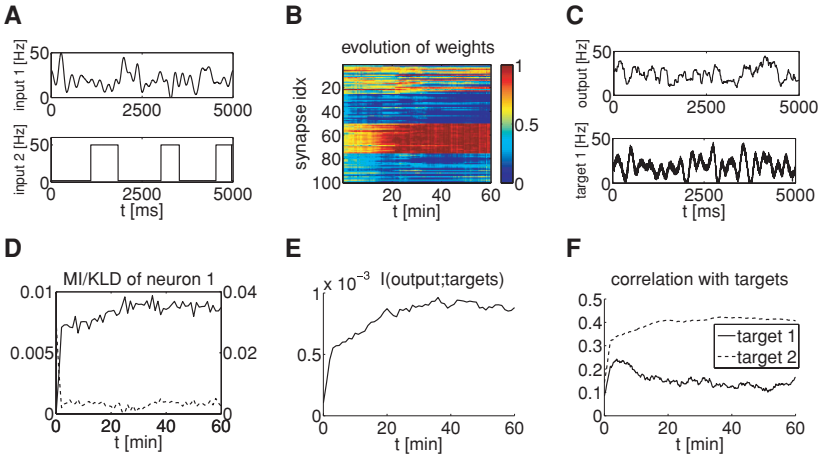

Figure 9.

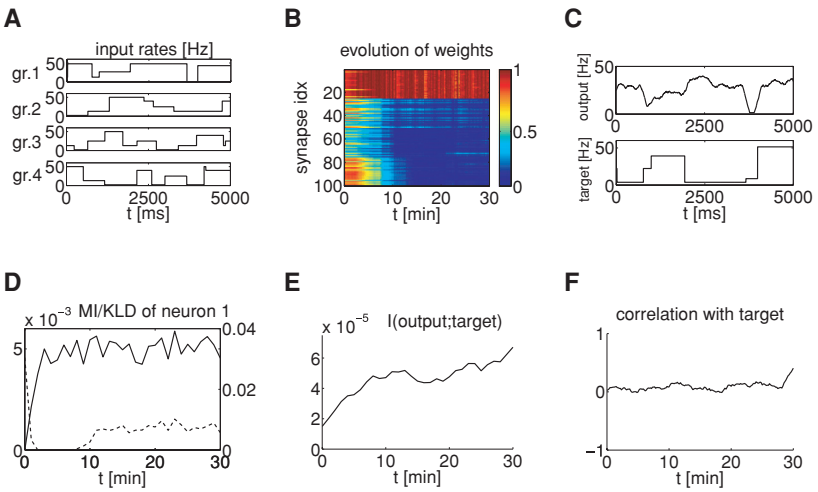

Figure 10.

of input spike trains. We consider two neurons receiving the same input at their synapses (see Figure 1B). For both neurons $i = 1, 2$, we maximize information transmission under the constraint that their outputs

Figure 9: Using two target signals at the same time with the spike-based rule, equation 3.8. (A) Modulation of input rates to input groups 1 and 2. (B) Evolution of weights during 60 minutes of learning (red: strong synapses, $w_{ij} \approx 1$; blue: depressed synapses, $w_{ij} \approx 0$.) Weights were initialized randomly between 0.10 and 0.12, $\alpha = 10^{-4}$, $\beta = 2 \cdot 10^3$, $\gamma = 50$. Input groups 1 and 2 receive Poisson spike trains with different rate modulations; groups 3 and 4 receive constant rate Poisson at 20 Hz, but each group is correlated with a coefficient of 0.5, and spike trains from different groups are uncorrelated. The first target spike train is a Poisson spike train with the same rate modulation as group 1, superimposed with gaussian noise ($\sigma = 2$ Hz). The second target spike train has a constant rate of 20 Hz and is correlated with coefficient 0.5 to input group 3. (C) Output rate and rate of target signal 1 during 5 s after learning. (D) Evolution of the average mutual information per time bin (solid line, left scale) between input and output and the Kullback-Leibler divergence per time bin (dashed line, right scale) as a function of time. Averages are calculated over segments of 1 minute. (E) Evolution of the average mutual information per time bin between output and both target spike trains as a function of time. (F) Trace of the current correlation between output rate and rate of target signal 1 (solid line) and the spike-spike correlation (dashed line) between the output and target spike train 2 during learning. Correlation coefficients are calculated every 10 s.

Figure 10: Extracting uncorrelated but statistically dependent information with the rate-based rule, equation 3.11. (A) Modulation of input rates to input groups 1 to 4. (B) Evolution of weights during 30 minutes of learning (red: strong synapses, $w_{ij} \approx 1$; blue: depressed synapses, $w_{ij} \approx 0$.) Weights were initialized randomly between 0.10 and 0.12, $\alpha = 10^{-6}$, $\beta = 2 \cdot 10^7$, $\gamma = 50$. Each input group receives Poisson input with different rate modulations; the rate modulation of the target is a function of the rate of input group 1. The rate of the target signal and the rate of input group 1 are decorrelated using the whitening transform described in the text. Nevertheless, the learning rule picks out inputs that have statistical dependencies with the target signal. (C) Output rate and rate of the target signal during 5 s after learning. (D) Evolution of the average mutual information per time bin (solid line, left scale) between input and output and the Kullback-Leibler divergence per time bin (dashed line, right scale) as a function of time. Averages are calculated over segments of 1 minute. (E) Evolution of the average mutual information per time bin between output and target spike train as a function of time. One clearly sees that this mutual information keeps increasing, whereas the mutual information between input and output (see D) stays at the same level. (F) Trace of the correlation between output rate and rate of the target signal during learning. Correlation coefficients are calculated every 10 s.

stay as statistically independent from each other as possible. That is, we maximize

$$\tilde{L}_i = I\left(\mathbf{X}^K; \mathbf{Y}_i^K\right) - \beta I\left(\mathbf{Y}_1^K; \mathbf{Y}_2^K\right) - \gamma D_{KL}\left(P\left(Y_i^K\right) \| \tilde{P}\left(Y_i^K\right)\right). \tag{6.1}$$

Since the same terms (up to the sign) are optimized in equations 3.1 and 6.1, we can derive a gradient ascent rule for the weights of neuron i, w_{ij}, analogous to section 3:

$$\frac{\Delta w_{ij}^k}{\Delta t} = \alpha C_{ij}^k \left[B_i^k(\gamma) - \beta \Delta t B_{12}^k\right] \tag{6.2}$$

(see Table 1 for a definition of the terms in this equation).

In order to compare this rule with the BCM model as in section 4.3, we consider the weight change of neuron 1 for the rate-based rule derived for the simplified neuron model,

$$\frac{\Delta w_{1j}^k}{\Delta t} = \alpha \nu_j^{pre,k} \tilde{\Phi}\left(\nu_1^k, \nu_2^k\right), \tag{6.3}$$

where $\tilde{\Phi}(\nu_1^k, \nu_2^k)$ is given by

$$\tilde{\Phi}\left(\nu_1^k, \nu_2^k\right) = f\left(\nu_1^k\right) \left\{ \log\left[\frac{\nu_1^k}{\bar{\nu}_1^k}\left(\frac{\bar{\nu}_1^k}{\tilde{g}}\right)^{-\gamma}\right] - \beta \Delta t \left[\nu_2^k \log \phi - \bar{\nu}_2^k(\phi - 1)\right] \right\}, \tag{6.4}$$

with $\phi = \bar{\nu}_{12}^k / (\bar{\nu}_1^k \bar{\nu}_2^k)$.

Compared to the IB rule, equation 4.3, the sign of the weight update has changed in equation 6.3, reflecting the different signs in the first two terms of the objective function 6.1 as compared to 3.1. The synaptic modification function, equation 6.4, is the same as $\Phi(\nu_1^k, \nu_2^k)$ in equation 4.4 except that γ in equation 4.4 is replaced by $-\gamma$. In the following discussion, we consider the case where the output rate of neuron 1 is already close to the target firing rate, so that $\bar{\nu}_1^k \approx \tilde{g}$. In this case, $\tilde{\Phi}(\nu_1^k, \nu_2^k)$ is approximately equal to $\Phi(\nu_1^k, \nu_2^k)$, and Figure 5 qualitatively also applies for $\tilde{\Phi}$.

Analogous arguments as in section 4.3 can be applied when comparing this rule with the BCM model. Because of the Hebbian nature of equation 6.3, values of Φ above 0 produce LTP and values below 0 produce LTD (see Figure 5). Again, for the special case $\phi = 1$ (see Figure 5B), the outputs are uncorrelated, and the learning rule reduces to the classical BCM rule: the output of neuron 2, ν_2^k, has no influence on the weight change Δw_{1j}^k of neuron 1. In case of anticorrelated outputs of the two neurons ($\phi < 1$; see Figure 5A), the learning rule will try to make them more correlated by increasing ν_1^k for large ν_2^k and decreasing ν_1^k for small ν_2^k. On the other hand, if the outputs are correlated ($\phi > 1$; see Figures 5C and 5D), anticorrelations

will be increased. For large values of ν_2^k, the output firing rate ν_1^k tends to decrease; for small values of ν_2^k, it increases. In this way, the learning rule tries to make these outputs statistically independent. Again, note that correlation and anticorrelation both contribute in the same way to mutual information.

6.1 An Approximation of the Learning Rule. The term B_{12}^k, equation 3.10, in the learning rule, equation 6.2, is nonlocal and difficult to implement by a spiking neuron in reality. In the following we provide an approximation to the learning rule, equation 6.2, in which we implement the effect of the term B_{12}^k by modifying the value $g(u_i(t^k))$ in the term B_i^k. This could provide an idea how this learning rule might be implemented in a biologically realistic circuit of neurons. More precisely, we let the weights of neuron i evolve according to the learning rule,

$$\frac{\Delta w_{ij}^k}{\Delta t} = \alpha C_{ij}^k \hat{B}_i^k(\gamma), \tag{6.5}$$

which is similar to the generalized BCM rule for spiking neurons presented in section 2, where

$$\begin{aligned}\hat{B}_1^k(\gamma) = {} & \frac{y_1^k}{\Delta t} \log\left[\frac{\hat{\bar{g}}_1(t^k)}{\bar{g}_1(t^k)}\left(\frac{\tilde{\bar{g}}}{\bar{g}_1(t^k)}\right)^{\gamma}\right] \\ & -\left(1-y_1^k\right) R_1(t^k)\left[\hat{\bar{g}}_1(t^k) - (1+\gamma)\bar{g}_1(t^k) + \gamma\tilde{\bar{g}}\right],\end{aligned} \tag{6.6}$$

is $B_i^k(\gamma)$ with a modified gain function $\hat{g}_i(t^k)$ (see Table 3). Note that we do not change the actual gain function (i.e., firing behavior) of the neuron; the modified gain function $\hat{g}_i(t^k)$ is effective only in the learning rule.

To find the desired expression for $\hat{g}_i(t^k)$, we compare the combined postsynaptic term $B_i^k(\gamma) - \beta\Delta t B_{12}^k$ in equation 6.2 with the simple postsynaptic term $\hat{B}_i^k(\gamma)$, equation 6.6, for both neurons and for the two cases that the neuron itself or the other neuron has emitted a spike (see appendix C). This results in a modified gain function for the learning rule of neuron $i = 1, 2$ of

$$\hat{g}_i(t^k) = g(u_i(t^k)) \cdot a_i(t^k)^{y_i^k(1-y_{3-i}^k)} + b_i(t^k) y_{3-i}^k (1 - y_i^k). \tag{6.7}$$

The term

$$a_i(t^k) = \exp\left[R_{3-i}(t^k)\beta\Delta t\left(\frac{\bar{\bar{g}}_{12}(t^k)}{\bar{g}_i(t^k)} - \bar{g}_{3-i}(t^k)\right)\right] \tag{6.8}$$

corresponds to a multiplicative change of $g(u_i(t^k))$ in case of spikes of neuron i itself. If the outputs have been correlated (i.e., $\bar{\bar{g}}_{12}(t^k) > \bar{g}_1(t^k)\bar{g}_2(t^k)$) the modified gain in equation 6.5 is increased; if the outputs have been anticorrelated ($\bar{\bar{g}}_{12}(t^k) < \bar{g}_1(t^k)\bar{g}_2(t^k)$), it is decreased. If, on the other hand,

Table 3: Summary of the Approximation of the Learning Rule for Extracting Independent Components.

The weights w_{ij} of neuron $i = 1, 2$ evolve according to the generalized BCM rule for spiking neurons. The weight change Δw_{ij}^k at time $t^k = k\Delta t$ is given by

$$\frac{\Delta w_{ij}^k}{\Delta t} = \alpha C_{ij}^k B_i^k(\gamma) \tag{6.5}$$

with a learning rate $\alpha > 0$ and optimization parameter $\gamma > 0$.

The correlation term C_{ij}^k and the postsynaptic term $B_i^k(\gamma)$ are given by

$$C_{ij}^k = C_{ij}^{k-1}\left(1 - \frac{\Delta t}{\tau_C}\right) + \sum_{n=1}^{k} \epsilon(t^k - t^n) x_j^n \frac{g'(u_i(t^k))}{g(u_i(t^k))}\left[y_i^k - \rho_i^k\right] \tag{3.6}$$

$$B_i^k(\gamma) = \frac{y_i^k}{\Delta t}\log\left[\frac{\hat{g}_i(t^k)}{\bar{g}_i(t^k)}\left(\frac{\tilde{g}}{\bar{g}_i(t^k)}\right)^{\gamma}\right] - (1 - y_i^k) R_i(t^k)\left[\hat{g}_i(t^k) - (1+\gamma)\bar{g}_i(t^k) + \gamma\tilde{g}\right] \tag{6.6}$$

(compare to equation 3.9 in Table 1).

The original gain value $g(u_i(t^k))$ is modified both additively and multiplicatively:

$$\hat{g}_i(t^k) = g(u_i(t^k)) \cdot a_i(t^k)^{y_i^k(1-y_{3-i}^k)} + b_i(t^k) y_{3-i}^k (1 - y_i^k), \tag{6.7}$$

where $y_i^k \in \{0, 1\}$ indicates an output spike of neuron i at time t^k.

If neuron i itself has spiked, the value $g(u_i(t^k))$ is multiplied with the following factor:

$$a_i(t^k) = \exp\left[R_{3-i}(t^k)\beta\Delta t\left(\frac{\bar{g}_{12}(t^k)}{\bar{g}_i(t^k)} - \bar{g}_{3-i}(t^k)\right)\right]. \tag{6.8}$$

If the other neuron (neuron $3 - i$) has spiked, the following term is added to the value $g(u_i(t^k))$:

$$b_i(t^k) = -\beta\left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_{3-i}(t^k)} - \bar{g}_i(t^k)\right]. \tag{6.9}$$

a spike is elicited by the other neuron (neuron $3 - i$) the value $g(u_i(t^k))$ is modified additively by the term

$$b_i(t^k) = -\beta \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_{3-i}(t^k)} - \bar{g}_i(t^k) \right]. \tag{6.9}$$

In case of correlated outputs, it is decreased; in case of anticorrelated outputs, it is increased.

Note that equations 6.5 to 6.9 provide only an approximation to the learning rule, equation 6.2, because we have considered only the cases where one of the two neurons spikes. The approximation presented here is still not local because the modified value $\hat{g}_i(t^k)$ still depends on nonlocal variables, for example, the average product of firing rates $\bar{g}_{12}(t^k)$. However, it indicates what a real biological learning rule would have to approximate. Each neuron needs information about the firing behavior of both neurons. In particular, a circuit of interneurons would be necessary to implement some of the terms in equations 6.7 to 6.9.

6.2 Extracting Different Correlation Groups. Figure 12 shows the results of an experiment where two neurons receive the same Poisson input with a rate of 20 Hz at their 100 synapses. The input is divided into two groups of 40 spike trains each, such that synapses 1 to 40 and 41 to 80 receive correlated input with a correlation coefficient of 0.5 within each group; however, any spike trains belonging to different input groups are uncorrelated. The remaining 20 synapses receive uncorrelated Poisson input (see Figure 11 for a sample of such input spike trains). Weights close to the maximal efficacy $w_{\max} = 1$ are developed for one of the groups of synapses that receives correlated input (group 2 in this case), whereas those for the other correlated group (group 1), as well as those for the uncorrelated group (group 3), stay low. Neuron 2 develops strong weights to the other correlated group of synapses (group 1), whereas the efficacies of the second correlated group (group 2) remain depressed, thereby trying to produce a statistically independent output. For both neurons, the mutual information is maximized, and the target output distribution of a constant firing rate of 30 Hz is approached well. After an initial increase in both the mutual information and the correlation between the outputs, where the weights of both neurons start to grow simultaneously, these amounts drop as both neurons develop strong efficacies to different parts of the input.

6.3 Comparison with Other Neural ICA Learning Rules. Neural learning algorithms based on information optimization principles, such as independent component analysis (ICA) (Hyvärinen et al., 2001), have previously been derived for rate-based models (Hyvärinen & Oja, 1996, 1998). However, an application to spiking neurons has still been missing. In this section,

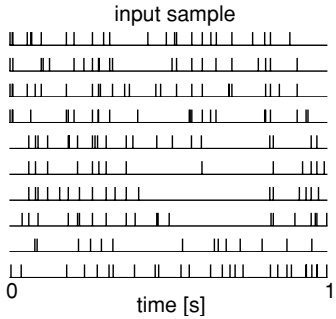

Figure 11: Demonstration of the difficulty of the ICA task for spike trains. Shown are 10 spike trains for 1 s that represent 10 out of 100 inputs in the experiment described in Figure 12. The top four spike trains (group 1) are correlated with a correlation coefficient of 0.5, as are spike trains 5–8 (group 2). However, spike trains from different groups are uncorrelated. The remaining bottom two input spike trains (group 3) are uncorrelated. Obviously it is quite difficult to detect which spike trains are correlated due to their rather weak correlation.

we have presented an ICA rule for spiking neurons that is able to detect statistical dependencies between the input rates and also between the timing of individual spikes, as shown in the experiment in Figure 12. Furthermore, while in ICA one usually assumes that the data are generated by a linear combination of statistically independent sources, we do not assume any model on how the data are generated. The experiment in Figure 12 also shows that our learning rule performs blind source separation even if the sources are not linearly mixed (which is not possible for a spiking input where the information is encoded in spike timings).

7 Discussion

Information bottleneck (IB) and independent component analysis (ICA) have been proposed as principles for unsupervised learning in lower cortical areas; however, learning rules that can implement these principles with spiking neurons have been missing. So far, synaptic update rules optimizing information-theoretic objectives have been presented mainly for rate models and real-valued units (Linsker, 1989; Bell & Sejnowski, 1995; Becker, 1996). In this article we have derived from information-theoretic principles learning rules that enable a stochastically spiking neuron to solve these tasks. We have shown in section 4.3 that these rules can be viewed as an extension to the classical Bienenstock-Cooper-Munro (BCM) rule (Bienenstock et al., 1982) and to its generalized variant for spiking neurons (Toyoizumi et al., 2005). Furthermore, we have demonstrated how they are related to

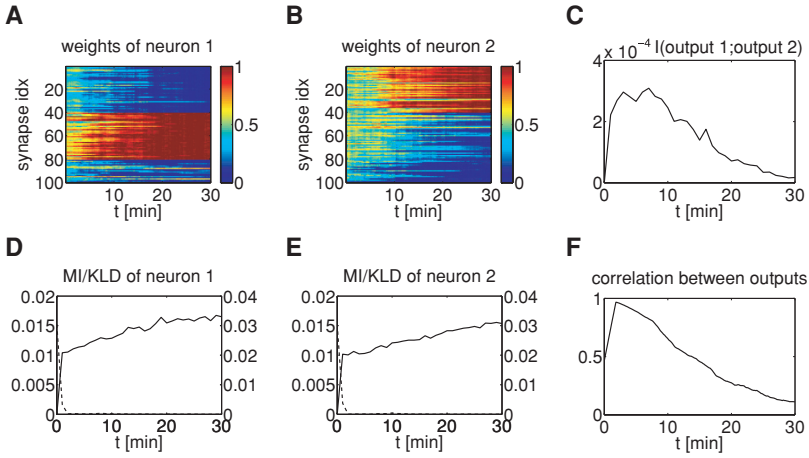

Figure 12: Extracting independent components from 100 input spike trains. (A, B) Evolution of weights during 30 minutes of learning for both postsynaptic neurons (red: strong synapses, $w_{ij} \approx 1$; blue: depressed synapses, $w_{ij} \approx 0$.) Weights were initialized randomly between 0.10 and 0.12, $\alpha = 5 \cdot 10^{-4}$, $\beta = 100$, $\gamma = 50$. (C) Evolution of the average mutual information per time bin between both output spike trains as a function of time. (D, E) Evolution of the average mutual information per time bin (solid line, left scale) between input and output and the Kullback-Leibler divergence per time bin for both neurons (dashed line, right scale) as a function of time. Averages are calculated over segments of 1 minute. (F) Trace of the current correlation between both output spike trains during learning. Correlation coefficients are calculated every 10 seconds.

traditional IB algorithms (see section 4.4) and neural ICA learning rules (see section 6.3). Our learning rules, which are optimal from the perspective of information theory, are not local in the sense that they use only information that is available at a single synapse without an auxiliary network of interneurons or other biological processes. But they tell us what type of information would have to be ideally provided by such auxiliary network and how the synapse should change its efficacy in order to approximate a theoretically optimal learning rule.

The learning rule for ICA that we have derived appears to be the first ICA learning rule for spiking neurons. We have demonstrated in Figures 11 and 12 that in particular, this learning rule enables spiking neurons to discover and remove dependencies in their input spike trains that are not encoded through correlations or other dependencies between their firing rates, but through correlations between the timing of individual spikes. But this ICA rule is also able to remove dependencies in firing rates.

Information bottleneck optimization is another and potentially more powerful method for deriving rules for learning that might shape the output

of projection neurons that send selected information to higher cortical areas or downward to the thalamus. In contrast to ICA, IB optimization need not be driven exclusively by the statistics of sensory input signals. Rather, IB optimization allows the selection of information from sensory inputs that is related to inputs from another sensory modality, to proprioceptive feedback, to expectations, or to rewards. Hence, it may contribute to the emergence of synergistic internal codes for relevant parts of the external world, which combine information from different sensory modalities (see Calvert, Spence, & Stein, 2004), causing in particular effects such as improved understanding of spoken language if the face of the speaker can be observed and to goal-oriented and task-dependent sensory processing (Sigala & Logothetis, 2002; Shuler & Bear, 2006; Fritz, Shamma, Elhilali, & Klein, 2003). Hence, IB learning rules share aspects of both unsupervised and supervised[5] learning. We have demonstrated through five computer experiments that the IB learning rules for spiking neurons that we have derived are capable of extracting information simultaneously from rates and from spike trains (see; Figures 6, 8, and 9), to extract input signals that are only partially related to the target signal (since the target is a sum of several input signals; see Figure 7), and to extract information that is related to two simultaneously presented target signals (which encode information in two different ways; see Figure 9). We have also demonstrated in Figure 10 that the learning rule can learn to extract information from the input that is not correlated with the target signal, but is related through higher-order statistical dependencies. Finally, we have demonstrated that the learning rules that we have derived work quite fast—in most cases, within a few minutes. We have also demonstrated that they are very stable (hence, do not require any regulation of learning rates), since their performance does not degrade during experiments of long duration. Furthermore, the firing rate of the learning neurons always stays within the desired range. In future work, it would be interesting to investigate applications of these learning rules to signal processing problems (e.g., noise filtering), since the IB approach promises to provide optimal solutions to some of these tasks.

The results of this article show only that biological neurons could in principle carry out ICA and IB analysis, and we have shown how close-to-optimal learning rules for spiking neurons would look like. We also have argued that both learning principles are very useful for any multisensory distributed cognitive system. This article poses the challenge

[5]Note that maximizing the mutual information between the output of a neuron and a target signal offers an interesting alternative to supervised learning for neurons, where the "code" that the neuron uses is left unspecified. While in a supervised learning task the target output is prescribed and should be reproduced as exactly as possible by the learning unit, this is not the case for an Infomax problem. Rather, information bottleneck can be viewed as a supervised selection of what is relevant, while the learning process itself and the choice of neural codes is unsupervised.

to neurophysiology to test through experiments in vivo and in vitro to what extent (and where) these learning principles are implemented in neural systems and how they are implemented through synaptic plasticity.

Appendix A: Details of the Derivations of the Learning Rules

A.1 Evaluation of Firing and Joint Firing Probabilities. To quantify the information between output spike trains $\mathbf{Y}_1^K$ and $\mathbf{Y}_2^K$ of length $K\Delta t$, we need an expression for the joint probability $P(Y_1^K, Y_2^K)$. For given input spike trains $X^k = (X_1^k, \ldots, X_N^k)$ up to time step k and postsynaptic spike history Y_i^{k-1}, we can write the probability of emitting a postsynaptic spike in the kth time step using the firing probability ρ_i^k, equation 2.2, as the binary distribution:

$$P\left(y_i^k \mid Y_i^{k-1}, X^k\right) = \left(\rho_i^k\right)^{y_i^k} \left(1 - \rho_i^k\right)^{(1-y_i^k)}. \tag{A.1}$$

The marginal probability, given only the postsynaptic history, can be written as

$$P\left(y_i^k \mid Y_i^{k-1}\right) = \left(\bar{\rho}_i^k\right)^{y_i^k} \left(1 - \bar{\rho}_i^k\right)^{(1-y_i^k)}, \tag{A.2}$$

where $\bar{\rho}_i^k = \langle \rho_i^k \rangle_{\mathbf{X}^k | Y_i^{k-1}} = \sum_{X^k} \rho_i^k P(X^k | Y_i^{k-1})$ is the average firing probability in the kth time step (where ρ_i^k depends, of course, on X^k and Y_i^k). The probability of an entire output spike train Y_i^K given the input X^K is then obtained by

$$P\left(Y_i^K \mid X^K\right) = \prod_{k=1}^{K} P\left(y_i^k \mid Y_i^{k-1}, X^k\right) \tag{A.3}$$

and, analogously, the probability of an output spike train by

$$P\left(Y_i^K\right) = \prod_{k=1}^{K} P\left(y_i^k \mid Y_i^{k-1}\right). \tag{A.4}$$

If two neurons receive the same input at their synapses and produce outputs Y_1^K and Y_2^K, we can write the joint probability of spiking in the kth time step given the postsynaptic histories and the input as

$$P\left(y_1^k, y_2^k \mid Y_1^{k-1}, Y_2^{k-1}, X^k\right) = P\left(y_1^k \mid Y_1^{k-1}, X^k\right) P\left(y_2^k \mid Y_2^{k-1}, X^k\right). \tag{A.5}$$

The marginal probability given only the postsynaptic histories can be written using equation A.1 as

$$\begin{aligned} &P(y_1^k, y_2^k | Y_1^{k-1}, Y_2^{k-1}) \\ &\quad = \left\langle \left(\rho_1^k\right)^{y_1^k} \left(1-\rho_1^k\right)^{(1-y_1^k)} \left(\rho_2^k\right)^{y_2^k} \left(1-\rho_2^k\right)^{(1-y_2^k)} \right\rangle_{\mathbf{X}^k | Y_1^{k-1}, Y_2^{k-1}} \\ &\quad = (\bar{\rho}_{12}^k)^{y_1^k y_2^k} \left(\bar{\rho}_1^k - \bar{\rho}_{12}^k\right)^{y_1^k (1-y_2^k)} (\bar{\rho}_2^k - \bar{\rho}_{12}^k)^{(1-y_1^k) y_2^k} \\ &\qquad \times \left(1 - \bar{\rho}_1^k - \bar{\rho}_2^k + \bar{\rho}_{12}^k\right)^{(1-y_1^k)(1-y_2^k)}, \end{aligned} \tag{A.6}$$

where $\bar{\rho}_i^k = \langle \rho_i^k \rangle_{\mathbf{X}^k | Y_1^{k-1}, Y_2^{k-1}} = \sum_{X^k} \rho_i^k P(X^k | Y_1^{k-1}, Y_2^{k-1})$ is the average firing probability of neuron i, given the postsynaptic history of both neurons, and $\bar{\rho}_{12}^k = \langle \rho_1^k \rho_2^k \rangle_{\mathbf{X}^k | Y_1^{k-1}, Y_2^{k-1}}$ is the average product of firing probabilities of both neurons. The joint probability of two entire output spike trains is then finally given as

$$P\left(Y_1^K, Y_2^K\right) = \prod_{k=1}^{K} P\left(y_1^k, y_2^k | Y_1^{k-1}, Y_2^{k-1}\right). \tag{A.7}$$

A.2 Evaluation of the Gradient of ΔL_{12}^k. We have to calculate the gradient $\partial \Delta L_{12}^k / \partial w_{1j}$, with

$$\Delta L_{12}^k = \left\langle \beta \log \frac{P\left(y_1^k, y_2^k \mid Y_1^{k-1}, Y_2^{k-1}\right)}{P\left(y_1^k \mid Y_1^{k-1}\right) P\left(y_2^k \mid Y_2^{k-1}\right)} \right\rangle_{\mathbf{X}^k, \mathbf{Y}_1^k, \mathbf{Y}_2^k}. \tag{A.8}$$

The strategy for the derivation is similar as for the terms considered in Toyoizumi et al. (2005), but a number of details are different.

We treat the nominator and the two product terms of the denominator in equation A.8 separately. The average of an arbitrary function f_w with arguments x, y_1, and y_2 is by definition

$$\begin{aligned} \left\langle f_w(x, y_1, y_2) \right\rangle_{\mathbf{x}, \mathbf{y}_1, \mathbf{y}_2} &= \sum_{x, y_1, y_2} p_w(x, y_1, y_2) f_w(x, y_1, y_2) \\ &= \left\langle \sum_{y_1} p_w(y_1 \mid x) f_w(x, y_1, y_2) \right\rangle_{\mathbf{x}, \mathbf{y}_2}, \end{aligned} \tag{A.9}$$

where $p_w(x, y_1, y_2) = p(x) p(y_2 \mid x) p_w(y_1 \mid x)$ denotes the joint probability of the triple (x, y_1, y_2) to occur, assuming that $\mathbf{y}_1$ is independent of $\mathbf{y}_2$ given $\mathbf{x}$. The subscript w indicates that both the probability distribution p_w and the function f_w depend on an additional parameter w.

Taking the derivative with respect to w, the product rule yields two terms,

$$\frac{\partial}{\partial w}\left\langle f_w(x, y_1, y_2)\right\rangle_{\mathbf{x},\mathbf{y}_1,\mathbf{y}_2} = \left\langle \sum_{y_1} p_w(y_1|x)\frac{\partial}{\partial w} f_w(x, y_1, y_2)\right\rangle_{\mathbf{x},\mathbf{y}_2} + \left\langle \sum_{y_1} \frac{\partial}{\partial w} p_w(y_1|x) f_w(x, y_1, y_2)\right\rangle_{\mathbf{x},\mathbf{y}_2}, \quad \text{(A.10)}$$

where the first term contains the derivative of the function f_w and the second term contains the derivative of the conditional probability p_w. Since

$$\frac{\partial}{\partial w} p_w(y_1 \mid x) = p_w(y_1 \mid x)\frac{\partial}{\partial w}\log p_w(y_1 \mid x), \quad \text{(A.11)}$$

the right-hand side of equation A.10 evaluates to

$$\left\langle \frac{\partial}{\partial w} f_w(x, y_1, y_2)\right\rangle_{\mathbf{x},\mathbf{y}_1,\mathbf{y}_2} + \left\langle \left[\frac{\partial}{\partial w}\log p_w(y_1|x)\right] f_w(x, y_1, y_2)\right\rangle_{\mathbf{x},\mathbf{y}_1,\mathbf{y}_2}, \quad \text{(A.12)}$$

that is, it can be written as an average over the joint distribution of x, y_1, and y_2.

Now we can evaluate each of the terms of equation A.8 using A.12. Considering the term $\frac{\partial}{\partial w_{1j}}\langle \log P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})\rangle_{\mathbf{X}^k,\mathbf{Y}_1^k,\mathbf{Y}_2^k}$ first, we get

$$\left\langle \frac{\partial}{\partial w_{1j}}\log P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})\right\rangle_{\mathbf{X}^k,\mathbf{Y}_1^k,\mathbf{Y}_2^k} + \left\langle \left[\frac{\partial}{\partial w_{1j}}\log P(Y_2^k|X^k)\right]\log P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})\right\rangle_{\mathbf{X}^k,\mathbf{Y}_1^k,\mathbf{Y}_2^k}. \quad \text{(A.13)}$$

We find that the first term of equation A.13 vanishes because

$$\begin{aligned}
&\left\langle \frac{\partial}{\partial w_{1j}}\log P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})\right\rangle_{\mathbf{X}^k,\mathbf{Y}_1^k,\mathbf{Y}_2^k} = \\
&= \left\langle \left\langle \frac{\partial}{\partial w_{1j}}\log P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})\right\rangle_{\mathbf{y}_1^k,\mathbf{y}_2^k|\mathbf{Y}_1^{k-1},\mathbf{Y}_2^{k-1}}\right\rangle_{\mathbf{Y}_1^{k-1},\mathbf{Y}_2^{k-1}} \\
&= \left\langle \sum_{y_1^k,y_2^k}\left[\frac{\partial}{\partial w_{1j}}\log P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})\right] P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})\right\rangle_{\mathbf{Y}_1^{k-1},\mathbf{Y}_2^{k-1}} \\
&= \left\langle \frac{\partial}{\partial w_{1j}}\left[\sum_{y_1^k,y_2^k} P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})\right]\right\rangle_{\mathbf{Y}_1^{k-1},\mathbf{Y}_2^{k-1}} = 0. \quad \text{(A.14)}
\end{aligned}$$

In the second line of equation A.14, we drop the expectation over $\mathbf{X}^k$, since the argument of the expectation operator is independent of the input spike train X^k, and use the identity $\langle\cdot\rangle_{\mathbf{Y}_1^k,\mathbf{Y}_2^k} = \langle\langle\cdot\rangle_{\mathbf{y}_1^k,\mathbf{y}_2^k|Y_1^{k-1},Y_2^{k-1}}\rangle_{\mathbf{Y}_1^{k-1},\mathbf{Y}_2^{k-1}}$. With the same argument, it can be shown that

$$\left\langle \frac{\partial}{\partial w_{1j}} \log P(y_i^k|Y_i^{k-1}) \right\rangle_{\mathbf{X}^k,\mathbf{Y}_1^k,\mathbf{Y}_2^k} = \left\langle \frac{\partial}{\partial w_{1j}} \log P(y_i^k|Y_i^{k-1}, X^k) \right\rangle_{\mathbf{X}^k,\mathbf{Y}_1^k,\mathbf{Y}_2^k} = 0 \tag{A.15}$$

for $i = 1, 2$. Hence, the only term that gives a nontrivial contribution in equation A.13 is the second one. With an analogous evaluation for the other terms in equation A.8, we finally have

$$\frac{\partial}{\partial w_{1j}} \Delta L_{12}^k = \left\langle \left[\frac{\partial}{\partial w_{1j}} \log P\left(Y_1^k \mid X^k\right) \right] \times \log \frac{P\left(y_1^k, y_2^k \mid Y_1^{k-1}, Y_2^{k-1}\right)}{P\left(y_1^k \mid Y_1^{k-1}\right) P\left(y_2^k \mid Y_2^{k-1}\right)} \right\rangle_{\mathbf{X}^k,\mathbf{Y}_1^k,\mathbf{Y}_2^k}. \tag{A.16}$$

Now we can identify the factors

$$C_{1j}^k := \frac{\partial}{\partial w_{1j}} \log P(Y_1^k|X^k) = \sum_{l=1}^{k} \left[\frac{y_1^l}{\rho_1^l} - \frac{1-y_1^l}{1-\rho_1^l} \right] \frac{\partial \rho_1^l}{\partial u_1} \sum_{n=1}^{l} \epsilon(t^l - t^n) x_j^n \tag{A.17}$$

and

$$\begin{aligned} F_{12}^k &:= \log \frac{P(y_1^k, y_2^k|Y_1^{k-1}, Y_2^{k-1})}{P(y_1^k|Y_1^{k-1}) P(y_2^k|Y_2^{k-1})} \\ &= y_1^k y_2^k \log \frac{{}^{=}\rho_{12}^k}{\bar{\rho}_1^k \bar{\rho}_2^k} + y_1^k (1-y_2^k) \log \frac{{}^{=}\rho_1^k - {}^{=}\rho_{12}^k}{\bar{\rho}_1^k - \bar{\rho}_1^k \bar{\rho}_2^k} + \\ &\quad + (1-y_1^k) y_2^k \log \frac{{}^{=}\rho_2^k - {}^{=}\rho_{12}^k}{\bar{\rho}_2^k - \bar{\rho}_1^k \bar{\rho}_2^k} \\ &\quad + (1-y_1^k)(1-y_2^k) \log \frac{1 - {}^{=}\rho_1^k - {}^{=}\rho_2^k + {}^{=}\rho_{12}^k}{1 - \bar{\rho}_1^k - \bar{\rho}_1^k + \bar{\rho}_1^k \bar{\rho}_2^k}. \end{aligned} \tag{A.18}$$

For computational reasons, we approximate the sum $\sum_{l=1}^{k}$ in the correlation term C_{1j}^{k}, equation A.17, by an exponential window with time constant $\tau_C = 1$ s (Toyoizumi et al., 2005):

$$C_{1j}^{k} = C_{1j}^{k-1}\left(1 - \frac{\Delta t}{\tau_C}\right) + \sum_{n=1}^{k} \epsilon(t^k - t^n) x_j^n \frac{g'(u_1(t^k))}{g(u_1(t^k))}\left[y_1^k - \rho_1^k\right]. \quad \text{(A.19)}$$

Furthermore, if we make the assumption ${}^{=}\rho_i^k = \bar{\rho}_i^k$ (see section A.3) we can simplify the term F_{12}^k, equation A.18, and write $\bar{\rho}_i^k = \bar{g}_i(t^k) R_i(t^k) \Delta t$ and ${}^{=}\rho_{12}^k = \bar{g}_{12}(t^k) R_1(t^k)\ R_2(t^k)(\Delta t)^2$ with $\bar{g}_i(t^k) = \langle g(u_i(t^k))\rangle_{\mathbf{X}^k|Y_i^{k-1}}$ and $\bar{g}_{12}(t^k) = \langle g(u_1(t^k)) g(u_2(t^k))\rangle_{\mathbf{X}^k|Y_1^{k-1},Y_2^{k-1}}$. Using the approximation $\log(1-x) \approx -x$ for small x, we get

$$\begin{aligned} F_{12}^k = {} & y_1^k y_2^k \log \frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)\bar{g}_2(t^k)} - y_1^k\left(1 - y_2^k\right) R_2(t^k)\Delta t \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k)\right] - \\ & - \left(1 - y_1^k\right) y_2^k R_1(t^k)\Delta t \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k)\right] + \\ & + \left(1 - y_1^k\right)\left(1 - y_2^k\right) R_1(t^k) R_2(t^k)(\Delta t)^2 \left[\bar{g}_{12}(t^k) - \bar{g}_1(t^k)\bar{g}_2(t^k)\right]. \end{aligned} \quad \text{(A.20)}$$

This approximation is valid for small Δt.

The weight change is then finally given by

$$\Delta \tilde{w}_{1j}^k = \alpha \left\langle C_{1j}^k \beta F_{12}^k \right\rangle_{\mathbf{X}^k, \mathbf{Y}_1^k, \mathbf{Y}_2^k}. \quad \text{(A.21)}$$

A.3 A Closer Look at the Firing Probabilities ${}^{=}\rho_i^k$ and $\bar{\rho}_i^k$. For simplicity, we assume $i = 1$. Using equations A.1 and A.2 we can write for ρ_1^k and $\bar{\rho}_1^k$:

$$\rho_1^k = P(y_1^k = 1 | X^k, Y_1^{k-1}), \quad \text{and} \quad \text{(A.22)}$$

$$\bar{\rho}_1^k = P(y_1^k = 1 | Y_1^{k-1}). \quad \text{(A.23)}$$

From ${}^{=}\rho_1^k = \langle \rho_1^k \rangle_{\mathbf{X}^k|Y_1^{k-1},Y_2^{k-1}}$ we find that

$${}^{=}\rho_1^k = P\left(y_1^k = 1 \mid Y_1^{k-1}, Y_2^{k-1}\right). \quad \text{(A.24)}$$

Qualitatively, any difference between $\bar{\rho}_1^k$ and ${}^{=}w\rho_1^k$ arises from the additional information that, given the postsynaptic history Y_1^{k-1}, the output of the other neuron, Y_2^{k-1}, conveys about a postsynaptic event at time step k. For a learning rule that uses the term F_{12}^k, equation 3.7, we have to calculate ${}^{=}\rho_i^k$ online. The average firing probabilities $\bar{\rho}_i^k = \langle \rho_i^k \rangle_{\mathbf{X}^k|Y_i^{k-1}}$ are implemented as running averages of ρ_i^k, as in Toyoizumi et al. (2005).

We can express ${}^{=}\rho_1^k$, equation A.24, using $\bar{\rho}_1^k$, A.23, that is,

$$ {}^{=}\rho_1^k = \bar{\rho}_1^k \cdot \frac{P(Y_2^{k-1}|y_1^k = 1, Y_1^{k-1})}{P(Y_2^{k-1}|Y_1^{k-1})}. \tag{A.25} $$

The second factor in equation A.25 is hard to evaluate online. However, if we assume that $y_1^k = 1$ is independent from Y_2^{k-1} given Y_1^{k-1}—that $P(y_1^k = 1, Y_2^{k-1}|Y_1^{k-1}) = P(y_1^k = 1|Y_1^{k-1})P(Y_2^{k-1}|Y_1^{k-1})$—we can set ${}^{=}\rho_1^k = \bar{\rho}_1^k$. In this case, since

$$ \bar{\rho}_1^k = \left\langle {}^{=}\rho_1^k \right\rangle_{\mathbf{Y}_2^{k-1}|Y_1^{k-1}}, \tag{A.26} $$

we replace ${}^{=}\rho_1^k$ by its mean value with respect to the distribution $P(Y_2^{k-1}|Y_1^{k-1})$.

A.4 Derivation of the Simplified Learning Rule. The starting point for the derivation for this simplified model is the weight update rule, equation 3.8,

$$ \begin{aligned} \frac{\Delta w_{1j}^k}{\Delta t} &= -\alpha \left\langle C_{1j}^k B_1^k(-\gamma) + \alpha\beta \Delta t C_{1j}^k B_{12}^k \right\rangle_{\mathbf{Y}_1^k, \mathbf{Y}_2^k, \mathbf{X}^k} \\ &= -\alpha \left\langle \left\langle C_{1j}^k B_1(-\gamma) \right\rangle_{\mathbf{Y}_1^k|X^k} + \alpha\beta \Delta t \left\langle C_{1j}^k B_{12}^k \right\rangle_{\mathbf{Y}_1^k, \mathbf{Y}_2^k|X^k} \right\rangle_{\mathbf{X}^k}, \end{aligned} \tag{A.27} $$

where in contrast to equation 3.8, we consider the batch version of the learning rule in which the weight update is averaged over the input and output distribution.

For notational convenience, we write the correlation term, equation A.17, as[6]

$$ \begin{aligned} C_{1j}^k &= \sum_{l=1}^{k} \left[\frac{y_1^l}{\rho_1^l} - \frac{1 - y_1^l}{1 - \rho_1^l} \right] \frac{\partial \rho_1^l}{\partial u_1} \sum_{n=1}^{l} \epsilon(t^l - t^n) x_j^n \\ &= \sum_{l=1}^{k} \left[y_1^l - \rho_1^l \right] \frac{(\rho_1^l)'}{\rho_1^l (1 - \rho_1^l)} \sum_{n=1}^{l} \epsilon(t^l - t^n) x_j^n \\ &= \sum_{l=1}^{k} K_1(l) \left[y_1^l - \rho_1^l \right], \end{aligned} $$

[6] For simplicity, we write $g(u)$ instead of $g_{alt}(u)$ throughout this section.

with

$$K_1(l) = \frac{(\rho_1^l)'}{\rho_1^l(1-\rho_1^l)} \sum_{n=1}^{l} \epsilon(t^l - t^n) x_j^n \approx \frac{g'(u_1(t^l))}{g(u_1(t^l))} \sum_{n=1}^{l} \epsilon(t^l - t^n) x_j^n.$$

Here, we used the approximation $\rho_1^l \approx g(u_1(t^l))\Delta t$, which holds for small Δt. Furthermore, we write the postsynaptic term as

$$B_1^k(-\gamma) = \frac{y_1^k}{\Delta t} B_{1A}^k + (1 - y_1^k) B_{1B}^k, \tag{A.28}$$

with

$$B_{1A}^k = \log\left[\frac{g(u_1(t^k))}{\bar{g}_1(t^k)} \left(\frac{\bar{g}_1(t^k)}{\tilde{g}}\right)^{\gamma}\right],$$
$$B_{1B}^k = -R_1(t^k)[g(u_1(t^k)) - (1-\gamma)\bar{g}_1(t^k) - \gamma\tilde{g}].$$

Since $\langle y_i^k \rangle_{\mathbf{Y}_i^k | X^k} = \rho_i^k$ and $\langle y_i^l y_i^k \rangle_{\mathbf{Y}_i^k | X^k} = \delta_{lk}\rho_i^k + \rho_i^l \rho_i^k (1 - \delta_{lk})$, where δ_{lk} is the Kronecker delta function, we get

$$\begin{aligned}
\langle C_{1j}^k B_1^k(-\gamma) \rangle_{\mathbf{Y}_1^k | X^k} &= \left\langle \sum_{l=1}^{k} K_1(l)[y_1^l - \rho_1^l] \left(\frac{y_1^k}{\Delta t} B_{1A}^k + (1 - y_1^k) B_{1B}^k\right) \right\rangle_{\mathbf{Y}_1^k | X^k} \\
&= \sum_{l=1}^{k} K_1(l)\delta_{lk} \left[\frac{\rho_1^k}{\Delta t} B_{1A}^k - \frac{(\rho_1^k)^2}{\Delta t} B_{1A}^k \right. \\
&\quad \left. - \rho_1^k B_{1B}^k + (\rho_1^k)^2 B_{1B}^k\right].
\end{aligned} \tag{A.29}$$

Since $\rho_i^k \approx g(u_i(t^k))\Delta t$, we get

$$\begin{aligned}
\langle C_{1j}^k B_1^k(-\gamma) \rangle_{\mathbf{Y}_1^k | X^k} &= K_1(k) \left[g(u_1(t^k)) B_{1A}^k - g(u_1(t^k))^2 \Delta t B_{1A}^k \right. \\
&\quad \left. - g(u_1(t^k))\Delta t B_{1B}^k + g(u_1(t^k))^2 (\Delta t)^2 B_{1B}^k\right] \\
&= K_1(k) \left[g(u_1(t^k)) B_{1A}^k + O(\Delta t)\right] \\
&\approx K_1(k) g(u_1(t^k)) B_{1A}^k,
\end{aligned} \tag{A.30}$$

where we assume small Δt. Substitution of $K_1(k)$ and B_{1A}^k yields

$$\langle C_{1j}^k B_1^k(-\gamma) \rangle_{\mathbf{Y}_1^k | X^k} \approx \nu_j^{pre,k} f(v_1^k) \log\left[\frac{\nu_1^k}{\bar{\nu}_1^k} \left(\frac{\bar{\nu}_1^k}{\tilde{g}}\right)^{\gamma}\right], \tag{A.31}$$

where the presynaptic rate at synapse j is denoted by $\nu_j^{pre,k} = a \sum_{n=1}^{k} \epsilon(t^k - t^n) x_j^n$ with a in units $(\mathrm{Vs})^{-1}$, and $\bar{\nu}_1^k$, $\bar{\nu}_2^k$, $\bar{\nu}_{12}^k$ are running averages of the output rate ν_1^k, the target rate ν_2^k, and the product of these values, $\nu_1^k \nu_2^k$. The rate ν_1^k is given directly by $g_{alt}(u_1(t^k))$. The function $f(\nu_1^k) = g'(g^{-1}(\nu_1^k))/a$ is proportional to the derivative of g with respect to u, evaluated at the current membrane potential.

For evaluation of the second term in equation A.27, we write B_{12}^k as

$$B_{12}^k = \frac{y_1^k y_2^k}{(\Delta t)^2} D_{12}^k - \frac{y_1^k (1 - y_2^k)}{\Delta t} D_1^k - \frac{(1 - y_1^k) y_2^k}{\Delta t} D_2^k + (1 - y_1^k)(1 - y_2^k) D_0, \tag{A.32}$$

with

$$\begin{aligned} D_{12}^k &= \log \frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)\bar{g}_2(t^k)}, \\ D_1^k &= \frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k), \\ D_2^k &= \frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k), \\ D_0^k &= \bar{g}_{12}(t^k) - \bar{g}_1(t^k)\bar{g}_2(t^k). \end{aligned}$$

We get

$$\begin{aligned} \left\langle C_{1j}^k B_{12}^k \right\rangle_{\mathbf{Y}_1^k, \mathbf{Y}_2^k | X^k} = \sum_{l=1}^{k} K_1(l) \Bigg\langle \left[y_1^l - \rho_1^l \right] \Bigg[& \frac{y_1^k y_2^k}{(\Delta t)^2} D_{12}^k - \\ & - \frac{y_1^k}{\Delta t} D_1^k + \frac{y_1^k y_2^k}{\Delta t} D_1^k - \frac{y_2^k}{\Delta t} D_2^k + \frac{y_1^k y_2^k}{\Delta t} D_2^k - \\ & - D_0^k + y_1^k D_0^k + y_2^k D_0^k - y_1^k y_2^k D_0^k \Big] \Bigg\rangle_{\mathbf{Y}_1^k, \mathbf{Y}_2^k | X^k}. \end{aligned}$$

For given input X^k, the two spike trains Y_1^k and Y_2^k are independent and $\langle y_1^l y_2^k \rangle_{\mathbf{Y}_1^k, \mathbf{Y}_2^k | X^k} = \rho_1^l \rho_2^k$. Furthermore, we use $\langle y_1^l y_1^k y_2^k \rangle_{\mathbf{Y}_1^k, \mathbf{Y}_2^k | X^k} = \langle y_1^l y_1^k \rangle_{\mathbf{Y}_1^k | X^k} \rho_2^k$, to get

$$\begin{aligned} \left\langle C_{1j}^k B_{12}^k \right\rangle_{\mathbf{Y}_1^k, \mathbf{Y}_2^k | X^k} &= K_1(k) g(u_1(t^k)) [g(u_1(t^k)) D_{12}^k - D_1^k + O(\Delta t)] \\ &\approx K_1(k) g(u_1(t^k)) [g(u_2(t^k)) D_{12}^k - D_1^k] \\ &= \nu_j^{pre}(t^k) f(\nu_1(t^k)) \left[\nu_2^k \log \frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k \bar{\nu}_2^k} - \left(\frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k} - \bar{\nu}_2^k \right) \right]. \end{aligned} \tag{A.33}$$

Again, the approximation is valid for small Δt.

Substitution of equations A.31 and A.33 into A.27 yields

$$\frac{\Delta w_{1j}^k}{\Delta t} = -\alpha \nu_j^{pre,k} f(\nu_1^k) \left\{ \log \left[\frac{\nu_1^k}{\bar{\nu}_1^k} \left(\frac{\bar{\nu}_1^k}{\tilde{g}} \right)^\gamma \right] - \beta \Delta t \left(\nu_2^k \log \left[\frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k \bar{\nu}_2^k} \right] - \bar{\nu}_2^k \left[\frac{\bar{\nu}_{12}^k}{\bar{\nu}_1^k \bar{\nu}_2^k} - 1 \right] \right) \right\}, \tag{A.34}$$

where the expectation $\langle \cdot \rangle_{\mathbf{x}}$ in equation A.27 is approximated by averaging over a single long trial under the assumption of a small learning rate α.

Appendix B: Whitening Transform

For the whitening transform used in the experiment described in section 5.5, we define a vector $\mathbf{x}(t) = [r_1(t) - \langle r_1 \rangle, r_2(t) - \langle r_2 \rangle]^T$, where $r_1(t)$ and $r_2(t)$ are the signals that should be whitened (the rate modulations of one input and the target signal in this case). Note that the averages of both signals are subtracted as to make $\mathbf{x}$ have zero mean. Furthermore, let $\mathbf{C} = E\{\mathbf{x}\mathbf{x}^T\}$ denote the 2-by-2 covariance matrix of $\mathbf{x}$, which is calculated in the simulations as the empirical covariance matrix of 10 s samples of $r_1(t)$ and $r_2(t)$. The whitening transform is then given by

$$\mathbf{T} = \mathbf{E}\mathbf{D}^{-\frac{1}{2}}\mathbf{E}^T, \tag{B.1}$$

where $\mathbf{E}$ is the orthogonal matrix of eigenvectors of $\mathbf{C}$ and $\mathbf{D}$ is the diagonal matrix of its eigenvalues, $\mathbf{D} = \mathrm{diag}(\lambda_1, \lambda_2)$ (i.e., $\mathbf{C} = \mathbf{E}\mathbf{D}\mathbf{E}^T$.). With this transformation, the vectors $\mathbf{Tx}$ have unit variance; in order to scale them back to the variance of the original signals, we define an additional scaling matrix $\mathbf{S} = \mathrm{diag}(\sigma_1, \sigma_2)$, where σ_1 and σ_2 are the standard deviations of r_1 and r_2, respectively. With the means added back, which have been subtracted before, the total transformation is then given by

$$\tilde{\mathbf{x}} = \mathbf{STx} + \begin{bmatrix} \langle r_1 \rangle \\ \langle r_2 \rangle \end{bmatrix}, \tag{B.2}$$

where the elements of $\tilde{\mathbf{x}}(t) = [\tilde{r}_1(t), \tilde{r}_2(t)]^T$ are uncorrelated.

Appendix C: Derivation of the Approximation in Section 6.1

Remember that the combined postsynaptic term of the learning rule of neuron i, equation 6.2, can be written as

$$A_i^k := B_i^k(\gamma) - \beta \Delta t B_{12}^k, \tag{C.1}$$

where

$$B_i^k(\gamma) = \frac{y_i^k}{\Delta t} \log \left[\frac{g(u_i(t^k))}{\bar{g}_i(t^k)} \left(\frac{\tilde{g}}{\bar{g}_i(t^k)} \right)^\gamma \right] - (1 - y_i^k) R_i(t^k) \left[g(u_i(t^k)) - (1+\gamma)\bar{g}_i(t^k) + \gamma \tilde{g} \right], \tag{C.2}$$

and

$$\begin{aligned} B_{12}^k = {} & \frac{y_1^k y_2^k}{(\Delta t)^2} \log \frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)\bar{g}_2(t^k)} - \frac{y_1^k}{\Delta t}(1 - y_2^k) R_2(t^k) \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k) \right] \\ & - \frac{y_2^k}{\Delta t}(1 - y_1^k) R_1(t^k) \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k) \right] \\ & + (1 - y_1^k)(1 - y_2^k) R_1(t^k) R_2(t^k) \left[\bar{g}_{12}(t^k) - \bar{g}_1(t^k)\bar{g}_2(t^k) \right]. \end{aligned} \tag{C.3}$$

For simplicity we consider only neuron 1 in the following; symmetric arguments apply for the case of neuron 2. We can distinguish four postsynaptic states for both neurons in each time step k: one where both are spiking ($y_1^k = y_2^k = 1$), one where neither of them emits a spike ($y_1^k = y_2^k = 0$), and two cases where only one of them fires ($y_1^k = 1$, $y_2^k = 0$, and $y_1^k = 0$, $y_2^k = 1$, respectively). For these four cases, the postsynaptic term, equation C.1, evaluates to

- $y_1^k = y_2^k = 1$:

$$A_1^k = \frac{1}{\Delta t} \log \left[\frac{g(u_1(t^k))}{\bar{g}_1(t^k)} \left(\frac{\tilde{g}}{\bar{g}_1(t^k)} \right)^\gamma \right] - \frac{1}{\Delta t} \beta \log \frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)\bar{g}_2(t^k)}, \tag{C.4}$$

- $y_1^k = 1$, $y_2^k = 0$:

$$A_1^k = \frac{1}{\Delta t} \log \left[\frac{g(u_1(t^k))}{\bar{g}_1(t^k)} \left(\frac{\tilde{g}}{\bar{g}_1(t^k)} \right)^\gamma \right] + \beta R_2(t^k) \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k) \right], \tag{C.5}$$

- $y_1^k = 0$, $y_2^k = 1$:

$$A_1^k = -R_1(t^k) \left[g(u_1(t^k)) - (1+\gamma)\bar{g}_1(t^k) + \gamma \tilde{g} \right] + \beta R_1(t^k) \left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k) \right], \tag{C.6}$$

- $y_1^k = y_2^k = 0$:

$$A_1^k = -R_1(t^k)\left[g(u_1(t^k)) - (1+\gamma)\bar{g}_1(t^k) + \gamma\tilde{g}\right] \\ -\beta\Delta t R_1(t^k)R_2(t^k)\left[\bar{g}_{12}(t^k) - \bar{g}_1(t^k)\bar{g}_2(t^k)\right]. \tag{C.7}$$

We want to model the contribution of the term B_{12}^k, equation C.3, by changing the value $g(u_1(t^k))$. That is, we again apply the simple postsynaptic BCM term,

$$\hat{B}_1^k(\gamma) = \frac{y_1^k}{\Delta t}\log\left[\frac{\hat{g}_1(t^k)}{\bar{g}_1(t^k)}\left(\frac{\tilde{g}}{\bar{g}_1(t^k)}\right)^{\gamma}\right] \\ -\left(1-y_1^k\right)R_1(t^k)\left[\hat{g}_1(t^k) - (1+\gamma)\bar{g}_1(t^k) + \gamma\tilde{g}\right], \tag{C.8}$$

instead of the combined postsynaptic term A_1^k, equation C.1, in the learning rule of neuron 1, but encapsulate the effect of the term B_{12}^k in changing the gain $g(u_1(t^k))$ into $\hat{g}_1(t^k)$ in this simple postsynaptic term $\hat{B}_1^k$.

We look for arithmetic expressions for $\hat{g}_1(t^k)$ by comparing formula C.8 with equations C.4 to C.7. We get

- $y_1^k = y_2^k = 1$:

$$\hat{g}_1(t^k) = g(u_1(t^k))\left(\frac{\bar{g}_1(t^k)\bar{g}_2(t^k)}{\bar{g}_{12}(t^k)}\right)^{\beta}, \tag{C.9}$$

- $y_1^k = 1$, $y_2^k = 0$:

$$\hat{g}_1(t^k) = g(u_1(t^k))\exp\left[R_2(t^k)\beta\Delta t\left(\frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k)\right)\right], \tag{C.10}$$

- $y_1^k = 0$, $y_2^k = 1$:

$$\hat{g}_1(t^k) = g(u_1(t^k)) - \beta\left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k)\right], \tag{C.11}$$

- $y_1^k = y_2^k = 0$:

$$\hat{g}_1(t^k) = g(u_1(t^k)) + R_2(t^k)\beta\Delta t\left[\bar{g}_{12}(t^k) - \bar{g}_1(t^k)\bar{g}_2(t^k)\right]. \tag{C.12}$$

However, Figure 3 suggests that significant effects of B_{12}^k are encountered only when one of the two neurons is firing; we also neglect the influence of simultaneous action potentials within the same time step as Δt gets small. Therefore, we focus only on cases C.10 and C.11 where exactly one of the two neurons is firing. The value $g(u_1(t^k))$ is then modified according to

$$\hat{g}_1(t^k) = g(u_1(t^k))\exp\left[R_2(t^k)\beta\Delta t\left(\frac{\bar{g}_{12}(t^k)}{\bar{g}_1(t^k)} - \bar{g}_2(t^k)\right)\right] \\ \text{if } y_1^k = 1,\ y_2^k = 0, \tag{C.13}$$

which corresponds to a multiplicative change, and

$$\hat{g}_1(t^k) = g(u_1(t^k)) - \tilde{\beta}\left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_2(t^k)} - \bar{g}_1(t^k)\right] \quad \text{if } y_2^k = 1,\ y_1^k = 0, \tag{C.14}$$

which corresponds to an additive change.

Summarizing, the modified value $\hat{g}_i(t^k)$ for neuron $i = 1, 2$ can be written as follows:

$$\hat{g}_i(t^k) = g(u_i(t^k)) \cdot a_i(t^k)^{y_i^k(1-y_{3-i}^k)} + b_i(t^k) y_{3-i}^k (1 - y_i^k). \tag{C.15}$$

The modulation terms $a_i(t^k)$ and $b_i(t^k)$ are given by

$$a_i(t^k) = \exp\left[R_{3-i}(t^k)\beta\Delta t\left(\frac{\bar{g}_{12}(t^k)}{\bar{g}_i(t^k)} - \bar{g}_{3-i}(t^k)\right)\right], \tag{C.16}$$

$$b_i(t^k) = -\beta\left[\frac{\bar{g}_{12}(t^k)}{\bar{g}_{3-i}(t^k)} - \bar{g}_i(t^k)\right]. \tag{C.17}$$

Acknowledgments

We thank Wulfram Gerstner and Jean-Pascal Pfister for helpful discussions. This article was written under partial support by the Austrian Science Fund FWF, S9102-N13 and P17229-N04, and was also supported by PASCAL, project IST2002-506778, and FACETS, project 15879, of the European Union.

References

Becker, S. (1996). Mutual information maximization: Models of cortical self-organization. *Network: Computation in Neural Systems*, *7*, 7–31.

Bell, A. J., & Sejnowski, T. J. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, *7*, 1129–1159.

Bienenstock, E. L., Cooper, L. N., & Munro, P. W. (1982). Theory for the development of neuron selectivity: Orientation specificity and binocular interaction in visual cortex. *J. Neurosci.*, *2*(1), 32–48.

Calvert, G., Spence, C., & Stein, B. (2004). *The handbook of multisensory processes*. Cambridge, MA: MIT Press.

Cover, T. M., & Thomas, J. A. (1991). *Elements of information theory*. New York: Wiley.

Fritz, J., Shamma, S., Elhilali, M., & Klein, D. (2003). Rapid task-related plasticity of spectrotemporal receptive fields in primary auditory cortex. *Nature Neuroscience*, *6*(11), 1216–1223.

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

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

Hyvärinen, A., Karhunen, J., & Oja, E. (2001). *Independent component analysis*. New York: Wiley.

Hyvärinen, A., & Oja, E. (1996). Simple neuron models for independent component analysis. *Int. Journal of Neural Systems*, *7*(6), 671–687.

Hyvärinen, A., & Oja, E. (1998). Independent component analysis by general nonlinear Hebbian-like learning rules. *Signal Processing*, *64*(3), 301–313.

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

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

Linsker, R. (1989). How to generate ordered maps by maximizing the mutual information between input and output signals. *Neural Computation*, *1*, 402–411.

Shuler, M. G., & Bear, M. F. (2006). Reward timing in the primary visual cortex. *Science*, *311*(5767), 1606–1609.

Sigala, N., & Logothetis, N. K. (2002). Visual categorization shapes feature selectivity in primate temporal cortex. *Nature*, *415*, 318–320.

Slonim, N. (2002). *The information bottleneck: Theory and applications*. Unpublished doctoral dissertation, Hebrew University, Jerusalem.

Tishby, N., Pereira, F. C., & Bialek, W. (1999). The information bottleneck method. In *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing* (pp. 368–377). Urbana: University of Illinois.

Toyoizumi, T., Pfister, J.-P., Aihara, K., & Gerstner, W. (2005). Generalized Bienenstock-Cooper-Munro rule for spiking neurons that maximizes information transmission. *Proc. Natl. Acad. Sci. USA*, *102*, 5239–5244.

Received January 8, 2007; accepted April 4, 2007.