

Training an STDP-Enabled Neuron with an Innocuous Teaching Signal

Nathan D. VanderKraats

Department of Computer and Information Science and Engineering

University of Florida

Gainesville, FL 32611

`ndv@cise.ufl.edu`

Subhajit Sengupta

Department of Computer and Information Science and Engineering

University of Florida

Gainesville, FL 32611

`ss5@cise.ufl.edu`

Arunava Banerjee

Department of Computer and Information Science and Engineering

University of Florida

Gainesville, FL 32611

`arunava@cise.ufl.edu`

December 4, 2008

Abstract

For a spiking neural system of multiple excitatory synapses and a single output neuron utilizing Spike Timing Dependent Plasticity (STDP), we investigate the effects of adding a single teaching input to train the system in a physiologically realistic fashion. This teaching signal, by directly affecting the output spike train, is made to indirectly affect the entire set of inputs' synaptic weights through STDP. Remarkably, this method is shown to increase the performance of an

output neuron executing a symbolic classification task on the inputs. Further, the resultant teaching signal is innocuous: statistically, it is virtually indistinguishable from a constant rate Poisson spike train over the duration of the inputs.

1 Introduction

Spike timing dependent plasticity (STDP) has been observed in many experimental situations, becoming the dominant theory of how synapses are updated by the neural signals they process. However, connecting STDP, a local phenomenon, to an overarching learning strategy for realistically-modeled spiking networks has heretofore remained elusive.

Any viable learning scheme for a system of neurons should contain a few assumptions. For one, STDP's effects should be solely local: one synapses' changes should not affect other synapses. Also, feedback within the system should be realistic. Rather than assuming a specific kind of code, such as rate coding [?] or synfire chains [?], general representation-independent spiketrains are desirable. Since cortical spiketrains are commonly known to have Poisson-like statistics [?], feedback in a model system should show similar statistics. Furthermore, the overall procedure should be biologically plausible, not relying on any analyses that are impossible for actual neurons.

Previous work in connecting STDP to global learning has begun by designating a system-wide objective function, then attempting to derive STDP using the assumed objective. For instance, Bell assumes that a neural system seeks to maximize network sensitivity [?] to increase the entropy of its outputs. Starting with this principle, he attempts to derive a local rule that bears similarity to STDP. Similarly, spiketrain variability [?] and information maximization [?] have been explored as global objective functions. Unfortunately, these attempts have fallen short in explaining the connection between STDP and learning, becoming mired in the inherent complexity of spiking systems. While the existence of a simple global objective function is attractive, a biological system may utilize an objective that is complicated or inefficient, meaning that there may be no clear mathematical principle from which to start.

1.1 A Reinforcement Learning Model

To overcome this issue, we explore a paradigm shift: rather than beginning with the system-level learning goals and attempting to derive STDP, we start by assuming STDP in a model spiking neuron. Using the task-based framework described in Chapter Decoder , a feedback loop is envisioned that mirrors the psychological notion of learning through reinforcement.

Constructing a complete neural learning loop is an ambitious task, and achieving this goal requires several nontrivial developments. The work of this proposal, therefore, focuses on a necessary piece of the larger solution. Assuming the existence of a teaching signal and knowledge of whether each training example is correctly classified, we address the question of whether a well-chosen teaching signal could push the system in a direction such that future classification is improved. Rather than allowing any arbitrary feedback signal, constraints are placed upon the teacher so that it will be compatible with the larger plan. For this proposal, the teacher is modified from a randomly-generated Poisson spiketrain, ensuring that the new teacher will be innocuous. A definition and justification of this innocuous signal is given in Section ???. Precisely how to push the system in the right direction with such a teacher, and how this affects learning in the system, is examined throughout Section ???.

The construction of the teaching signal, as well as the exact circuitry of the feedback loop, is beyond the scope of this proposal. However, research in closed-loop learning, insinuated by phenomena such as motor babbling, suggests that a teaching signal could be presented alongside the same stimulus that created it for the purposes of reinforcement learning [?].

2 Model of the Neuron and Inputs

The model is a simple network consisting of a single output neuron innervated by a large number of inputs and a single teaching input, as depicted in Figure ???. Spiking dynamics in the system are governed by the first-order Spike Response Model [?], which was described in Section ???. In all experiments, $\beta = 6$ (dimensionless), $\tau = 15msec$, $R = -1000mV$, and $\gamma = 1.6msec$. The synapses on the output neuron implement additive STDP [?] as detailed in Section ???. For this work, $\bar{g}_{max} = 40$ by default, $\tau_+ = \tau_- = 20msec$, and the values of A_+ and A_- will be given for each trial.

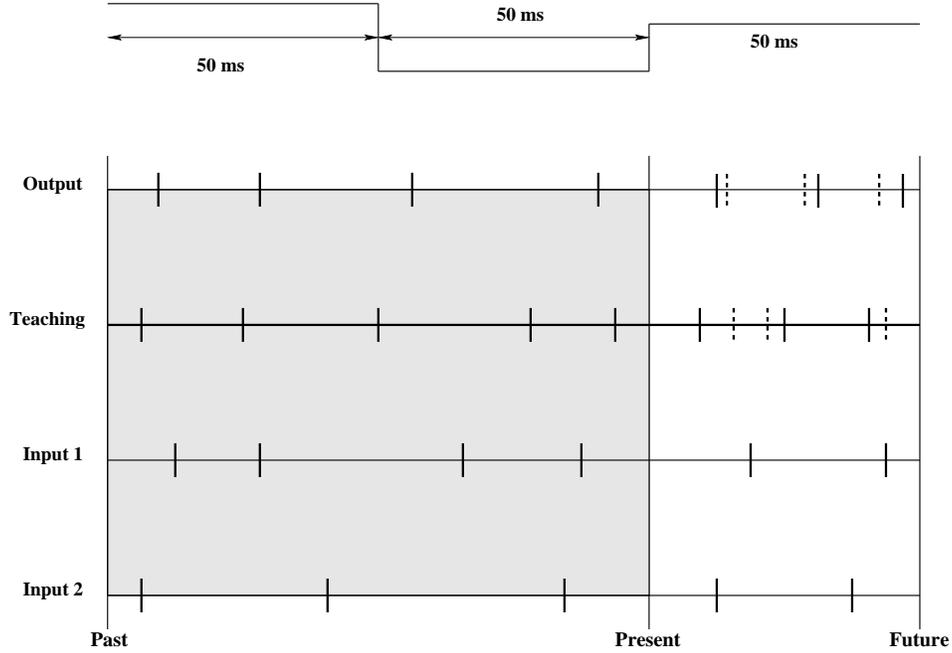


Figure 1: The two-pass teaching algorithm. Desired output perturbations, the dashed spikes in the output line, are used to find optimal teaching perturbations, the dashed teaching spikes. The simulation is then rerun using the new teaching spiketrain.

The synaptic weights between inputs and output are initialized randomly between 15 and 25. As necessary in additive STDP models, these values are bounded during runtime at a minimum value $\bar{g}_{min} = 1$ and a maximum value \bar{g}_{max} . Although additive STDP is used, the method generalizes easily to other forms of STDP [?].

For inputs to the model neuron, the Meddis Inner-Hair Cell model is again levied for the generation of realistic auditory nerve data. All experiments use 40 exponentially-increasing center frequencies from $100Hz$ to $5000Hz$. Two auditory nerve responses are created for each of the center frequencies, producing a total of 80 input neurons.

3 Definition of the Learning Task

3.1 Experimental Overview

Given the basic system above, consider a general framework that separates symbolic inputs using the output neuron, using the task-based discrimination method described in Chapter Decoder. Each symbolic input, then, corresponds to an equivalence class of spiketrains. Reinforcement is introduced into the system through a two-stage process. The input to the system is defined as a sequence of short, fixed length spike trains, each representing some semantic symbol. First, an input segment, plus the non-informative teaching input, is passed through the system, yielding an output spike train. Based on the method described in Section ??, the output is evaluated, producing a vector in the direction in which the output should be changed to maximally improve performance. Using this vector, changes are made to the teaching signal that would move the output in the desired direction given the same inputs. Finally, the simulation is rewound and rerun again using the modified teacher, effectively moving the system in the desired direction.

One important assumption is contained in this plan. At any point in time, the state of the system is described by its synaptic weights, which completely determine its spike responses to any input. The optimal correction for a given sample, therefore, would be achieved by directly changing the weights. Unfortunately, this manipulation is not biologically plausible, since the weights must only be changed through STDP. Because of this requirement, any system relying on a teacher for learning must make the assumption that output corrections alone are capable of pulling the synaptic weights in the right direction, on average.

Ideally, given any set of synaptic weights, a Bayesian classifier would be used to determine the most probable input class given any observed output spike train [?]. However, due to the variety of potential inputs, not to mention the astronomical number of synaptic weight combinations, creating the underlying distributions for a Bayesian method is clearly impractical. Therefore, an online classifier is desirable. An online technique has the additional benefit of being robust in the face of concept drift [?], where the underlying model slowly changes as examples are presented. In the case of a neural system with STDP, this drifting occurs because the synaptic weights that define the system are constantly updated. While an appraisal of the system could be performed by temporarily fixing the synaptic weights for an evaluation

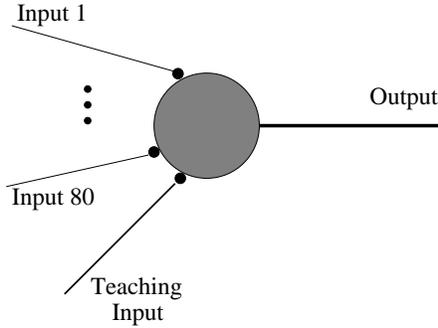


Figure 2: Output neuron with 80 input synapses and one teaching synapse.

period, this workaround should be avoided since no guarantees exist that the performance with a fixed weight set will mimic performance in the natural situation of ever-changing weights. Furthermore, such a fixed evaluation is not physiologically realizable.

As a consequence of the above, we make the assumption that moving an output spiketrain in a greedy direction, improving its own classifiability, will translate into synaptic weight movements that increase average performance for the system on the whole.

3.2 Learning Task

As an initial demonstration of the global learning capabilities of the method, consider a simple puretone frequency discrimination task, as discussed in Section ???. The auditory stimuli are 50msec iso-amplitudinal puretone blips ranging in frequency between 90Hz and 5200Hz . The classes are divided at 2645Hz , the midpoint of the range. While this task seems superficially simple, any dichotomy of frequencies can be obtained by asking a series of such high/low questions.

In some respects, working with symbolic tasks presents more of a challenge than the homogeneous modification rules examined by past researchers [?][?]. Specifically, any rule that specifies the relationship between a particular output spike and its input spikes operates merely by making a similar modification on each observation, without incorporating the performance of the system at that point in time. On the other hand, a task-based method utilizes actual feedback for arbitrary symbolic classes, rather than just ad-

justing the sensitivity of a group of neurons to a particular set of inputs.

3.3 Innocuous Reinforcement

To attain effective, biologically-plausible reinforcement, the nature of the teaching signal must be deliberated. For simple binary classification tasks, one could imagine a naïve teaching signal that spikes with an extremely high rate for class 1 and a low rate for class 0. However, such rate-based reinforcements are undesirable. For one, this teaching signal cannot be turned off without effect: its mere presence or absence denotes that some degree of reinforcement is occurring. Further, rate-based reinforcement lacks physiological appeal, and it is unclear whether such an approach can be generalized to arbitrary symbolic input classes.

Therefore, the teaching signal should encode its reinforcements temporally. First, for the non-informative first pass teaching input, a constant rate Poisson spiketrain is randomly generated. To make the signal informative for the second pass, the spikes in this teacher are slightly perturbed, incrementally changing the output spiketrains. Teaching signals of this nature will have very nearly Poisson spike distributions, reflecting biologically-observed spike statistics. Comparisons between some perturbed teaching signals and their base Poisson spiketrains are presented in Figures ?? and ??.

4 Teaching Method

4.1 Objective Function

As stated in Section ??, the ideal reinforcement strategy would be to derive an objective function of the synaptic weights of the input: $E(Q_1, \dots, Q_n)$. For every given set of synaptic weights, or Q -configuration, this function would relate how good or bad the system is at making its classifications. Then, a gradient descent over the synaptic weight space could locate the optimal discriminant. Unfortunately, this objective function seems impossible to determine directly. Even if one had access to the prodigious amounts of data necessary to estimate conditional probability distributions for every potential Q -configuration, enacting such a solution is physiologically unreasonable, since real neurons must operate in an online manner.

Since determining the gradient of the system itself is not possible for any

online scheme, the only information that can be utilized to direct a given training period is the output spiketrain. Therefore, we propose a method by which the system is taught to move each output spiketrain a small amount in a desirable direction, with the hope that this greedy decision will lead to a long-term performance gain for the system as a whole.

As in Section ??, each output modulation period is considered as a point in the spike times feature space. For the selection of the statistical classifier, a linear discriminant has several attractive properties, including ease of computation and physiological validity. Additionally, since a linear classifier produces a hyperplane boundary between positive and negative classes, the orientation of this hyperplane can be used to readily assess the performance of any given data point. A correction may be applied to a data point even when the point is correctly classified, which will serve to increase class separation.

Any incremental learning algorithm may be implemented as a classifier. For ease of use, we select the perceptron, as presented in Section ??, which is both online and linear [?]. The learning rate η is set to 0.001 by default. It should be noted that the use of the perceptron is only a tool to gain insight into the performance of the current system. The full input discrimination is performed by first multiplexing the input signals through the nonlinear output neuron, and then effecting the linear separation. Therefore, the overall discrimination power is superior to a linear classifier, being more reminiscent of a kernel-based technique.

The output corrections are applied in the following manner. First, the perceptron’s hyperplane is initialized. To avoid pathological convergence issues, a small portion of the training data is examined to ensure that the starting hyperplane is in the same feature space vicinity as the data set. The initial weight vector is:

$$\mathcal{W} = \mu(\mathbf{C}_1) - \mu(\mathbf{C}_2) \tag{1}$$

where $\mu(\mathbf{C}_i)$ denotes the mean of all points in the initial set belonging to class i . Likewise, the initial bias is set to:

$$\mathcal{B} = \frac{(\|\mu(\mathbf{C}_1)\|^2 - \|\mu(\mathbf{C}_2)\|^2)}{2} \tag{2}$$

After the initial set, each training output spiketrain is evaluated with respect to the current perceptron hyperplane. Regardless of whether the

point was correctly classified, a correction is computed, serving to separate the two classes on every example. This correction is simply a normalized vector orthogonal to the discriminating hyperplane:

$$\Delta \mathbf{y} = \frac{\mathcal{W}}{\|\mathcal{W}\|} z \quad (3)$$

where $z \in \{-1, +1\}$ is the correct class label of the current point. If the point is misclassified, the hyperplane is adjusted by the perceptron rules detailed in Section ??.

4.2 Optimal Teaching Perturbations

Using the sample output's correction vector, a suitable perturbation of the teaching spikes - and indirectly, the synaptic weights - can be calculated to produce the desired output perturbation. We emphasize that *all* of the synapses onto the output neuron are perturbed indirectly through the perturbations in the output spike train. As a first step, the output spikes and synaptic weight perturbations must be written as functions of the teaching perturbations.

The synaptic weights of the system are only updated when a spike is generated. The following calculations presuppose that the system is about to produce an output spike, denoting the impending output spike as y_{new} . Using the SRM with global threshold Υ :

$$\sum_{i=0}^n \sum_{j=1}^{M_i} Q_i^j * P_i(x_i^j) = \Upsilon \quad (4)$$

where n denotes the total number of neurons, and M_i signifies the total number of spikes for neuron i within the time window T . By convention, let $i = 0$ represent the output spiketrain, $i = 1$ denote the teaching input, and $i = 2 \dots n$ be the non-teaching inputs. Therefore, x_i^j is the time elapsed since the j^{th} most recent input spike for synapse $i = 1 \dots n$. Similarly, x_0^j denotes the time of the j^{th} most recent output spike. Finally, Q_i^j denotes the synaptic strength between input i and the output neuron at the time of spike x_i^j . These synaptic weight histories must be retained until the spike is no longer in the active window, as a consequence of the SRM. For compactness in the derivation, we define Q_0^j as a constant, 1, for all output spikes j .

Finally, $P_i(t)$ denotes the fixed PSP/AHP profile for a given spike on neuron i :

$$P_i(t) = \begin{cases} \frac{1}{d\sqrt{t}} e^{-\frac{\beta d^2}{t}} e^{-\frac{t}{\tau}} + C_{AHP} & \text{if } i \neq 0 \\ R e^{-\frac{t}{\tau}} + C_{PSP} & \text{if } i = 0 \end{cases} \quad (5)$$

where all model parameters other than t remain static. Specifically, C_{AHP} denotes the constant AHP response with respect to input spike movements, and C_{PSP} denotes the constant PSP response with respect to output spike movements. Now, to describe the output perturbation Δy_{new} caused by a set of past spike perturbations Δx_i^j and past synaptic weight perturbations ΔQ_i^j , for $i = 0 \dots n$, $j = 1 \dots M_i$, note that:

$$\sum_{i=0}^n \sum_{j=1}^{M_i} (Q_i^j + \Delta Q_i^j) * P_i(x_i^j + \Delta x_i^j - \Delta y_{new}) = \Upsilon \quad (6)$$

Applying a first-order Taylor expansion for $P_i(x_i^j)$:

$$\sum_{i,j} (Q_i^j + \Delta Q_i^j) * \left(P_i(x_i^j) + \frac{\partial P_i(x_i^j)}{\partial x_i^j} (\Delta x_i^j - \Delta y_{new}) \right) = \Upsilon \quad (7)$$

Rearranging, dropping all non-first-order terms, and noting the equality in Equation ??:

$$\sum_{i,j} \Delta Q_i^j P_i(x_i^j) + \sum_{i,j} Q_i^j \frac{\partial P_i(x_i^j)}{\partial x_i^j} (\Delta x_i^j - \Delta y_{new}) = \Upsilon \quad (8)$$

It follows that:

$$\Delta y_{new} = \frac{\sum_{i,j} P_i(x_i^j) \Delta Q_i^j + \sum_{i,j} Q_i^j \frac{\partial P_i(x_i^j)}{\partial x_i^j} \Delta x_i^j}{\sum_{i,j} Q_i^j \frac{\partial P_i(x_i^j)}{\partial x_i^j}} \quad (9)$$

Next, we must formulate the perturbation in a synaptic weight update as a function of all past perturbations of input spikes, output spikes, and synaptic weights. To avoid notational confusion, we denote this update perturbation as ΔR_i , with R_i representing the current weight between neuron i and the output. Using the notation from above with an impending spike y_{new} , a

positive synaptic update is denoted $R_i \rightarrow R_i + \bar{g}_{max} F_i$. The cumulative effect of all positive synaptic updates is thus:

$$F_i = \sum_{j=1}^{M_i} A_+ e^{\frac{-x_i^j}{\tau_+}} \quad (10)$$

where M_i denotes the total number of spikes for neuron i lying within the past STDP efficacy window, which is assumed for simplicity to be the same length as the spike efficacy time window T . For a set of input perturbations $\Delta x_i^j, j = 1 \dots M_i$ and output perturbation Δy_{new} as above, the new F_i will be:

$$F_i^{(pert)} = F_i + \Delta F_i = \sum_{j=1}^{M_i} A_+ e^{\frac{-(x_i^j + \Delta x_i^j - \Delta y_{new})}{\tau_+}} \quad (11)$$

Applying a first-order Taylor expansion for $e^{\frac{-(x_i^j + \Delta x_i^j - \Delta y_{new})}{\tau_+}}$, it follows that:

$$\Delta F_i = \sum_{j=1}^{M_i} -A_+ e^{\frac{-x_i^j}{\tau_+}} * (\Delta x_i^j - \Delta y_{new}) \left(\frac{1}{\tau_+} \right) \quad (12)$$

Now, the change in the update of R_i due to all perturbations, for positive updates only, is:

$$\begin{aligned} \Delta R_i &= \bar{g}_{max} F_i^{(pert)} - \bar{g}_{max} F_i \\ &= \bar{g}_{max} * \sum_{j=1}^{M_i} -A_+ e^{\frac{-x_i^j}{\tau_+}} * (\Delta x_i^j - \Delta y_{new}) \left(\frac{1}{\tau_+} \right) \end{aligned} \quad (13)$$

Where Δy_{new} , from Equation ??, is a function of fixed spike and weight perturbations. Similarly, the cumulative negative synaptic updates for a synapse i with impending input spike $x_{i,new}$ are:

$$\Delta R_i = \bar{g}_{max} * \sum_{j=1}^{M_i} -A_- e^{\frac{-x_0^j}{\tau_-}} * (\Delta x_0^j - \Delta x_{i,new}) \left(\frac{1}{\tau_-} \right) \quad (14)$$

where $\Delta x_{i,new}$ is the prescribed perturbation of the impending input spike.

As mentioned in Section ??, hard bounds are commonly added to STDP models to prevent unlimited reinforcement. When a synaptic update would result in a value beyond the prescribed range $[\bar{g}_{min}, \bar{g}_{max}]$, the synaptic strength R_i is set to the boundary, and ΔR_i is set to 0.

Next, perturbations to spikes on the teaching synapse, Δx_1^j for all j within the current window T must be applied so that the output is perturbed in the appropriate direction. Using Equation ??, the incremental change in each output spike, Δy^k , for all k within the output window, may be described as a function of the change in each teaching input spike Δx_1^j and the change in each synaptic weight ΔQ_1^j , forming a Jacobian matrix \mathbf{J} . The optimal teaching spike perturbations are then a solution to the linear system:

$$\begin{bmatrix} \frac{\partial y^1}{\partial x_1^1} & \frac{\partial y^1}{\partial x_1^2} & \cdots & \frac{\partial y^1}{\partial x_1^m} & \frac{\partial y^1}{\partial Q_1^1} & \frac{\partial y^1}{\partial Q_1^2} & \cdots & \frac{\partial y^1}{\partial Q_1^m} \\ \frac{\partial y^2}{\partial x_1^1} & \frac{\partial y^2}{\partial x_1^2} & \cdots & \frac{\partial y^2}{\partial x_1^m} & \frac{\partial y^2}{\partial Q_1^1} & \frac{\partial y^2}{\partial Q_1^2} & \cdots & \frac{\partial y^2}{\partial Q_1^m} \\ \vdots & \vdots & \ddots & \vdots & & & & \\ \frac{\partial y^n}{\partial x_1^1} & \frac{\partial y^n}{\partial x_1^2} & \cdots & \frac{\partial y^n}{\partial x_1^m} & \frac{\partial y^n}{\partial Q_1^1} & \frac{\partial y^n}{\partial Q_1^2} & \cdots & \frac{\partial y^n}{\partial Q_1^m} \end{bmatrix} \begin{bmatrix} \Delta x_1^1 \\ \Delta x_1^2 \\ \vdots \\ \Delta x_1^m \\ \Delta Q_1^1 \\ \Delta Q_1^2 \\ \vdots \\ \Delta Q_1^m \end{bmatrix} = \begin{bmatrix} \Delta y^1 \\ \Delta y^2 \\ \vdots \\ \Delta y^n \end{bmatrix} \quad (15)$$

The rows of this matrix can be constructed in an iterative fashion by considering a sliding window starting at $[-T, 0]$ and moving forward to $[0, T]$. Whenever a new output spike is encountered at the forward edge of the window, a new row is created in \mathbf{J} . Whenever an input spike is encountered, the synaptic weight structures are updated according to the negative updates of Equation ??.

Since this system may be overdetermined or underdetermined depending on the number of input and output spikes and their relative positions for a given modulation period, the system is solved using the Moore-Penrose pseudoinverse. This solution is desirable in the overdetermined case, because it will minimize the norm of the error $\|\mathbf{J}\Delta\mathbf{x} - \Delta\mathbf{y}\|$, and in the underdetermined case because it will yield the solution $\Delta\mathbf{x}$ that minimizes $\|\Delta\mathbf{x}\|$.

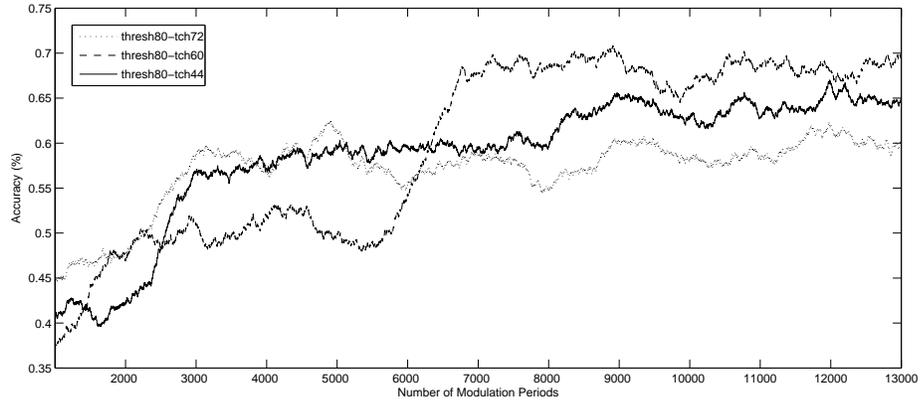


Figure 3: Moving average accuracy computed over 1000 modulation periods for runs initialized at various parameter values.

5 Results

5.1 Classification Improvement

Plotting a moving average allows visualization of the performance of a classifier that is changing over time. Figure ?? shows such a plot, using 1000 modulation periods per average. To avoid contamination of the accuracies by the teaching process, the performance is plotted before any teaching corrections occur.

Synaptic distances are initialized to random values between 1 and 2 for all experiments. Similarly, the synaptic weights are initialized randomly between 15 and 25. Each experiment uses a fixed threshold for the output neuron, and a constant spike rate for the initial Poisson teaching signal. Results for several experiments, each with different parameters, and random initializations are shown in Figure ?. For most parameter settings, the teaching signal was capable of driving the synaptic strengths of the neuron in a direction that improved classification.

5.2 Innocuousness

The differences between the original non-teaching Poisson spike train and the appropriately perturbed teaching spike train were hardly distinguish-

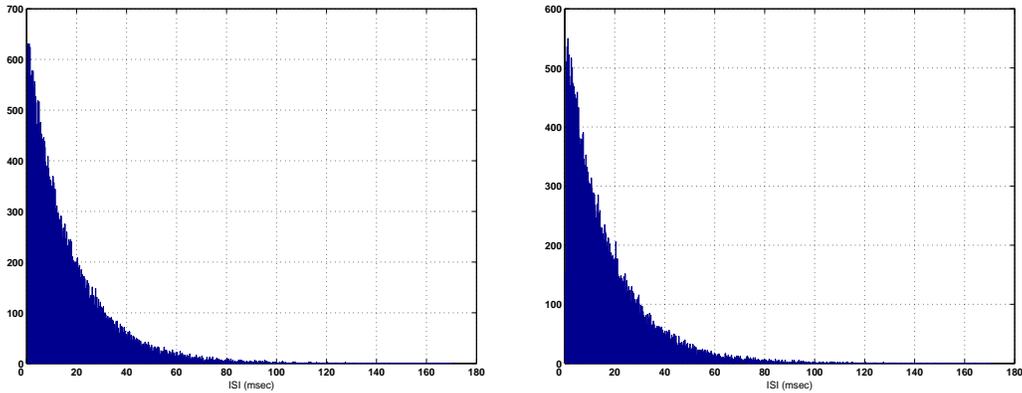


Figure 4: Histogram of the interspike interval (ISI) distribution of the original non-teaching Poisson spike train (left), and the appropriately perturbed teaching spike train (right). Note that the ISI distribution is largely undisturbed.

able. The *coefficient of variation* (CV) was 0.9665 for the Poisson spike train and 0.9651 for the teaching spike train. Visually, the interspike interval distributions were also barely differentiable, as shown in Figure ??). The conditional distribution function $p(x_i^t | x_{j,new})$, the probability of a spike occurring for neuron i at time t given a current spike on neuron j at time $t = 0$, fully determines the rate at which the synaptic strength drifts. As displayed in Figure ??, no statistically significant differences were found between the drift for the teaching synapses and the drift for non-teaching synapses.

6 Conclusion

While the preliminary results presented here have not yet been completely explored, the implications of the initial findings are quite intriguing. The evidence suggests a mechanism by which feedback could be used in the brain to learn an arbitrary symbolic task, using only STDP and the most basic spike time dynamics. Even more surprising is the conclusion that such signals may be virtually undetectable in biology, masquerading as mundane background noise.

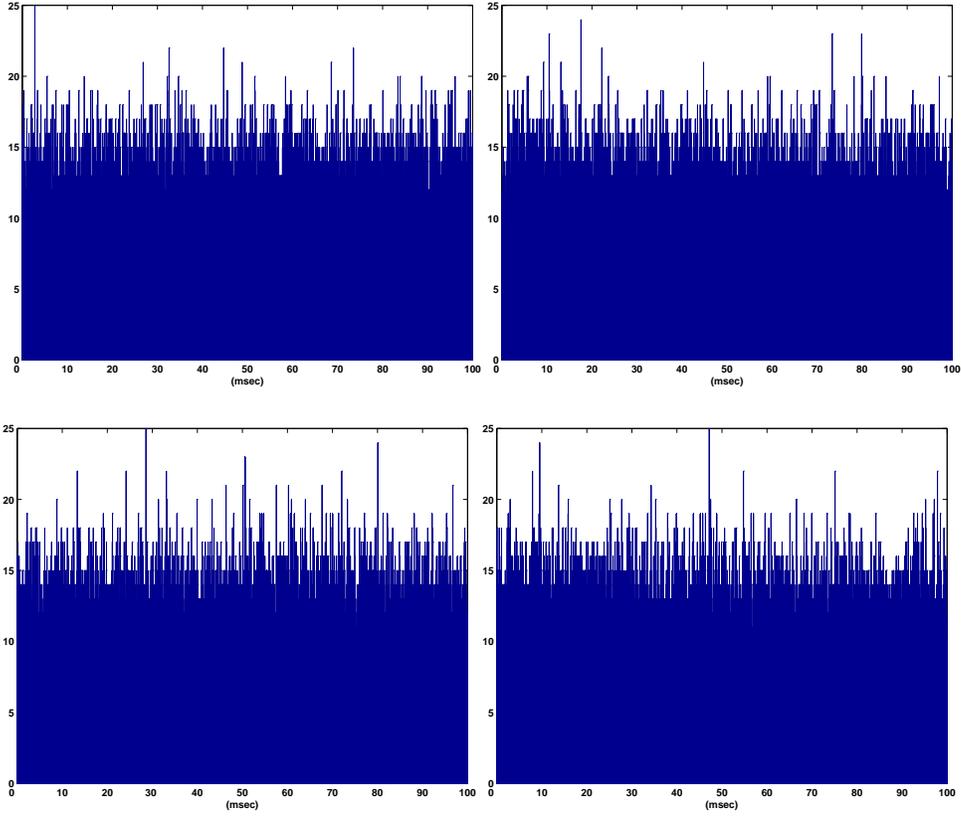


Figure 5: *Top left:* Histogram of the conditional probability $p(x_0^t|x_{j,new})$ for $j = 2 \dots 80$, the probability of finding an output spike t ms into the past given a non-teaching input spike at present. *Top right:* Histogram for $p(x_i^t|x_{0,new})$ for $i = 2 \dots 80$. *Bottom left and right:* Corresponding histograms for $j = 1$, $i = 1$, respectively, comparing output spikes to teaching input spikes.

References

- [1] Squire, L. R., Bloom, F. E., McConnell, S. K., Roberts, J. L., Spitzer, N. C., & Zigmond, M. J. (2003). *Fundamental neuroscience*. (2nd ed.). London: Academic Press.
- [2] Bear, M. F., Connors, B. W., & Paradiso, M. A. (2007). *Neuroscience: exploring the brain*. (3rd ed.). Baltimore: Lippincott Williams & Wilkins.
- [3] Banerjee, A. (2001). On the Phase-Space Dynamics of Systems of Spiking Neurons. I: Model and Experiments. *Neural Computation*, 13(1), 161-193.
- [4] Altschuler, R. A., Bobbin, R. P., Clopton, B. M., & Hoffman, D. W. (1991). *Neurobiology of hearing: The central auditory system*. New York: Raven Press.
- [5] Rieke, F., Warland, D., de Ruyter van Steveninck, R., & Bialek, W. (1997). *Spikes: Exploring the neural code*. Cambridge: MIT Press.
- [6] Dayan, P. & Abbott, L. F. (2001). *Theoretical neuroscience: computational and mathematical modeling of neural systems*. Cambridge: MIT Press.
- [7] Mainen, Z. F., & Sejnowski, T. J. (1995). Reliability of spike timing in neocortical neurons. *Science*, 268, 1503-1506.
- [8] Gerstner, W. & Kistler, W. (2002). *Spiking neuron models: single neurons, populations, plasticity*. Cambridge: Cambridge Univ. Press.
- [9] MacGregor, R. J., & Lewis, E. R. (1977). *Neural Modeling*. New York: Plenum Press.
- [10] Kistler, W., Gerstner, W., & van Hemmen, J. L. (1997). Reduction of the Hodgkin-Huxley equations to a threshold model. *Neural Computation*, 9, 1069-1100.
- [11] Bi, G. Q. & Poo, M. M. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neuro.*, 18, 10464-72.
- [12] Song, S., Miller, K., & Abbott, L. (2000). Competitive Hebbian learning through spike-timing-dependent plasticity. *Nat Neuro*, 3, 919-926.
- [13] van Rossum, M. C. W., Bi, G. Q., & Turrigiano, G. G. (2000). Stable Hebbian Learning from Spike Timing-Dependent Plasticity. *J. Neuro.*, 20(23), 8812-8821.

- [14] Moore, B. C. J. (2004). *An introduction to the psychology of hearing*. (5th ed). London: Academic Press.
- [15] Meddis, R. (1986). Simulation of mechanical to neural transduction in the auditory receptor. *J. Acoust. Soc. Am.*, 79, 702-711.
- [16] Meddis, R. (1988). Simulation of auditory-neural transduction: Further studies. *J. Acoust. Soc. Am.*, 83, 1056-1063.
- [17] Lopez-Poveda, E. A., OMard, L. P., & Meddis, R. (2001). A human nonlinear cochlear filterbank. *J. Acoust. Soc. Am.*, 110, 3107-3118.
- [18] Sumner, C. J., Lopez-Poveda, E. A., OMard, L. P., & Meddis, R. (2002). A revised model of the inner-hair cell and auditory nerve complex. *J. Acoust. Soc. Am.*, 111, 2178-2188.
- [19] <http://www.pdn.cam.ac.uk/groups/dsam/index.html>. Online.
- [20] Vapnik, V. N. (1999). *The Nature of Statistical Learning Theory*. (2nd ed). Berlin: Springer.
- [21] Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2, 121-167.
- [22] Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern Recognition*. New York: John Wiley & Sons.
- [23] Dvoretzky, A., Kiefer, J., & Wolfowitz, J. (1956). Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. *Ann. Math. Stat.*, 27(3), 642-669.
- [24] Shlens, J., Kennel, M. B., Abarbanel, H. D. I., & Chichilnisky, E. J. (2007). Estimating information rates with confidence intervals in neural spike trains. *Neural Computation*, 19, 1683-1719.
- [25] Cover, T. M., & Thomas, J. A. (2006). *Elements of Information Theory*. (2nd ed). New Jersey: Wiley.
- [26] Shannon, C. E. (1948) A mathematical theory of communication. *The Bell System Technical Journal*, 27, 379-423, 623-656.
- [27] Warland, D. K., Reinagel, P., & Meister, M. (1997). Decoding visual information from a population of retinal ganglion cells. *J. Neurophys*, 78, 2336-2350.

- [28] Wessberg, J., Stambaugh, C. R., Kralik, J. D., Beck, P. D., Laubach, M., Chapin, J. K., Kim, J., Biggs, S. J., Srinivasan, M. A., Nicolelis, M. A. (2000). Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature*, 408(6810), 361-5.
- [29] Serruya, M. D., Hatsopoulos, N. G., Paninski, L., Fellows, M. R., Donoghue, J. P. (2002). *Nature*, 416(6877), 141-2.
- [30] Wu, W., Gao, Y., Bienenstock, E., Donoghue, J. P., & Black, M. J. (2005). Bayesian population decoding of motor cortical activity using a Kalman filter. *Neural Comp*, 18, 80-118.
- [31] Nirenberg, S., Jacobs, A., Fridman, G., Latham, P., Douglas, R., Alam, N., & Prusky, G. (2006). Ruling out and ruling in neural codes. *J. Vision*, 6, 889.
- [32] Pouget, A., Dayan, P., Zemel, R. (2000). Information processing with population codes. *Nat Rev Neuro*, 1, 125-132.
- [33] Strong, S., Koberle, R., de Ruyter van Steveninck, R., & Bialek, W. (1998). Entropy and information in neural spike trains. *Phys Rev Lett*, 80, 197-200.
- [34] Kennel, M. B., Shlens, J., Abarbanel, H. D. I., & Chichilnisky, E. J. (2005). Estimating entropy rates with Bayesian confidence intervals. *Neural Computation*, 17, 1531-1576.
- [35] Wolpert, D. & Wolf, D. (1995). Estimating functions of probability distributions from a finite set of samples. *Phys Rev E*, 52, 6841-6854.
- [36] Miller, G. (1955). Note on the bias of information estimates. In H. Quastler (Ed.), *Information theory in psychology II-B* (pp. 95-100). Glencoe, IL: Free Press.
- [37] Paninski, L. Estimation of entropy and mutual information. (2003). *Neural Computation*, 15, 1191-1253.
- [38] Pola, G., Petersen, R. S., Thiele, A., Young, M. P., & Panzeri, S. (2005). Data-robust tight lower bounds to the information carried by spike times of a neuronal population. *Neural Computation*, 17, 1962-2005.
- [39] Montemurro, M. A., Senatore, R., & Panzeri, S. (2007). Tight data-robust bounds to mutual information combining shuffling and model selection techniques. *Neural Computation*, 19, 2913-2957.

- [40] Mehring, C., Hehl, U., Kubo, M., Diesmann, M., & Aertsen, A. (2003). Activity dynamics and propagation of synchronous spiking in locally connected random networks. *Biol Cyber*, 88, 395
- [41] Mazurek, M. E. & Shadlen, M. N. (2002). Limits to the temporal fidelity of cortical spike rate signals. *Nat Neuro*, 5, 463-471.
- [42] Diesmann, M., Gewaltig, M. O., & Aertsen, A. (1999). Stable propagation of synchronous spiking in cortical neural networks. *Nature*, 402(6761), 529-33.
- [43] Nirenberg, S., & Victor, J. D. (2007). Analyzing the activity of large populations of neurons: how tractable is the problem? *Curr Opin Neurobiol*, 17, 397-400.
- [44] Shlens, J., Field, G. D., Gauthier, J. L., Grivich, M. I., Petrusca, D., Sher, A., Litke, A. M., & Chichilnisky, E. J. (2006). The structure of multi-neuron firing patterns in primate retina. *J. Neuro*, 26, 8254-8266.
- [45] Schneidman, E. Berry, M. J. II, Segev, R., & Bialek, W. (2006). Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087), 1007-1012.
- [46] Latham, P. & Nirenberg, S. (2005). Synergy, redundancy, and independence in population codes, revisited. *J. Neuro*, 25, 5195-5206.
- [47] VanRullen, R., Guyonneau, R., & Thorpe, S. J. (2005). Spike times make sense. *Trends Neuro*, 28(1), 1-4.
- [48] Reich, D. S., Mechler, F., Purpura, K. P., & Victor, J. D. (2000). Interspike intervals, receptive fields, and information encoding in primary visual cortex. *J Neuro*, 20(5), 1964-1974.
- [49] Gollisch, T., & Meister, M. (2008). Rapid Neural Coding in the Retina with Relative Spike Latencies. *Science*, 319(5866), 1108-111.
- [50] Joachims, T. (1999). Making large-scale SVM learning practical. In Schölkopf, B., Burges, C., & Smola, A. (Ed.), *Advances in kernel methods - support vector learning*. Cambridge: MIT Press.
- [51] Joachims, T. (2002). *Learning to classify text using support vector machines*. Norwell: Kluwer Press.
- [52] <http://svmlight.joachims.org/>. Online.

- [53] Fletcher, N. H., & Rossing, T. D. (2005). *The physics of musical instruments*. New York: Springer.
- [54] Bell, A. J. & Parra, L. C. (2004). Maximising sensitivity in a spiking network. *Advances in Neural Information Processing Systems 17*, Cambridge: MIT Press.
- [55] Bohte, S. M. & Mozer, M. C. (2004). Reducing spike train variability: A computational theory of spike-timing dependent plasticity. *Advances in Neural Information Processing Systems 17*, Cambridge: MIT Press.
- [56] Chechik, G. (2003). Spike-timing dependent plasticity and relevant mutual information maximization. *Neural Computation*, 15(7), 1481-1510.
- [57] Zelaznik, H. N. (1996). *Advances in Motor Learning and Control*. Champaign: Human Kinetics.
- [58] Widmer, G., & Kubat, M. (1996). Learning in the presence of concept drift and hidden contexts. *Machine Learning*, 23, 69-101.
- [59] Wellner, J. A. (1992). Empirical processes in action: a review. *Inter. Stat. Rev.*, 60(3), 247-269.
- [60] DeStefano, J., & Learned-Miller, E. (2008). A probabilistic upper bound on differential entropy. *IEEE Transactions on Information Theory*, Under Revision.