J Neurophysiol 92: 2274–2282, 2004.
First published April 7, 2004; 10.1152/jn.01133.2003.

Temporal Sparseness of the Premotor Drive Is Important for Rapid Learning in a Neural Network Model of Birdsong

Ila R. Fiete,[1,4] Richard H.R. Hahnloser,[5] Michale S. Fee,[3,4] and H. Sebastian Seung[2,4]

[1]*Department of Physics, Harvard University, Cambridge 02138;* [2]*Howard Hughes Medical Institute,* [3]*McGovern Institute for Brain Research, and* [4]*Department of Brain & Cognitive Sciences, Massachusetts Institute of Technology, Cambridge Massachusetts 02139; and* [5]*Institute for Neuroinformatics, Universität Zürich/Eidgenössische Technische Hochschule Zürich, 8057 Zurich Switzerland*

Submitted 25 November 2003; accepted in final form 2 April 2004

Fiete, Ila R., Richard H. R. Hahnloser, Michale S. Fee, and H. Sebastian Seung. Temporal sparseness of the premotor drive is important for rapid learning in a neural network model of birdsong. *J Neurophysiol* 92: 2274–2282, 2004. First published April 7, 2004; 10.1152/jn.01133.2003. Sparse neural codes have been widely observed in cortical sensory and motor areas. A striking example of sparse temporal coding is in the song-related premotor area high vocal center (HVC) of songbirds: The motor neurons innervating avian vocal muscles are driven by premotor nucleus robustus archistriatalis (RA), which is in turn driven by nucleus HVC. Recent experiments reveal that RA-projecting HVC neurons fire just one burst per song motif. However, the function of this remarkable temporal sparseness has remained unclear. Because birdsong is a clear example of a *learned* complex motor behavior, we explore in a neural network model with the help of numerical and analytical techniques the possible role of sparse premotor neural codes in song-related motor learning. In numerical simulations with nonlinear neurons, as HVC activity is made progressively less sparse, the minimum learning time increases significantly. Heuristically, this slowdown arises from increasing interference in the weight updates for different synapses. If activity in HVC is sparse, synaptic interference is reduced, and is minimized if each synapse from HVC to RA is used only once in the motif, which is the situation observed experimentally. Our numerical results are corroborated by a theoretical analysis of learning in linear networks, for which we derive a relationship between sparse activity, synaptic interference, and learning time. If songbirds acquire their songs under significant pressure to learn quickly, this study predicts that HVC activity, currently measured only in adults, should also be sparse during the sensorimotor phase in the juvenile bird. We discuss the relevance of these results, linking sparse codes and learning speed, to other multilayered sensory and motor systems.

INTRODUCTION

Birdsong is a complex, learned motor behavior driven by a discrete set of premotor brain nuclei with well-studied anatomy. Neural activity, too, has been characterized in these nuclei, through recordings in awake singing birds, making the birdsong circuit a uniquely rich and accessible system for the study of motor coding and learning.

Juvenile male songbirds learn their songs from adult male tutors of the same species. Singing is used for courtship and territorial displays, and in evolutionary terms is an important skill for birds to master. A zebra finch song consists of 3 to 5 identical repetitions of an approximately 1-s song motif.

Syringeal and respiratory motoneurons responsible for song are driven by precisely executed sequences of neural activity in the premotor nucleus robustus archistriatalis (RA) (Simpson and Vicario 1990) of songbirds. Activity in RA is driven by excitatory feedforward inputs from the forebrain nucleus high vocal center (HVC, Bottjer et al. 1984; Nottebohm et al. 1982, 1976), whose RA-projecting neural population displays temporally sparse, precise, and stereotyped sequential activity. Individual RA-projecting HVC neurons burst just once in an entire approximately 1-s song motif (″unary” coding), and fire almost no spikes elsewhere in the motif (Hahnloser et al. 2002). Each HVC burst is of high firing rate (600–700 Hz) and typically lasts for about 6 ms. Burst-onset times for different RA-projecting HVC neurons are distributed across the motif. However, each HVC neuron bursts reliably at precisely the same time point (referenced to some acoustic landmark in the motif) in repeated renditions of the motif.

Song learning is thought to involve plasticity of synapses from HVC to RA. This is because these synapses display anatomical evidence of extensive synaptic growth and redistribution (Herrmann and Arnold 1991; Sakaguchi and Saito 1996) and physiological evidence of synaptic change and maturation (Mooney 1992; Stark and Perkel 1999) during the critical period. The temporal sparseness of HVC activity implies that these HVC–RA synapses are used in a very special manner during song: that is, each synapse is used during only one instant in the motif. Is there any functional significance to this way of using synapses? Here we investigate the possibility that it facilitates song learning.

Intuitively, the situation where each synapse participates in the production of just one short part of the motif seems ideally suited for minimizing interference between different synapses during learning. In this paper we make the intuitive argument more concrete through both computer simulations and mathematical analysis of a simple neural network model of birdsong learning.

It has been observed that interference between synapses can hinder learning in artificial *recurrent* neural networks (Hermann et al. 1995; Meunier et al. 1991; Tsodyks and Feigelman 1988; Willshaw et al. 1969). Because of the multilayered architecture of the song motor system, we are here motivated to study the effect in a *feedforward* multilayer network.

Experiments indicate that the young bird uses the mismatch or error between its own vocalizations and a desired song template (an internally stored copy of a tutor song) to iteratively modify its song to match the template (Brainard and

Address for reprint requests and other correspondence: I. R. Fiete, Kavli Institute for Theoretical Physics, Kohn Hall, UC Santa Barbara, Santa Barbara, CA 93106 (E-mail: prasad@kitp.ucsb.edu).

The costs of publication of this article were defrayed in part by the payment of page charges. The article must therefore be hereby marked “*advertisement*” in accordance with 18 U.S.C. Section 1734 solely to indicate this fact.

Doupe 2000; Konishi 1965). Thus, the goal of learning in our feedforward model network (of HVC, RA, and an output motor layer), is to alter the initial output sequence of motor activity by gradual adjustment of the HVC-to-RA weights, until the output sequence matches a specified desired sequence (Doya and Sejnowski 1995; Troyer and Doupe 2000).

It is not known how the brain translates goal-directed problems such as song imitation into prescriptions for synaptic change, although it is thought that if distance to the goal is quantified in a reward (error) function, neural and synaptic changes may occur in directions that increase the reward (decrease error), thus performing hill-climbing on the function. A common computational approach in modeling this phenomenon is to define such an error function, then move on the error surface toward the minimum along the gradient, or direction of steepest descent. This can be done by direct gradient calculation in single-layer networks, or by backpropagation, a simple technique for gradient descent in multilayer networks. Hill climbing can also be achieved by more biologically plausible learning algorithms that perform a stochastic approximation of gradient following without needing to explicitly compute the gradient (Bartlett and Baxter 1999; Seung 2003; Williams 1992). For simplicity, we apply learning by direct gradient following (backpropagation). Because the various gradient-based learning rules described above are in a mathematically similar class, we expect sparseness arguments made in the context of one learning rule to generalize to the others in the same class.

METHODS

General framework

We study a multilayer feedforward network (Fig. 1) with an HVC layer that provides sequential inputs to the network and drives activity in the hidden layer RA; the output layer of motor units is driven by activity in RA. HVC activities are written as $h_i(t)$, RA activities as $r_j(t)$, and output activities as $o_k(t)$, with

$$r_j(t) = f\Big[\sum_{i=1}^{N_h} W_{ji} h_i(t) - \theta_j\Big] \tag{1}$$

and

$$O_k(t) = \sum_{j=1}^{N_r} A_{kj} r_j(t) \tag{2}$$

where N_h, N_r, and N_o are the numbers of units in the HVC, RA, and motor layers, respectively; f is the activation function of RA neurons; and θ_j is the threshold for the jth RA neuron. The plastic weights from HVC to RA are given by the matrix *W;* because there is no direct evidence of plasticity in the connections from RA to the motor neurons, we take these weights to be fixed, and represent them by a fixed weight matrix *A.*

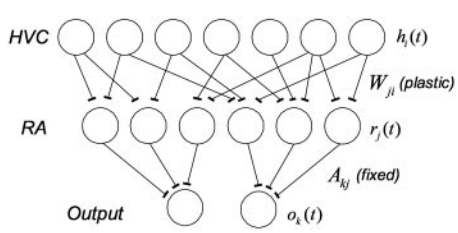

FIG. 1. High vocal center (HVC), robustus archistriatalis (RA), and the output layer are arranged with feed-forward plastic weights *W* from HVC to RA, and fixed weights *A* from RA to the output. HVC activities provide sequential inputs to the network, and the output units are the read-outs. RA neurons form a "hidden" layer.

Observational evidence suggests that vocal motor learning in the zebra finch segments roughly into 2 phases: first, a temporal motor sequence is established, and later the notes and syllables occurring in that motor sequence become more distinct, diversified, and refined (Tchernichovski et al. 2001). In that the goal of this work is to study the effects of HVC sparseness on the learning of feedforward premotor representations, we do not deal with the formation of sequences within HVC; instead we focus on the formation and refinement of HVC–motor representations as seen in the latter phase. The sequential patterns of HVC activities and desired output activities are externally imposed (see below for numerical details) in our simulations, and do not change throughout learning; the goal of the network is to learn to match the actual outputs $o_k(t)$ of the network, driven by HVC activity, with the desired outputs $\mathrm{d}_k(t)$, through adjustment of the plastic weights *W*. In one pass through the song motif, called an *epoch,* the network outputs are computed from *Eqs. 1* and *2*. The total network error for that epoch is determined from the objective function

$$C = \int_0^T dt \sum_{k=1}^{N_o} [\mathrm{d}_k(t) - o_k(t)]^2 \tag{3}$$

For learning, network weights *W* are adjusted after each epoch to minimize this cost function according to the backpropagation gradient-descent rule

$$\Delta W_{ji} = -\eta \frac{\partial C}{\partial W_{ji}} = \eta \int_0^T \mathrm{d}t \sum_{k=1}^{N_o} 2[\mathrm{d}_k(t) - o_k(t)] A_{kj} f_j' h_i \tag{4}$$

where f_j' is the derivative of the activation function of RA neuron *j*, and the parameter η scales the overall size of the weight update.

Numerical details of nonlinear network simulations

We simulate learning in the network described above, with N_h = 500 HVC neurons, N_r = 800 RA neurons, and N_o = 2 output units. Assuming that each HVC neuron bursts *B* times per motif, activity for the *i*th HVC neuron is fixed by choosing *B* onset times $\{t_i^1, t_i^2, \ldots, t_i^B\}$ at random from the entire time interval *T*. A burst is then modeled as a simple binary pulse of duration τ_b, so that $h_i(t) = 1$ for $\{t_i^1 \le t \le t_i^1 + \tau_b, t_i^2 \le t \le t_i^2 + \tau_b, \ldots, t_i^B \le t \le t_i^B + \tau_b\}$, and $h_i(t) = 0$ otherwise (Fig. 2*A*). We use values of $B = 1, 2, 4, 8$, and based on experimental observations of the HVC burst length (Hahnloser et al. 2002), use $\tau_b = 6$ ms. We assume a nonlinear form for the RA activation function, given by the sigmoid $f(x) = r_{\max}/(1 + e^{-2x/a})$, so $f'(x) = f(x)[r_{\max} - f(x)](2/sr_{\max})$, with $r_{\max} = 600$ Hz and $s = 5$ (*s* is a parameter that stretches the analog part of the response; large values of *s* produce analog neurons with a linear regime and saturation, whereas the $s \to 0$ limit produces binary neurons. In experimental current-injection studies, RA neurons show a range of linear response up to at least 100 Hz (Spiro et al. 1999), and routinely fire bursts of spikes at 500 Hz during song, motivating our choice of $s = 5$ and $r_{\max} = 600$ Hz. In all simulations, the total duration of the simulated song motif is $T = 150$ ms, and time is discretized with a grain of $dt = 0.1$ ms. The initial HVC-to-RA weights W_{ij} are picked randomly from the interval [0, 1/*B*] (scaling with *B* to keep the summed drive to RA fixed as the number of bursts per neuron per motif is varied in HVC), with 40% of them ($P_{dil} = 0.4$) randomly diluted to zero. The threshold for RA neurons is given by $\theta = 1.2(1 - P_{dil}) N_h \tau_b / T$, where $N_h \tau_b / T$ is the average input received by the average RA neuron from HVC at each time in the song; the factor 1.2 is chosen to keep RA activity low initially. Each RA neuron projects to one output neuron (i.e., the RA-to-output weight matrix *A* is

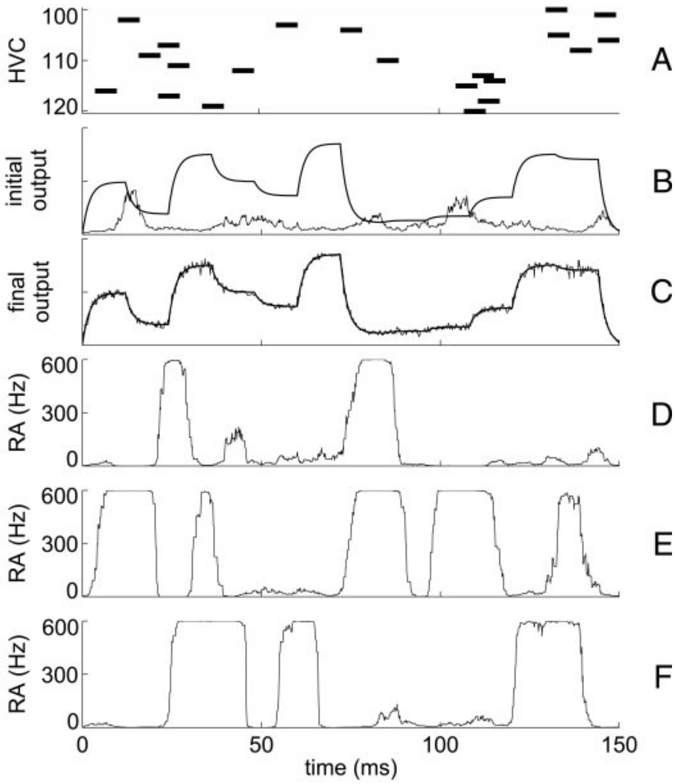

FIG. 2. *A:* activity of RA-projecting HVC neurons as a function of time, shown for 20 of the 500 neurons in the simulation. Black bars indicate that the neuron is bursting at that time, whereas otherwise the neuron is silent. *b:* desired (thick line) and actual (thin line) output activity for one of the 2 output units, before learning begins. *C:* desired (thick line) and actual (thin line) activity of the same output unit after learning; the second output behaves similarly. *D–F:* example of the activities of 3 RA units, after learning (see text for further discussion).

block-diagonal), and equal numbers of RA neurons project to each output. The nonzero entries of A are chosen from a Gaussian distribution with mean 1 and SD 1/4. Desired sequences $d_k(t)$ for the output units are fixed by choosing a sequence of steps of 12-ms duration and random heights chosen from the interval $[0, N_r/(8N_o)]$, and are smoothed with a 2-ms linear low-pass filter. The gradient-following rule, *Eq. 4,* is used to update the weights W after each epoch.

To study the effects of sparse HVC activity on learning speed, we performed 4 groups of simulations where B, the number of bursts per HVC neuron per song motif, was fixed at $B = 1, 2, 4$, or 8, respectively. For each B, we performed several sets of learning trials with a separate, systematically varied value of the overall learning step-size η for each set (more details below). Within each set of simulations, consisting of 15 trials each with fixed η, the weights A and W were drawn randomly and independently for every trial, as described above. All other parameters, including the desired outputs $d_k(t)$, were kept fixed for all B and all η. Initially 25 evenly spaced values of η were chosen for each B, always in a range where some of the values were too large and resulted in divergence of the learning curve, whereas most values resulted in decreasing errors. The (15-trial) averaged learning curves for each η were judged to be rapidly or slowly converging based on the number of epochs taken to cross a preselected, reasonably small error value (see below); only learning curves with nonincreasing error over the length of the simulation were considered. Typically, very small values of η result in very slow learning, whereas very large values lead to divergence. Thus, the best learning speeds could be obtained by a choice of η away from both extremes. To make sure the learning curves chosen for comparison as a function of B were reasonably close to the best possible curve for each B, we picked 2 values of η for each B that resulted in the 2 fastest averaged learning curves, and used these as endpoints in another set of learning trials with 10 values of η spaced between the endpoints. For each η, we again averaged 15 trials. By this process, a value of $\eta = \eta^*(B)$ was found that resulted in the fastest learning for each B.

The threshold error value at which we consider the network to have learned the task is when it reached an error of 0.02 or better [corresponding to $\int dt\, \Sigma_k (d_k - o_k)^2 < 1\% \int dt\, \Sigma_k d_k^2$, thin horizontal line in Fig. 3; for an example of the output performance in what we consider to be a well-learned task, see Fig. 2*c* where $\int dt\, \Sigma_k (d_k - o_k)^2 = 0.15\% \int dt\, \Sigma_k d_k^2$]; learning speeds are judged by the number of epochs taken for the learning curves to reach this value.

Parameter variations and ranges

The network converged to produce outputs close to the desired outputs over a large range of parameters, so long as a sufficiently small value of the learning rate parameter, η, was used. This is expected, because with small η, the learning rule follows the gradient of the error function, and will converge to a local minimum of the error surface; more interestingly, the dependency of learning time on B (see RESULTS) was also consistent across a large parameter range.

In simulation, we tried variations where W was drawn from a Gaussian, instead of uniform, random distribution; the initial weight dilution, P_{dil}, ranged from 0 to 0.6 (0–60% of the initial weights initially diluted to 0); half of all nonzero weights from RA to each output unit (in A) were made negative, mimicking push–pull rather than just pull control over the outputs; the numbers of HVC, RA, and output units were independently varied by factors of 0.5 and 2; the simulated song length ranged from 80 to 400 ms; RA unit activation functions were taken to be linear or sigmoidal. In all of these cases, it was possible to find η so that the simulations converged to the desired output, and the dependency of learning time on B was found to be qualitatively the same as for the specific parameters described here.

The results shown here are with parameters chosen according to the following priorities. *1*) Simulate the largest network that would run in a reasonable amount of time. We used $N_h = 500$, $N_r = 800$, and $N_o = 2$, in place of $N_h \approx 20{,}000$, $N_r \approx 7{,}000$, and $N_o \approx 7$ in the actual bird, where N_o is taken to be the number of individual vocal muscles controlled by RA. The simulated song length T had to be scaled down

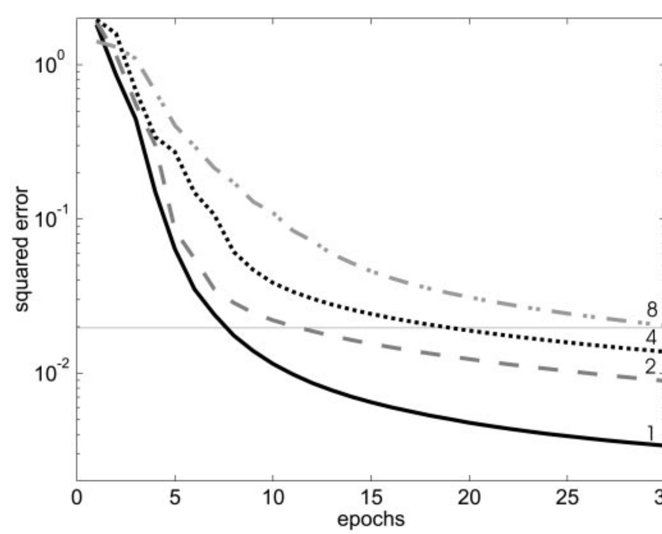

FIG. 3. Four curves track error as a function of epoch while learning with $B = 1, 2, 4$, and 8 bursts per HVC neuron per simulated song segment. For each B, the overall weight update step size was optimized to give the fastest possible monotonic convergence toward zero error. Number of epochs taken to reach a prespecified learning criterion (thin horizontal line) grows sharply with B, nearly doubling each time B doubles.

to compensate for the reduced HVC and RA model populations driving song; thus $T = 150$ ms instead of the approximately 600- to 1,000-ms duration of a typical zebra finch song motif. *2*) Initiate (before learning) the HVC–RA weights and RA neural thresholds so that the initial activity in RA is low and nonuniform. This was done because we noticed that, interestingly, if initiated in this way, the postlearning activity in the model RA neurons reliably resembles that of RA neurons in the actual songbird (see RESULTS).

Numerical eigenvalue computation

For each of $B = 1, 2, 4$, or 8, we randomly generated a matrix of HVC activity (as described above) with $N_h = 3000$, $T = 300$ ms, $\tau_b = 6$ ms, and $dt = 0.1$ ms. For each B, the HVC equal-time cross-correlation matrix $Q_{ij} = \Sigma_{t=0}^{T} h_i(t)h_j(t)$ was computed, and its eigenvalues computed numerically.

RESULTS

Simulations

We simulated learning by gradient following (as described in *Eqs. 1–4* and METHODS) in a feedforward network consisting of an HVC, an RA, and a motor output layer (Fig. 1). Sample input (HVC activity) and the initial and desired outputs (for one of 2 output units) are shown in Fig. 2, *A* and *B*, respectively. In the simulation of Fig. 2, each HVC neuron is active exactly once in the song motif. After several epochs of learning (gradient descent on the mismatch between actual and desired outputs), activity in the output units closely matches the desired outputs; Fig. 2*C*. Note that in our model, the RA neurons act as hidden units and their patterns of activity are not explicitly constrained. The activities of 3 randomly selected RA neurons from the model network after learning is complete are shown in Fig. 2, *D–F*. It is interesting to note that with sigmoid RA activation functions, if initial connections between HVC and RA are weak and random and if initial RA activity is low, the emergent activity patterns of RA neurons in the trained network qualitatively resemble the behavior of real RA neurons recorded in vivo during singing (Yu and Margoliash 1986; A. Leonardo and M. S. Fee, unpublished observations): for example, individual RA unit activity is not well correlated with the outputs, the distribution of single-burst durations of RA neurons resembles that of RA neurons in the singing zebra finch, and similar patterns of output activity may be driven by rather different patterns of activity in RA.

Our goal is to examine the effects of the sparseness of HVC drive on the learning speed of the network. We repeated the learning simulations, as pictured in Fig. 2, with fixed values for the song length, single-burst duration of HVC neurons, and network size, but varied B, the number of bursts fired per HVC neuron per song motif (see METHODS). Figure 3 shows the results of this study; the 4 learning curves correspond to simulations where the number of bursts per HVC neuron is varied to be $B = 1, 2, 4$, or 8, respectively. Each curve in Fig. 3 is an average over 15 trials that start with different random initial weights W and A but with a single fixed B. The network is considered to have learned the task when the error drops below a prespecified error tolerance, signified by the thin horizontal line. For each value of B, the task of learning was realizable (i.e., the network could successfully learn the desired outputs). Also for each B, the overall coefficient controlling the weight-update step size was optimized to give the fastest learning possible (see METHODS); thus the learning speed comparision is between the best-case multitrial average curves for each B.

In going from $B = 1$ burst per HVC neuron per motif to 2 bursts, we see in Fig. 3 that the learning time (number of iterations for the learning curve to intersect the learning criterion line) nearly *doubles;* the same happens in going from 2 bursts to 4, or 4 to 8. This apparently strong dependence of learning time on the number of HVC bursts is surprising, considering that in all cases (all B) the learning task was realizable, and that the premotor HVC drive in going from $B = 1$ to $B = 4$, for example, was still relatively sparse. The effect, that increasing B leads to increased learning time, persisted over a wide range of network parameters (see note on parameter choices in METHODS). To better understand the process by which more densely distributed HVC bursts per motif lead to slower learning, and why this effect is robust across a broad range of parameters, we turn to an analysis of learning in a linearized version of the network.

Linear analysis

We found the basic effect of the slowdown of learning with temporally denser HVC codes to be present regardless of many changes in network properties, such as network size, length of simulated motif, and choice of RA activation function. To isolate the critical factors involved in the learning slowdown, we study the learning curves of a network with the same architecture and learning rule as above, but with linear RA activation functions, $f(y) = y$. Although this is a simplification, a linear network permits us to analytically derive the dependence of the learning curves on B, the number of times each HVC neuron bursts during a song motif. Moreover, linear analysis lends itself to a convenient geometric interpretation of the learning process.

RELATION BETWEEN LEARNING SPEED AND HVC ACTIVITY. If RA units are linear, the error function C of *Eq. 3* becomes a quadratic surface over the multidimensional space $\{W\}$ of the HVC-to-RA weights (see APPENDIX)

$$C = \mathrm{Tr}\,\{AWQW^TA^T\} \tag{5}$$

In geometric terms, Q is a matrix that determines the shape of the quadratic surface, because its eigenvalues specify the overall shape (steepness or flatness) of the quadratic surface along the various directions in weight space. Large eigenvalues correspond to steep directions, and small eigenvalues to shallow ones. In terms of network activities, Q is the zero-time-lag correlation matrix of HVC activity: element Q_{ij} reflects the equal-time cross-correlations in the activity of neurons i and j, summed over all times in the motif. For example, if the 2 neurons are always coactive, Q_{ij} is large, and if they are never coactive, $Q_{ij} = 0$. The importance of Q in shaping the error surface emerges from the fact that HVC activity determines which synapses W are active in driving the output, how often they are used, and thus whether and when they must be modified to reduce error.

Learning corresponds to moving downward on the paraboloid quadratic surface by adjustment of the underlying network weights W. Learning by gradient descent, *Eq. 4,* means that the downward movement follows along the direction of the gradient or path steepest descent toward the minimum of the error surface. The total error may be broken down into components of error along different directions in the weight space $\{W\}$, and it is well known that in a linear system, these component errors decrease as decaying exponentials with different decay rates; these decay rates are determined by the shape of the error surface in that direction. Specifically, with certain assumptions on the distribution of fixed weights A, the optimal (leading to fastest learning) decay rates are given by ratios of the eigenvalues of Q $\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$, with the largest eigenvalue, λ_1 (see APPENDIX). The learning speed along the direction parallel to eigenvector α can be defined as the decay rate along that direction

$$\nu_\alpha = \lambda_\alpha/\lambda_1 \tag{6}$$

For learning to converge, all ν_α values must be less than 1 and greater than 0; this is necessarily true here because all eigenvalues of Q are guaranteed to be positive, and the factor of $1/\lambda_1$ effectively sets the maximum learning speed to be less than 1. Within these limits, the larger all the ν_α, the larger the decay rate, and so the total error will decrease more rapidly. It is instructive to consider two cases: *1*) all eigenvalues are essentially equal, and *2*) all eigenvalues are equal but one, which is very much larger. In *case 1*, we see from *Eq. 6* that the learning speeds along all α are equal and equal to 1; the geometric interpretation is that the error surface is isotropic, Fig. 4*A*, and learning can proceed (equally) rapidly in all directions of the error surface. In *case 2*, the error surface is strongly anisotropic Fig. 4*B*. Learning will still be fast along the (steep) direction corresponding to λ_1, given that $\nu_1 = 1$. However, learning along all other directions will be much slower because all remaining $\nu_\alpha \ll 1$. Geometrically speaking, the maximum weight-update step size is constrained by the steepest direction, since small steps in weight space lead to large changes in error and can quickly lead to divergent error. Because the remaining directions are much shallower, the small weight-space step size constraint leads to much smaller decreases in error per epoch along all other directions, resulting in a sharp slowdown in the overall learning.

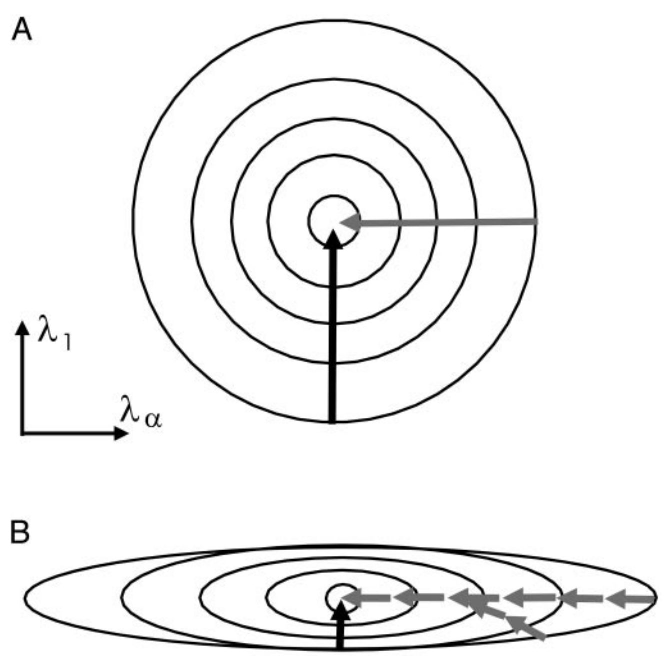

FIG. 4. Ellipses are contours of equal error, and a varying density of these contours corresponds to varying steepness on the error surface (high density = steeper). *A*: starting from a given error, the maximum allowable step size in weight space is the same, regardless of the direction from which the minimum is approached. *B*: on an anisotropic surface, the steepest direction (corresponding to the eigenvector with largest eigenvalue, and designated here by λ_1) dictates the maximum allowable step size in weight space, and constrains learning in all other directions (λ_α) as well.

Hence, a narrowly distributed range of eigenvalues leads to faster learning, whereas singularly large eigenvalues that stand out from the rest broaden the range and cause a slowdown.

MEAN-FIELD DERIVATION: LEARNING TIME GROWTH WITH SYNAPTIC INTERFERENCE. With *Eq. 6*, the problem of deriving learning curves is essentially reduced to the problem of computing the eigenvalues of the correlation matrix Q. Certain important features of the eigenvalue distribution can be derived from a mean-field matrix $\langle Q\rangle$, obtained by replacing each element of the correlation matrix with its ensemble-averaged expectation value (see APPENDIX); moreover, $\langle Q\rangle$ elucidates the relationship between features of HVC activity and features of the eigenvalue spectrum. As B is increased, the HVC autocorrelations (diagonal elements of $\langle Q\rangle$) increase as B, whereas the cross-correlations (off-diagonal) increase as a small factor times B^2. The cross-correlations contribute to only the largest eigenvalue of $\langle Q\rangle$, causing it to scale as B^2, whereas all remaining eigenvalues scale as B. Therefore the largest eigenvalue of $\langle Q\rangle$ is a direct reflection of cross-correlations in HVC activity. Because cross-correlations in HVC activity lead to interference or cross-correlations in the use of different HVC–RA synapses in driving the song motif, the size of the largest eigenvalue equivalently reflects the degree of synaptic interference in the HVC–RA synapses. Let $\nu_\alpha(B)$ designate the learning speed along the αth eigenvector of Q as a function of B. The mean-field eigenvalue calculation yields (see APPENDIX)

$$\nu_1(B) = \nu_1(1) \quad \text{in steepest direction } (\alpha = 1) \tag{7}$$

$$\nu_\alpha(B) = \frac{\nu_\alpha(1)}{B} \quad \text{all other directions } (\alpha > 1) \tag{8}$$

In other words, as B is increased, the optimal learning speeds decreases as $1/B$ along *all* directions in weight space except along the direction corresponding to λ_1, whose optimal learning speed remains unchanged. Because the cumulative initial error will generically have significant error components in several directions, the cumulative learning speeds will noticably decrease as B is increased. According to the mean-field results above, the learning time with $B = 2$ will be approximately *twice as long* as for $B = 1$ because of synaptic interference. It is important to note, also, that the effects of increasing B on learning speed should be noticeable soon after learning has begun and the first transients (corresponding to learning along the first eigenvector) have passed, and not just toward the end of learning, where only the fine features remain to be learned. That is, the effects of multiple bursts on learning speed are manifest whether the output is learned relatively crudely or to great final precision.

This is all in good agreement with the overall decrease in learning speeds observed in the nonlinear network simulations of the last section. In the linear analysis, we see moreover that the scaling of learning time with B is an essential one (see APPENDIX): given a fixed network size, motif length, and HVC single-burst duration, increasing the number of bursts per HVC neuron per motif necessarily leads to a reduction in the optimal learning speed for the network, with no adjustable parameters to remove this dependency. In other words: learning with multiple bursts per HVC neuron per motif will be slower than learning with fewer bursts, independent of the HVC–RA network size, the motif length, and the single-burst duration, so long as these parameters are kept fixed while the number of bursts is varied in the comparison of learning time.

The mean-field analysis also sheds light on the identity of the eigenvector with the largest eigenvalue λ_1: it is the *common mode* eigenvector, with all positive entries, that corresponds to a simultaneous increase or decrease, for all parts of the motif, in the summed drive from HVC to the motor outputs. It is intuitive that this is the most "volatile" mode, leading to explosive growth of network activity. The remaining modes are *differential,* allowing rearrangements of the motor drive from moment to moment in the song without a large net change in the mean strength of the drive.

NUMERICAL VERIFICATION OF MEAN-FIELD CALCULATION. The vastly simplified mean-field derivation of the scaling of learning speed with B (from the eigenvalues of $\langle Q \rangle$) neglected variance and other higher-order statistics of Q. To check the results of the analysis, therefore, we numerically compute the eigenvalues of Q from randomly generated HVC activity matrices (see METHODS). The results are shown in Fig. 5, and agree well with the mean-field analysis. In Fig. 5*a*, we plot the top 300 $B = 1$ eigenvalues, together with the top 300 $B = 8$ eigenvalues scaled by 1/8. All the eigenvalues for $B = 1$ form a continuum, and the scaled $B = 8$ eigenvalues sit on the same continuum, except for the top eigenvalue, which is much larger than the rest. The gap between the topmost eigenvalue and all the rest for $B > 1$ is better seen in the *inset* of Fig. 5*a*, where the largest eigenvalue scales as B^2, whereas the second-largest scales as B. This causes learning speeds to scale as $1/B$ (Fig. 6), as derived in *Eq. 8.*

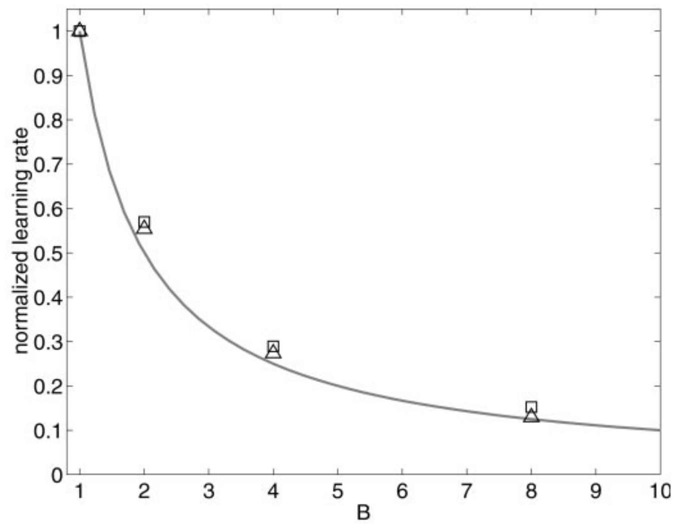

FIG. 6. Scaling of learning speed as a function of B, normalized by the learning speed for the case $B = 1$ [$\nu_\alpha(B)/\nu_\alpha(1)$ vs. B], plotted for learning along 2 directions (eigenvectors) in the space of weights: modes $\alpha = 2$ (△) and $\alpha = 200$ (□). These points are obtained from the numerical calculation of the eigenvalues of Q. Solid line: predicted scaling of learning speed with B, for all $\alpha > 1$, from the mean-field correlation matrix $\langle Q \rangle$.

The numerical computation shows that there is a spread in the eigenvalue continuum even when $B = 1$: because of the small but nonnegligible HVC single-burst duration, and continuous spread of burst-onset times, the activities of different HVC neurons have partial overlaps with each other. This already leads to slower learning than if bursts were completely nonoverlapping. However, as we have seen in the preceding analysis, increasing the number of bursts per HVC neuron leads to larger correlations in HVC activity and a considerably greater spread of eigenvalues, and thus to slower descent on the error surface.

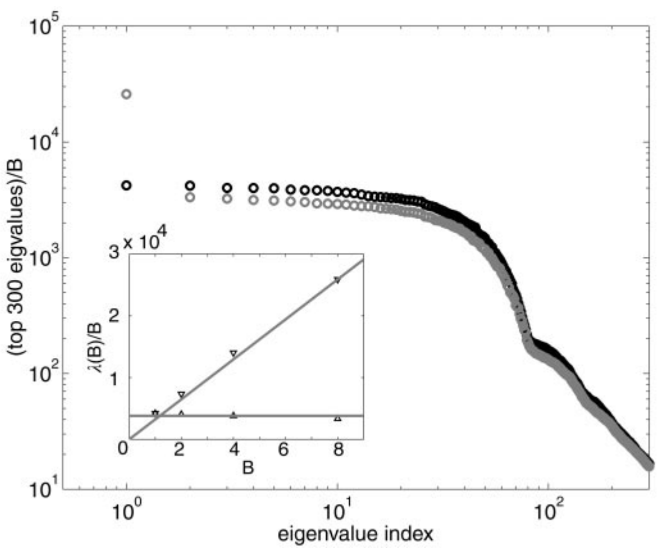

FIG. 5. Top 300 eigenvalues of the correlation matrix Q, divided by B, for $B = 1$ bursts per HVC neuron per song segment (black circles), and for $B = 8$ (gray circles). *Inset*: scaling of λ_1 (▽) and λ_2 (△) with B, from numerical calculations. We see that $\lambda_1/B \sim B$, whereas $\lambda_2/B \sim$ *const.* Solid lines show the same scaling, derived from $\langle Q \rangle$.

DISCUSSION

Summary

We have built a simplified framework to analyze the learning of premotor representations in the songbird premotor circuit, given a sparse premotor drive from HVC, a set of plastic connections between HVC and RA, and a gradient learning rule that minimizes the mismatch between the tutor and pupil songs. Within this framework, we have demonstrated how temporally sparse activity allows the fast learning of premotor representations, and have quantified, in a network of linear neurons, the dependency of learning rate on the number of times an HVC neuron is active during a motif. Sparsely active HVC neurons have small cross-correlations: increasing the number of HVC bursts per motif increases cross-correlations in HVC activity, which leads to correlated changes of synaptic weights. To keep network activity from diverging because of the correlated weight changes, the maximum allowable weight-update size must be constrained; this normalization decreases the step size for other, uncorrelated weight changes that are required for learning. Thus the overall learning speed decreases with increasing numbers of HVC bursts per motif. Although our analytical description is based on linear units, the simulations (Fig. 3) of learning in nonlinear units and the heuristic explanation of increased synaptic interference with increased numbers of bursts point to a broader relevance of this analysis to networks with more realistic neurons.

Relation to past work

Several motor and sensory brain areas display sparse neural codes. This work augments other theoretical studies that argue in favor of the utility of sparse codes in various contexts, such as information theory, coding fidelity, decoding ease, and learning efficiency (Foldiak 1995). We have presented a quantitative analysis of the relationship between sparse representations in layers coding high-level activity (in this case, abstract sequential activity in HVC) and learning speed in a multilayer feedforward network.

Questions about training time in networks such as this one have been studied in the machine learning community, resulting in prescriptions to speed up learning by rescaling the learning rate parameter (overall weight update step size) differently along the different eigenvectors, or by reparameterizing neuronal activities to make the error surface more isotropic. In a work closely related to this one, LeCun et al. (1991) in particular recommend that the eigenvector associated with the largest eigenvalue be subtracted from the learning updates, or that symmetrically active $\{-1, 1\}$ neuronal units be used in the input layer instead of asymmetric $\{0, 1\}$ units, thus reducing the anisotropy of the error surface by reducing the mean of the off-diagonal entries of the input-unit correlation matrix and so bringing the largest eigenvalue closer to the remaining ones. Given that neural firing rates are zero or positive, the activity of individual neurons in biological networks is necessarily asymmetric. Furthermore, although the learning rate parameter (overall step size) may easily be tuned at the individual synaptic level, it is, not obvious how to apply separate learning rates to separate eigenvectors in a biologically plausible way, since individual synapses participate in multiple eigenvectors. Therefore, we suggest that with the use of unary HVC activity in birdsong learning, biology may have found its own solution to these very problems.

Different learning rules

We have also performed simulations of learning by stochastic weight perturbation, a reinforcement algorithm that drives learning by making stochastic estimates of the gradient without explicitly computing it; we obtain preliminary results from simulation that are qualitatively similar to the ones stated for direct gradient learning in this paper, finding that learning is faster when the number of HVC bursts is small. In fact, if biology does indeed make use of stochastic reinforcement algorithms to perform goal-related learning, the impetus to increase learning speed through sparse coding may be considerably greater because such stochastic gradient algorithms are typically much slower overall than algorithms that can directly compute gradients and move along them.

Correlations in HVC activity

In this work, each HVC model neuron can equivalently be viewed as a subpopulation of perfectly correlated (i.e., always coactive) neurons. We studied the case where each strongly self-correlated subpopulation bursts one or multiple times, but where the individual subpopulations are independent of each other. This picture is fully consistent both with the HVC data on RA-projecting neurons (Hahnloser et al. 2002), and with recurrent synfire chainlike models for the generation of sequential activity in populations of neurons.

Nevertheless, it *is* possible to imagine a case where the subpopulations are correlated with each other: if for example, the simultaneous bursting of 2 subpopulations in one part of the motif makes it more likely that, when they each burst again in other parts of the motif, they will burst together. Such correlations between neural subpopulations would enhance the correlations in the overall population activity at different times in the song, increasing synaptic interference compared to the independent subpopulation case, and increasing the overall anisotropy of the error surface. In this case, our qualitative results on the advantage of sparse coding for learning would be the same; in detail, the slowdown resulting from nonsparse coding would be more pronounced, from the additional contribution of intergroup cross-correlations, than described for the independent subpopulation case.

Juvenile HVC activity

Single-unit HVC recordings have been made only in adult birds, where the coding is seen to be unary (single burst of activity per neuron per motif). We wondered what the role of such extreme sparseness in HVC might be if it were present during the learning process, and found that it could confer a great advantage in terms of learning speed. On this basis, we predict that if songbirds acquire their songs under pressure to learn quickly, then sparseness of HVC activity could be integral to the learning process and should thus already be present in the HVC of juvenile birds in the early and mid sensorimotor period, instead of arising as an emergent property late in song learning.

Relevance to other sensory and motor systems

The aspect of motor learning we have explored here is the mapping of a set of sparse, high-level neural (HVC) patterns onto a denser set of low-level motor activations, in a multilayer feedforward network model of song generation. Because biological sensory and motor processing areas tend to be multilayered with important feedforward components, these results relating sparseness to learning speed in the formation of feedforward maps should be relevant in a broad range of systems. Examples of sparse coding can be seen in rat auditory cortex neurons responding to tone pips (Deweese et al. 2003); temporally and spatially sparse responses to natural scenes in ferret visual cortex (Weliky et al. 2003); sparse representations of location in hippocampal place cells; highly selective corticostriatal activity in macaque motor cortex (Turner and Delong 2000); and sparse coding of odor identity in Kenyon cells of the locust mushroom body (Perez-Orive et al. 2002; Theunissen 2003). The results of our study suggest that such sparse sensory and motor codes may facilitate the learning of feedforward representations.

On the other hand, one might wonder why, if sparse coding confers a significant advantage in terms of learning speed, are not most neural representations ultrasparse or unary? One reason is that sparse coding carries a cost: the representational capacity of a very sparsely coded network is low. Thus, the advantages of sparse coding must be balanced against capacity constraints. Such capacity constraints may dominate, or at least play a more important role, in systems other than the zebra finch HVC. For example, songbirds that memorize much larger song repertoires may be subject to HVC volume constraints, and in these cases, we expect the coding to be sparse, for fast learning, but not necessarily unary, as in the finch.

Other implications of sparse coding

We do not intend to imply that the only role of sparse coding in the zebra finch HVC is the reduction of synaptic interference in the learning of feedforward HVC-to-RA weights. Temporally sparse coding could play an important role in mitigating the problem of temporal credit assignment in learning, which is encountered when feedback about a performance arrives significantly later than the neural activities that generated it. Moreover, sparse codes in HVC may play an important role not just in motor aspects of song learning and production but in song recognition as well (Lewicki and Konishi 1995; Margoliash 1986; Margoliash and Fortune 1992; Volman 1993).

APPENDIX

Learning curve

With linear RA neurons, we define the network equations to be $r = W\,h$, $o = AW\,h \equiv Xh$, where h is the $N_h \times N_s$ matrix of HVC activity, r is the $N_r \times N_s$ matrix of RA activity, o is a $N_o \times N_s$ matrix of output activity, A is the matrix of fixed weights from RA to the outputs, and W is the matrix of plastic weights from HVC to RA. $N_s = T/\mathrm{d}t$ is the number of discrete time bins in the motif, where T is the motif length and dt is the grain size. With a change of variables $Q \equiv hh^T$ and $x = X - X^*$, where X^* is defined by $X^*h = d$ (X^* exists if a solution exists, i.e., if the learning task is realizable), the cost function is $C = \frac{1}{2}\mathrm{Tr}\{xQx^T\}$. Applying a gradient descent update, *Eq. 4*, on C, we have that $x \rightarrow (x - \eta AA^T\,xQ)$, where η is a positive scalar that scales the overall learning step size. If each RA neuron projects to one output unit, and if the summed synaptic weights to each output are approximately equal, AA^T becomes a scalar matrix that can be absorbed into η. Thus, the multilayer perceptron problem with 2 layers of weights becomes effectively a single-layer perceptron, and we have that after n iterations $x^{(n)} = x^{(0)}\,(1 - \eta Q)^n$, so

$$C^{(n)} = \frac{1}{2}\,\mathrm{Tr}\,\{x^{(0)}(1 - \eta Q)^n Q(1 - \eta Q^T)^n x^{(0)T}\} \tag{A1}$$

In the eigenvector basis of the Hessian matrix Q (eigenvalues $\{\lambda_{\alpha=1,\ldots,N_h}\}$ arranged in nonincreasing order, $\lambda_1 \geq \ldots \geq \lambda_{N_h}$), with projection of the kth row of $x^{(0)}$ along the αth eigenvector given by $\chi_{k\alpha}$, the error after n learning iterations is given by

$$C^{(n)} = \frac{1}{2}\sum_{\alpha,k}(1 - \eta\lambda_\alpha)^{2n}\lambda_\alpha|\chi_{k\alpha}|^2 \tag{A2}$$

Let $c_\alpha \equiv \lambda_\alpha\,\Sigma_k\,|\chi_{k\alpha}|^2$ represent the initial error along the αth eigenvector. The total error evolves iteratively by multiplication of the initial errors c_α by a factor $(1 - \eta\lambda_\alpha)^2$ per iteration; η must be chosen small enough so that $|1 - \eta\lambda_\alpha| < 1$ for each α, to allow error to decrease and for the learning curve of *Eq. A2* to converge. Hence, η must be less than $2/\lambda_1$, and it is easy to see that the optimal choice for η is $\eta^* = 1/\lambda_1$ ($\eta < 1/\lambda_1$ leads to overdamped convergence, whereas $1/\lambda_1 < \eta < 2/\lambda_1$ displays underdamped oscillatory convergence).

Analysis of eigenvalues

The mean-field matrix $\langle Q\rangle$ is formed by replacing all elements of Q by their ensemble-averaged expectation values (i.e., generate Q and average, element by element, over several trials). Therefore, $\langle Q\rangle = BN_bI + (B^2\,N_b^2/N_a)\mathbf{11}^T$, where $N_b \equiv \tau_b/\mathrm{d}t$. There are only 2 distinct eigenvalues, $\lambda_1 = BN_b + B^2\,N_b^{\,2}\,[(N_h - 1)/N_a] \approx B^2\,N_h\,N_b\,(\tau_b/T)$ (provided $N_h\,\tau_b/T \gg 1$) corresponding to the common mode eigenvector, and $\lambda_2 = BN_b - B^2N_b^{\,2}/N_a \approx BN_b$ (provided $T \gg B_{\tau_b}$ *corresponding to the* $N_h - 1$ differential modes. [This effect, of an eigenvalue spectrum with one 'large' eigenvalue, is generic for N $\times$ N matrices with random entries of mean a on the diagonal and b on the off-diagonal, if $b \gg a/N$ (Edwards and Jones 1976).] Hence the learning rate for all modes $\alpha > 1$ is given by $\nu_\alpha = (1/B)\,(T/N_h\tau_b) \sim 1/B$, as in *Eq. 8* (Fig. 6).

ACKNOWLEDGMENTS

We are grateful to R. Tedrake for help in setting up a system to run parallel simulations, to M. Mehta and J. Werfel for comments on the manuscript, to X. Xie and M. Tresch for helpful discussions, and to the referees for suggestions.

GRANTS

This work was supported by grants from National Institutes of Health, Howard Hughes Medical Institute, and National Science Foundation (PHY 99-07949).

REFERENCES

Bartlett P and Baxter J. Hebbian synaptic modifications in spiking neurons that learn. Australian National University, Canberra, 0200 Australia, 1999. *Technical report.*

Bottjer S, Miesner E, and Arnold A. Forebrain lesions disrupt development but not maintenance of song in passerine birds. *Science* 224: 901–903, 1984.

Brainard M and Doupe A. Auditory feedback in learning and maintenance of vocal behavior. *Nat Rev Neurosci* 1: 31–40, 2000.

Deweese M, Wehr M, and Zador A. Binary spiking in auditory cortex. *J Neurosci* 23: 7940–7949, 2003.

Doya K and Sejnowski T. A novel reinforcement model of birdsong vocalization learning. In: *Advances in Neural Information Processing Systems 7*, edited by Tesauro G, Touretzky D, and Leen T. Cambridge, MA: MIT Press, 1995, p. 101–108.

Edwards S and Jones R. The eigenvalue spectrum of a large symmetric random matrix. *J Phys A* 9: 1595, 1976.

Foldiak P. Sparse coding in the primate cortex. In: *The Handbook of Brain Theory and Neural Networks*, edited by Arbib M. Cambridge, MA: MIT Press, 1995, p. 895–898.

Hahnloser R, Kozhevnikov A, and Fee M. An ultra-sparse code underlies the generation of neural sequences in a songbird. *Nature* 419: 65–70, 2002.

Hermann M, Hertz J, and Prugel-Bennett A. Analysis of synfire chains. *Network* 6: 403–414, 1995.

Herrmann K and Arnold A. The development of afferent projections to the robust archistriatal nucleus in male zebra finches: a quantitative electron microscopic study. *J Neurosci* 11: 2063–2074, 1991.

Konishi M. The role of auditory feedback in the control of vocalization in the white-crowned sparrow. *Z Tierpsychol* 22: 770–783, 1965.

Le Cun Y, Kanter I, and Solla S. Eigenvalues of covariance matrices: application to neural learning. *Phys Rev Lett* 66: 2396–2399, 1991.

Lewicki M and Konishi M. Mechanisms underlying the sensitivity of songbird forebrain neurons to temporal order. *Proc Natl Acad Sci USA* 92: 5582–5586, 1995.

Margoliash D. Preference for autogenous song by auditory neurons in a song system nucleus of the white-crowned sparrow. *J Neurosci* 6: 1643–1661, 1986.

Margoliash D and Fortune E. Temporal and harmonic combination-sensitive neurons in the zebra finch's HVC. *J Neurosci* 12: 4309–4326, 1992.

Meunier C, Yanai H, and Amari S. Sparsely coded associative memories: capacity and dynamical properties. *Netw Comput Neural Syst* 2: 469–487, 1991.

Mooney R. Synaptic basis for developmental plasticity in a birdsong nucleus. *J Neurosci* 12: 2464–2477, 1992.

Nottebohm F, Kelley D, and Paton J. Connections of vocal control nuclei in the canary telencephalon. *J Comp Neurol* 207: 344–357, 1982.

Nottebohm F, Stokes T, and Leonard C. Central control of song in the canary, *Serinus canarius. J Comp Neurol* 165: 457–486, 1976.

Perez-Orive J, Mazor O, Turner G, Cassenaer S, Wilson R, and Laurent G. Oscillations and sparsening of odor representations in the mushroom body. *Science* 297: 359–365, 2002.

Sakaguchi H and Saito ND. Developmental changes in axon terminals visualized by immunofluorescence for the growth-associated protein, gap-43, in the robust nucleus of the archistriatum of the zebra finch. *Brain Res Dev Brain Res* 95: 245–251, 1996.

Seung H. Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. *Neuron* 40: 1063–1073, 2003.

Simpson H and Vicario D. Brain pathways for learned and unlearned vocalizations differ in zebra finches. *J Neurosci* 10: 1541–1556, 1990.

Spiro JE, Dalva MB, and Mooney R. Long-range inhibition within the zebra finch song nucleus RA. *J Neurophysiol* 81: 3007–3020, 1999.

Stark L and Perkel D. Two-stage, input-specific synaptic maturation in a nucleus essential for vocal production in the zebra finch. *J Neurosci* 19: 9107–9116, 1999.

Tchernichovski O, Mitra P, Lints T, and Nottebohm F. Dynamics of the vocal imitation process: how a zebra finch learns its song. *Science* 291: 2564–2569, 2001.

Theunissen F. From synchrony to sparseness. *Trends Neurosci* 26: 61–64, 2003.

Troyer T and Doupe A. An associational model of birdsong sensorimotor learning. I. Efference copy and the learning of song syllables. *J Neurophysiol* 84: 1204–1223, 2000.

Tsodyks M and Feigelman M. Enhanced storage capacity in neural networks with low level of activity. *Europhys Lett* 6: 101, 1988.

Turner R and Delong M. Corticostriatal activity in primary motor cortex of the macaque. *J Neurosci* 20: 7096–7108, 2000.

Volman S. Development of neural selectivity for birdsong during vocal learning. *J Neurosci* 13: 4737–4747, 1993.

Weliky M, Fiser J, Hunt R, and Wagner D. Coding of natural scenes in primary visual cortex. *Neuron* 37: 703–718, 2003.

Williams R. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* 8: 229–256, 1992.

Willshaw D, Buneman O, and Longuet-Higgins H. Non-holographic associative memory. *Nature* 222: 960–962, 1969.

Yu A and Margoliash D. Temporal hierarchical control of singing in birds. *Science* 273: 1871–1875, 1986.