Special Issue on the Role and Control of Random Events in Biological Systems

LEARNING SYNFIRE CHAINS: TURNING NOISE INTO SIGNAL

JOHN HERTZ* and ADAM PRÜGEL-BENNETT†
Nordita, Blegdamsvej 17, DK-2100 Copenhagen Ø, Denmark

We develop a model of cortical coding of stimuli by the sequences of activation patterns that they ignite in an initially random network. Hebbian learning then stabilizes these sequences, making them attractors of the dynamics. There is a competition between the capacity of the network and the stability of the sequences; for small stability parameter ϵ (the strength of the mean stabilizing PSP in the neurons in a learned sequence) the capacity is proportional to $1/\epsilon^2$. For ϵ of the order of or less than the PSPs of the untrained network, the capacity exceeds that for sequences learned from *tabula rasa*.

1. Introduction

This contribution deals with the problem of the neural code: how is perceptual and cognitive activity coded in the activity of neurons? There are many aspects to this problem, but the level we wish to focus on can be illustrated by an analogy with a computer.[1] What would we regard as the really fundamental answer to the question, "How does a computer work?" If we take the cover off and find where the disk drive cables are connected, etc., we have answered the question at one level. Another relevant level is that of the software, studied independent of the machine itself. It is important to understand the system at both of these levels, especially if does not work and we want to fix it. However, in the view we take here, the really basic answer would be: There is a CPU chip which does boolean computations using transistors in integrated circuits. In these computations, all information is represented as bit strings which are acted on by the instruction set of the processor.

We want to pose and answer the corresponding question about the brain. Unfortunately, we do not even know how to ask the question correctly, though we are beginning to be able to formulate the problem. We believe that cortical neurons communicate with each other by spike trains, but we do not know the code. It has long been a working assumption among neurophysiologists that the precise timing of the spikes does not carry information; the relevant quantity is simply the total number of spikes received by a neuron within a period of the order of its membrane time constant (about 10–20 ms). A careful investigation in visual cortical neurons of macaque monkeys lends support to this hypothesis, at least insofar as information about visual stimuli is concerned.[2,3] However, a growing body of evidence[4–6] suggests that in areas of frontal cortex involved in cognitive processing, the timing of spikes at the millisecond level may be important. The finding of these studies is that the same spike sequences occur frequently in repeated trials for a given behavioral state, and that just which sequences occur depends on this state. While spike timing is only one feature of this problem, it is a fundamental one, and we cannot deal with other aspects of neural coding until it is resolved.

What sort of computational principles lie behind the occurrence of precisely-timed spike sequences? Abeles and his co-workers have suggested that they reflect a pattern of spatiotemporal activity in which specific pools of neurons fire in rapid succession. They argue that if there are strong excitatory synapses between successive pools, the dynamics of integrate-and-fire neurons acts to stabilize

*E-mail: hertz@nordita.dk
†E-mail: adam@nordita.dk

firing patterns in which the neurons within a single pool become highly synchronized, thereby stabilizing the overall timing of the spatiotemporal activity pattern, which they call a "synfire chain". Their simulations of small systems support this picture, and in previous work[7] we carried out a simple model analysis of the stability of synfire chain propagation against both extrinsic noise and synaptic crosstalk with other chains or other links in the same chain. A similar theoretical analysis was carried out by Bienenstock.[8]

If synfire chains are found in real brains, the strong synaptic connections between successive pools must develop through some form of unsupervised learning. In this work we develop a simple model for how this can happen, based on a Hebbian algorithm. The scenario starts with a network of random asymmetric connections, which give rise to chaotic spontaneous activity. An external stimulus activates a particular pool of neurons at $t = 0$, setting off a particular activity trajectory determined by the random couplings. The excitatory synapses between successively-excited pools of neurons in this trajectory are then strengthened by Hebbian learning, so that the next time the same stimulus is presented (i.e. the network is started from the same initial condition) the evolution of the system will follow the same trajectory, even in the presence of some noise. Thus it becomes an attractor of the network dynamics, and further presentations of the stimulus can make it more stable. The same can be done for a number of different stimuli. In this way each stimulus is encoded by such an attractor sequence.

We focus our attention on the capacity problem for this system: how many stimuli can be encoded in this way in a network of a given size? Note that this is a different problem from the conventional capacity problem we studied in our previous work. There one asks to store sequences in which every step in the trajectory is specified. Here we only ask that some locally stable trajectory starting from each initial condition exist; no condition is placed on the trajectories except that they avoid each other.

There is a natural tradeoff between the encoding capacity and the stability of the trajectories, which is determined by the learning strength. Greater stability means larger basins of attraction, leaving room for fewer trajectories in the state space. The focus of this paper is to investigate this tradeoff and its consequences for the way a brain that works using the synfire mechanism.

2. Model

We work with a simple model of N binary neurons with firing states $S_i = 1$ (firing) or 0 (not firing). They are connected by synapses of strength J_{ij} which we take to be initially random with unit variance. We adopt n-winner-take-all (nWTA) discete-time dynamics with no memory of the postsynaptic potential from one time step to the next (the limit of small membrane time constant). The equations of motion are

$$S_i(t+1) = \Theta\left(\sum_j J_{ij} S_j(t) - \mu(t)\right) , \qquad (1)$$

where the threshold $\mu(t)$ is chosen such that exactly n of the neurons fire. Later we will add extrinsic noise of variance B to the postsynaptic potential, but we restrict ourselves to the noiseless case for now. Call the threshold for this case μ_0. Since the initial couplings J^0_{ij} are random and N is large, the distribution of postsynaptic potentials h_i is Gaussian, and to make the right number fire, μ_0 must be chosen so that

$$\frac{n}{N} \equiv f = \int_{\mu_0}^{\infty} \frac{\mathrm{d}h}{\sqrt{2\pi}} \mathrm{e}^{-h^2/2} \equiv H(\mu_0) . \qquad (2)$$

We suppose further that the fraction of active neurons is very small: $f = \mathrm{O}(1/n)$. (This means that $n = \mathrm{O}\sqrt{N}$.) It is then legitimate to use the asymptotic form ($x \gg 1$)

$$H(x) \approx \frac{1}{\sqrt{2\pi}x} \mathrm{e}^{-x^2/2} , \qquad (3)$$

which leads to

$$\mu_0 \approx \sqrt{\log \frac{1}{f^2}} . \qquad (4)$$

Before any learning takes place, this network has chaotic dynamics, with the length of its attractors exponential in N.

We will suppose that what a stimulus does is to turn on a particular set of n neurons, prescribing an initial condition for the dynamics. For each such initial condition, the network will evolve through a

different trajectory. Each trajectory can then be seen as the network's encoding of the stimulus that initiated it, provided that it

1. can be made sufficiently stable against noise and,
2. does not enter the basin of attraction of a trajectory initiated by another stimulus we also wish to encode.

We are interested in following the dynamics for some number of time steps, T, for a stimulus set of some size M, i.e. in stabilizing M trajectories, each of length T. The total number of steps is MT, and we will measure the capacity as the ratio of this number to the network size:

$$\alpha = \frac{MT}{N} \,. \tag{5}$$

We will suppose (for reasons of convenience rather than realism) that the system does "batch" learning. That is, the network is run through all M trajectories and then synaptic strengths are changed. We use a Hebb rule with a one-step delay between pre- and postsynaptic activity:

$$\Delta J_{ij} = \frac{\epsilon}{n} S_i(t) S_j(t-1) \,. \tag{6}$$

One cycle of learning then leads to synapses

$$J_{ij} = J_{ij}^0 + \frac{\epsilon}{n} \sum_{\tau=1}^{T} \sum_{m=1}^{M} S_i^m(\tau+1) S_j^m(\tau) \,. \tag{7}$$

where $S_i^m(t)$ gives the firing state of neuron i at time t after the untrained network was started out in the configuration $\{S_i^m(0)\}$ by stimulus number m.

So far we have supposed that just one of the trajectories is activated at any given time. However, such sparse patterns as we are considering can propagate essentially independently of each other if the constraint in the n-WTA dynamics allows several of them to be initiated. Thus we will also consider the case where the number of active neurons is equal to pn, with p an integer. In this case the dynamics should better be called "np-winner-take-all".

3. Learning

It is helpful in studying the effect of the learning to study the PSP distribution at a given time for

1. the neurons that fired at this time during the first evolution of the network from the initial state imposed by one stimulus, and
2. the rest of the neurons.

By definition, these are the portions of the initial Gaussian above and below μ_0:

$$P_+^0(h) = \frac{\Theta(h-\mu_0)}{\sqrt{2\pi f}} \mathrm{e}^{-h^2/2} \,, \tag{8}$$

$$P_-^0(h) = \frac{\Theta(\mu_0-h)}{\sqrt{2\pi(1-f)}} \mathrm{e}^{-h^2/2} \,. \tag{9}$$

Let us look at the learning from the frame of reference of a particular time step t_0 in the evolution of one of the trajectories, m_0. The n^2 synapses connecting the neurons active at the previous step with those active at the present step are strengthened by ϵ/n as a result of this step in this trajectory. The PSP on the sites active at the present step are thereby increased by ϵ, pushing the distribution $P_+(h)$ up by ϵ. In addition, the terms in Eq. (7) from $m \neq m_0$ and $t \neq t_0$ give rise to a Gaussian random contribution to h for all neurons. Its mean is irrelevant, since a uniform shift of PSPs does not affect the n-WTA dynamics. Its variance is [see Ref. 7, Eq. (11)]

$$\epsilon^2 A = \epsilon^2 \alpha f p (1 + n f p) \,. \tag{10}$$

The resulting PSP distributions are thus obtained by convolving Eq. (8), shifted upward by ϵ, and Eq. (9) with Gaussians of this variance:

$$P_+(h) = \frac{1}{f} \int_{\mu_1}^{\infty} \frac{\mathrm{d}h'}{\sqrt{2\pi}} \mathrm{e}^{-\frac{1}{2}h'^2} \frac{1}{\sqrt{2\pi\epsilon^2 A}} \mathrm{e}^{-\frac{(h-\epsilon-h')^2}{2\epsilon^2 A}} \,, \tag{11}$$

$$P_-(h) = \frac{1}{1-f} \int_{\infty}^{\mu_1} \frac{\mathrm{d}h'}{\sqrt{2\pi}} \mathrm{e}^{-\frac{1}{2}h'^2} \frac{1}{\sqrt{2\pi\epsilon^2 A}} \mathrm{e}^{-\frac{(h-h')^2}{2\epsilon^2 A}} \,. \tag{12}$$

The n-WTA dynamics impose the condition

$$f = \int_{-\infty}^{\mu_1} [(1-f) P_-(h) + f P_+(h)] \mathrm{d}h \,, \tag{13}$$

to fix the new threshold μ_1. Then the probability that a neuron which should fire at t (a neuron for which $S_i^m(t) = 1$) does so is

$$m_1 = \int_{\mu}^{\infty} P_+(h) \mathrm{d}h = 1 - \int_{-\infty}^{\mu} P_+(h) \mathrm{d}h \,. \tag{14}$$

For small enough ϵ, the second Gaussian factors in Eqs. (11) and (12) will be much narrower than the first ones, so

$$P_+(h) \approx \frac{e^{-\mu_0^2/2}}{f\sqrt{2\pi}} H\left(\frac{\mu_0+\epsilon-h}{\epsilon\sqrt{A}}\right) \approx \mu_0 H\left(\frac{\mu_0+\epsilon-h}{\epsilon\sqrt{A}}\right) \quad (15)$$

and

$$P_-(h) \approx \frac{e^{-\mu_0^2/2}}{(1-f)\sqrt{2\pi}} H\left(\frac{\mu_0-h}{\epsilon\sqrt{A}}\right) \approx \frac{f\mu_0}{1-f} H\left(\frac{\mu_0-h}{\epsilon\sqrt{A}}\right), \quad (16)$$

where we have again used the asymptotic form [Eq. (3)] to relate μ_0 and f in the inital factors. From Eq. (13) we find immediately that the shift in threshold is exactly $\epsilon/2$, so the probability of a neuron which should fire failing to do so is

$$1-m_1 = \mu_0\epsilon\sqrt{A}\int_{\frac{1}{2\sqrt{A}}}^{\infty} du H(u). \quad (17)$$

This is the error rate for one pool, assuming that there was no error at the preceding step (i.e. the error at the first step after a re-presentation of a stimulus resets the initial state of the network for a post-learning run). After many steps in this run, the error will approach a limit m^* which we can obtain by repeating the above argument with the upward shift of $P_+(h)$ equal to ϵm^* instead of ϵ. The threshold shift is then reduced correspondingly to $\epsilon m^*/2$, and in place of Eq. (16) we find

$$1-m^* = \mu_0\epsilon\sqrt{A}\int_{\frac{m^*}{2\sqrt{A}}}^{\infty} du H(u). \quad (18)$$

Whenever ϵ is small enough to permit the above approximations, the solution of Eq. (16) always gives a fixed point value of $1-m^* \ll 1$. This means that the system will follow almost exactly the same trajectory as before the learning. This is not surprising. We have not introduced any noise, so the system would follow the same trajectory even without the weight changes, which only serve to stabilize the trajectory against possible noise.

If one cycle of learning works this well, why not try another cycle? If we do this, the system will propagate through the same sequence of states and the same weight changes as before, will be made. The result will be the same as if ϵ were twice as large. We can perform many learning cycles (or, essentially equivalently, learn from scratch with a stronger ϵ) until ϵ is so large that our approximations no longer hold. This happens when the second Gaussian factors in Eqs. (11) and (12) no longer vary rapidly relative to the first ones, which is when

$$\epsilon^2 A = \epsilon^2 fp\alpha(1+nfp) \approx \frac{1}{\mu_0^2} = \frac{1}{\log(1/f^2)}. \quad (19)$$

But this equation without the factor of ϵ^2 is just the condition for the critical capacity α_c^0 for storing specified sequences (*tabula rasa* learning)[7]:

$$fp\alpha_c^0(1+nfp)\log(1/f^2) = 1. \quad (20)$$

Thus, when the net learning strength ϵ reaches unity, the capacity is limited by α_c^0. For smaller ϵ we have

$$\alpha_c(\epsilon) = \frac{\alpha_c^0}{\epsilon^2}. \quad (21)$$

Thus we can store synfire sequences at well above the capacity limit for specified sequences, provided the normalized learning strength is less than unity, i.e. that the shifts in PSPs arising from the synaptic changes during learning are smaller than the PSP's before learning. The capacity at $\epsilon = 0$ (before learning) is infinite by construction; learning serves to stabilize the trajectories at the cost of reducing the capacity to a finite value.

Near the capacity limit, the performance of the network is extremely sensitive to noise. The effect of noise in the model is to add an extrinsic term B (the variance of the added noise) to the intrinsic ("crosstalk") noise of Eq. (10). Then the factors of $\epsilon^2 A$ that appear in the above development become

$$\epsilon^2 A = \epsilon^2\alpha fp(1+nfp) + B. \quad (22)$$

There is a critical value of B, found by using this $\epsilon^2 A$ in the condition of Eq. (19),

$$B_c = \frac{1-\alpha/\alpha_c(\epsilon)}{\log(1/f^2)}, \quad (23)$$

beyond which the trajectories lose stability. Even in the most favorable case, where the load is very small ($\alpha \ll \alpha_c(\epsilon)$), the noise has to be smaller than $1/\log(1/f^2)$, and as one approaches the capacity limit $\alpha = \alpha_c(\epsilon)$, the sensitivity to noise diverges.

4. Simulation

To test the capacity calculation we performed simulations. In these we draw couplings J_{ij}^0, from a Gaussian distribution with zero mean. We then normalized the couplings on each neuron so that

$$\frac{1}{N}\sum_{j=1}^{N}(J_{ij}^0)^2 = \frac{1}{n}\,. \tag{24}$$

This reduces finite size fluctuation effects. We performed learning in batch mode for T steps using the update rule

$$\Delta J_{ij} = \frac{\epsilon}{n}S_i(t)\left(S_j(t-1) - \frac{1}{n}h_i(t-1)J_{ij}\right)\,. \tag{25}$$

The second term ensures that J_{ij} remains approximately normalized. We then compared the firing sequence $S_i(t)$ and $S_i'(t)$ generated by the original couplings J_{ij}^0 and the new set of couplings J_{ij} for T time steps. We define the critical capacity, T_c as the average length of sequence we can learn such that the firing sequences overlap by at least 50% for all T_c time steps. In Fig. 1 we show T_c versus the learning strength ϵ for systems of size $N = 100$ and $N = 200$ with $f = n/N$ of 5%. The finite size effects are very considerable. In particular, the storage capacity is an order of magnitude smaller than that for the asymptotic regime $N \to \infty$ and $f \to 0$. Nevertheless it is clear that as the strength of learning increases the capacity falls asymptotically toward its value for *tabula rasa* learning.

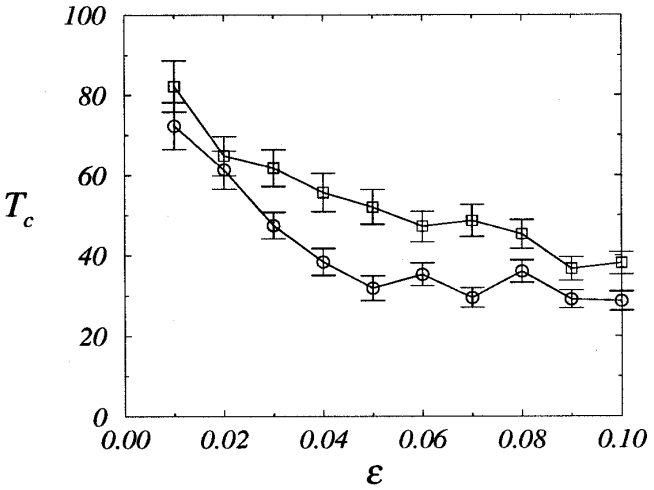

Fig. 1. The capacity T_c is shown as a function of the learning rate ϵ for systems of size $N = 100$, (○), and $N = 200$, (□), with $f = n/N$ of 5%. Each data point was averaged over 100 different random matrices. The *tabula rasa* critical storage capacity is 13.4 ± 1 for $N = 100$ and 18.5 ± 2 for $N = 200$.

5. Discussion

The theme of this conference is the role of noise in biological systems, particularly the brain. In this contribution we have sketched a picture of cortical computation in which the distinction between signal and noise is subtle. The network acts as a very crude model of a cortical column, which we can think of as one processor in the brain's multiprocessor architecture. It encodes afferent signals by the random temporal activity patterns evoked by those inputs. By virtue of Hebbian learning, these patterns acquire stability, allowing them to become signal rather than noise for parts of the brain which receive signals from this local network. Thus we have a case of noise acquiring reproducibility and thereby the potential to become signal as a result of a microscopically simple self-organization process.

Although it is difficult to apply our simple model calculations directly to real brains, some qualitative conclusions about the nature of learning in such systems can be expected to be robust. Too-weak learning will not be robust against noise, but too-strong learning will unnecessarily limit capacity. It can be of interest to study how the brain balances these two priorities in real learning situations.

In our previous investigation we estimated the capacity of a network of 10^4–10^5 neurons (the size of a typical cortical column), starting with the *tabula rasa* formula of Eq. (20) and reducing it with estimates of finite-size corrections and the effects of finite membrane time constant and synaptic dilution. The result was a total capacity of perhaps 10 sequences of length 1 second. This number is just barely on the edge of plausibility, given the experimental observation[6] that at least this many sequences can be active simultaneously. We would like the size of the repertoire of messages that a column can use, to be larger than this so that it can convey information in the set that is active. The extra $1/\epsilon^2$ factor that we find here could easily enhance the capacity by an order of magnitude or more, making room for this necessary degree of freedom.

References

1. We are indebted for this analogy to M. Abeles, who proposed it in a discussion at the conference "Supercomputing in Brain Research", KFA Jülich, November 1994.
2. J. Heller, J. Hertz, T. Kjær and B. Richmond 1995, "Information flow and temporal coding in primate pattern vision," *J. Computational Neurosci.* **2**, 175–193.
3. J. Hertz, J. Heller, T. Kjær and B. Richmond 1995, Information spectroscopy of single neurons," *Int. J. Neural Systems* **6** (suppl), 123–132.
4. M. Abeles 1991, *Corticonics: Neuronal Circuits of the Cerebral Cortex* (Cambridge Univrsity Press).
5. M. Abeles, E. Vaadia, H. Bergman, Y. Prut, I. Haalman and H. Slovin 1993, "Dynamics of neural interactions in the frontal cortex of behaving monkeys," *Concepts in Neuroscience* **4**, 131–158.
6. M. Abeles, H. Bergmann, E. Margalit and E. Vaadia 1993, "Spatiotemporal firing patterns in the frontal cortex of behaving monkeys," *J. Neurophysiol* **70**, 1629–1638.
7. M. Herrmann, J. Hertz and A. Prügel-Bennett 1995, "Analysis of synfire chains," *Network: Computation in Neural Systems* **6**, 403–414.
8. E. Bienenstock 1995, "A model of neocortex," *Network: Computation in Neural Systems* **6**, 179–224.