Network: Computation in Neural Systems
September 2008; 19(3): 236–252

informa
healthcare

Change-based inference for invariant discrimination

REZA MOAZZEZI & PETER DAYAN

Gatsby Computational Neuroscience Unit, Alexandra House, 17 Queen Square, London, WC1N 3AR, United Kingdom

(Received 5 April 2008; revised 30 June 2008; accepted 1 July 2008)

Abstract
Under a conventional view of information processing in recurrently connected populations of neurons, computations consist in mapping inputs onto terminal attractor states of the dynamical interactions. However, there is evidence that substantial information representation and processing can occur over the course of the initial evolution of the dynamical states of such populations, a possibility that has attractive computational properties. Here, we suggest a model that explores one such property, namely, the invariance to an irrelevant feature dimension that arises from monitoring not the state of the population, but rather (a statistic of) the change in this state over time. We illustrate our proposal in the context of the bisection task, a paradigmatic example of perceptual learning for which an attractor-state recurrent model has previously been suggested. We show a change-based inference scheme that achieves near optimal performance in the task (with invariance to translation), is robust to high levels of dynamical noise and variations of the synaptic weight matrix, and indeed admits a computationally straightforward learning rule.

Keywords: *Population coding, network models, perceptual learning*

Introduction

It is seductive to view the complexities of neocortical information processing through the architectural filter of a canonical, repeating, structure formed by a single layer of interconnected neurons (Douglas et al. 1989). A key current candidate for such a structure is the surface attractor recurrent network (Seung 1996; Zhang 1996), which is a (typically) non-linear, but dynamically stable, recurrently coupled net, whose attractor states form a low-dimensional manifold (notably a

Correspondence: R. Moazzezi, Gatsby Computational Neuroscience Unit, Alexandra House, 17 Queen Square, London, WC1N 3AR, UK. Tel: 0044 (20) 7679 1189. Fax: 0044 (20) 7679 1173. E-mail: remo@gatsby.ucl.ac.uk

ISSN 0954-898X print/ISSN 1361-6536 online/08/030236–252
DOI: 10.1080/09548980802314917

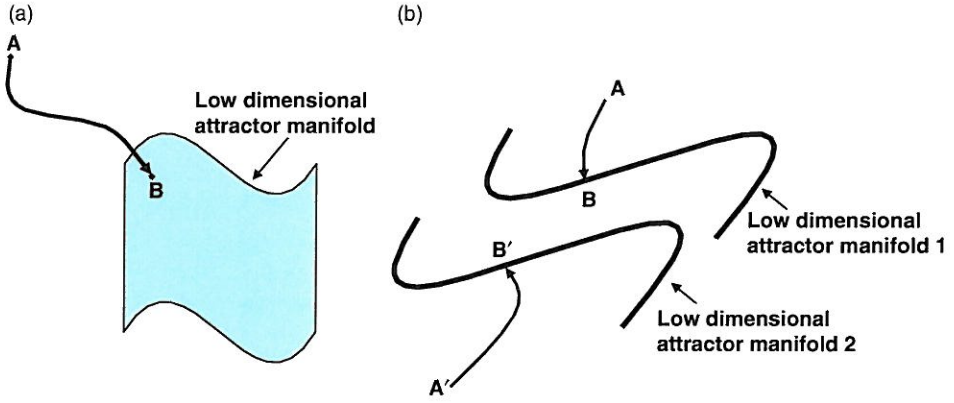

Figure 1. Cartoons depicting standard computational paradigms for networks. (a) Point A indicates the input driven state of the network and point B indicates the state of the network after converging to a point on the low dimensional attractor of the network. (b) A potential solution to bisection task. There are two low dimensional attractors for two possibilities and each line is parameterized by the dimension to which the computation is supposed to be invariant. This figure appears in color in the online edition of this article.

one-dimensional line) in the space of activities of all the neurons. For such nets, the population is taken to *represent* information according to the location on the low-dimensional attractive manifold at which it resides (Figure 1a), and to *compute* by projecting the initial activity of the population to the manifold (Seung 1996; Zhang 1996; Camperi and Wang 1997; Pouget et al. 1998; Compte et al. 2000; Deneve et al. 2001; Wang 2001; Wu et al. 2001; Renart et al. 2003; Wu and Amari 2005).

Two factors militate against this hypothesis. First, some recent studies have shown that the transient activities of populations shortly after stimuli are presented actually admit easy extraction of substantial information. In fact, in some cases, the attractor states themselves are actually less informative. For instance, the activity of the projection cells in the olfactory lobe of insects is seemingly maximally informative during transient excursions (away from both the initial onset and the terminal seemingly-stable state), and with the Kenyon cells, the main targets of these projection cells, being rather unresponsive to the stable state (Mazor and Laurent 2005). This motivates an investigation into computations associated with the transient activity of populations as they evolve towards an attractor. The second problem is that cortical activities rarely exhibit the sort of stable levels that would be consistent with representing converged attractor states. This is true of sensory-driven areas such as V1 (Vinje and Gallant 2000; Reinagel 2001) as well as areas such as prefrontal cortex, which are typically considered as the most likely candidates for implementing attractor structures (Brody et al. 2003).

Once the straightjacket of considering only terminal values is removed, a wider range of possibilities for the readout of information beckons. Here we consider one such – namely, monitoring the *change* in a statistic of the activity of the network over time. We explore a particular computationally desirable characteristic of this, namely, *invariance* to the initial position of the activity with respect to the low dimensional manifold. This makes it possible to separate the pattern expressed in the activity of the network, which can be considered as 'what' information, from exactly 'where' on the manifold this pattern appears. Assessing change in the

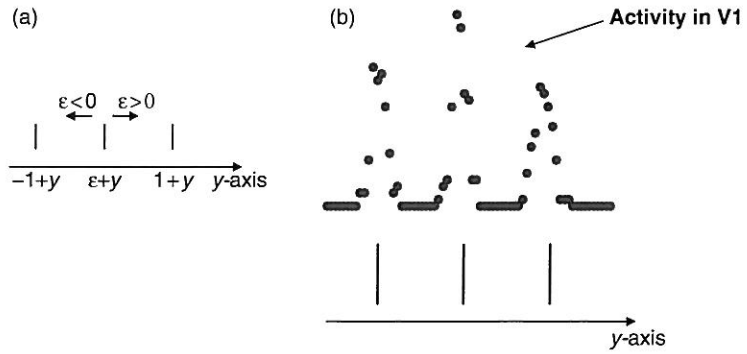

Figure 2. Bisection task and input representation. (a) Bisection task. Three parallel bars are presented, and subjects must decide whether the middle bar is closer to the left or right end bars. The distance between the two outer bars is always fixed at two units of distance. y represents the position of the whole array which changes from trial to trial because of eye movements between trials. ε represents the deviation of the middle bar relative to the middle of the two outer bars, so the task demands assessing whether $\varepsilon \leq 0$. (b) The three bars mentioned in (a) elicit three (Poisson-distributed) noisy bumps of activity in a layer of units. This input is labelled by I (Figure 3). The readout (potentially a member of another layer of units) reports on the sign of ε based on these activities.

network activity focuses purely on the 'what', eliminating any deleterious effect of the 'where'.

Here, we consider a paradigmatic case of a task that has been studied exactly in terms of the invariance issues (Li and Dayan 2001; Zhaoping et al. 2003). The bisection task is a psychophysical problem used to assess perceptual learning (Crist et al. 1997; Fahle and Poggio 2002). In it, subjects see three, small, nearly evenly-spaced bars, and have to decide whether the middle bar is closer to the one on the right or the left (Figure 2a). This decision should be invariant to the overall location on the retina of the stimulus array – and, even if the experimenter always presents it at the same point on the screen, the observers' involuntary micro-saccades imply that it will always shift around (Alpern 1972). Note that after training, humans show positive transfer up to 8° away from the site of learning (Crist et al. 1997). Under the standard surface attractor view, this invariance problem could be solved by embedding two different surface attractors into the recurrent connections, one each for the central bar being nearer left and right outer bars, and making a decision based on one of these two at which the terminal state of the network resides (Figure 1b). Here, we consider instead assessing how the centre of mass of the neural activity evolves over the course of settling of the network (Figure 3). Not only is this change automatically invariant to position, but also, as a readout, we show: (i) its performance is near optimal; (ii) it is surprisingly robust to very high levels of dynamical noise which affects the evolution of the network's activity; and (iii) variations of the recurrent weight matrix. Finally, (iv) we show that such near-optimal weight matrices can be acquired through a (currently non-biologically- plausible) learning procedure.

In Section "The model", we describe the task and the basic structure of the network. In Section "Results", we report results based on hand-crafted weights. In Section "Learning the weights", we show how weights can be learned using the BackPropagation Through Time (BPTT) algorithm. Finally, in Section "Discussion" we consider the relationship to other work.

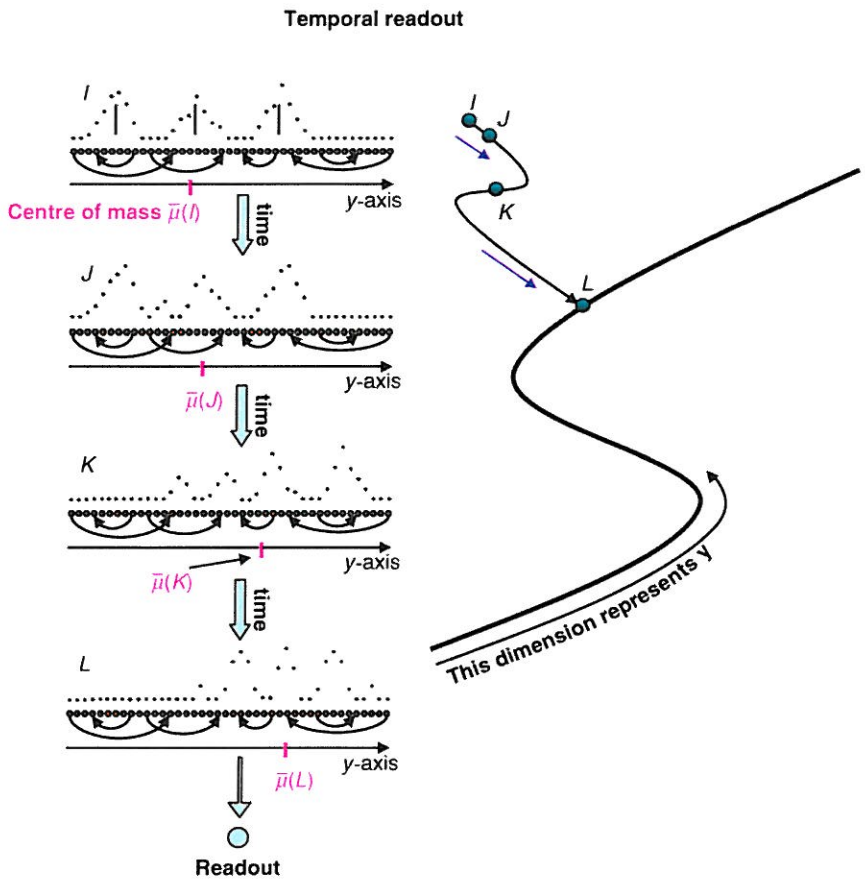

Figure 3. Recurrent processing and change-based readout. Cartoons depicting the change-based readout mechanism for the case of the bisection task. In a recurrent paradigm, lateral connections between the units within a layer define a low dimensional point or line attractor structure in the high dimensional space of the activities of the units. As shown, attraction here is to a line of points defined across y for $\varepsilon = 0$. Blue arrows show the direction of the evolution of the state of the network. The initial activities (I) are mapped through attraction to activities L that represent a point on the line. Conventional, static, readout is based on these activities. By contrast, the change-based readout is based on measuring the change as the activities relax towards the attractor in the centre of mass $\bar{\mu}(K) - \bar{\mu}(J)$ of the activities (magenta lines) between two times (J and K) close to the initial transient. This works as here even with only one line attractor that has no information about the sign of ε. This figure appears in color in the online edition of this article.

The model

Bisection task

Figure 2(a) illustrates the bisection task. It shows the three bars, at positions $-1+y$, $\varepsilon+y$ and $1+y$. The task is to determine the sign of ε, which indicates to which of the two outer bars, the inner bar is closer. The invariance problem comes in the form of the nuisance variable y, which quantifies the position of the overall array in the input, and whose value is irrelevant to the task.

Input. Figure 2(b) shows the abstraction we employed for the coding of the input, which is based on that in Li and Dayan (2001) and Zhaoping et al. (2003). Here, there are $N=321$ input neurons (for simplicity, we only considered units with vertical tuning which is the orientation of the bar). Unit i had preferred topographic position x_i along the line defined by the task. The distance between two neighbouring units, $x_{i+1}-x_i$ was 0.05. Each single bar was assumed to drive the input units according to a Gaussian tuning curve; the mean feedforward input to unit i arising from the three bars was:

$$\bar{a}_i = \lambda e^{-((-1+y)-x_i)^2/2\sigma^2} + \lambda e^{-((\varepsilon+y)-x_i)^2/2\sigma^2} + \lambda e^{-((1+y)-x_i)^2/2\sigma^2}, \tag{1}$$

where σ, the width of tuning curve, was 0.1 and $\lambda = 20\,\text{Hz}$ was the maximum height of each bump. We assumed that the distance between the neighbouring bars was much greater than the tuning width of the units and as a result each active unit was activated non-trivially by only one of the bars. The actual feedforward drive a_i associated with unit i followed a Poisson distribution with mean value $\bar{a}_i$

$$P(a_i|\bar{a}_i) = \frac{e^{-\bar{a}_i}}{a_i!}\bar{a}_i^{a_i} \tag{2}$$

with all the units being independent. The actual input to unit i was ηa_i (Figure 2b).

Processing. Model neurons in the network followed rate-based dynamics for their firing rates $\mathbf{u}(t) = (u_1(t), u_2(t), \ldots, u_N(t))$:

$$\begin{aligned} \tau\frac{\mathrm{d}u_i}{\mathrm{d}t} &= -u_i + \alpha a_i + C + \sum_{j=1}^{N} W_{ij} f(u_j) \\ u_i(0) &= \eta a_i \quad 1 \le i \le N, \end{aligned} \tag{3}$$

where $f(x)=x$ if $x>0$ and $f(x)=0$ if $x\le 0$, $\tau = 20\,\text{ms}$ was the time constant and was the same for all units, $\mathbf{W}=\{W_{ij}\}$ was the recurrent weight matrix. $\mathbf{C}=(C,C,\ldots,C)$ was a uniform non-informative input vector to the network which maintained the activity of the net, and $\mathbf{a}=(a_1,a_2,\ldots,a_N)$ was the stimulus driven (informative) input activity (that initializes the differential equation). To summarize the parameters, we used $\sigma=0.1$, $\tau=20$, $\lambda=20$ and $\eta/C=7.5$ throughout Sections "Results." Two slightly different paradigms for the effect of the feedforward input are popular in the literature. In the first, transient input, case (modelled by $\alpha=0$), the neurons were assumed to be initialized by upstream mechanisms to values reflecting **a**, but then the activities evolved with only the uniform, uninformative, input **C**. In the second, persistent input, case, the network was driven by **a** throughout its evolution (i.e. $\alpha=\eta$). With persistent input, the activities converged, but would never actually have reached the line attractor (in the cases this exists), even in the limit of infinite time.

For the simulations in Figure 6, the dynamic evolution of the net was corrupted by Gaussian, mean-dependent noise. We used a forward Euler discretization (with $\Delta t = 0.2\,\text{ms}$) of the resulting stochastic differential

equation:

$$\tau \Delta u_i(t) = -u_i(t)\Delta t + C\Delta t + \sum_{j=1}^{N} W_{ij}(f(u_j(t))\Delta t + \sqrt{f(u_j(t))}z_j(t)\sqrt{\Delta t})$$
$$u_i(0) = \eta a_i, \quad 1 \leq i \leq N, \tag{4}$$

where $z_i(t) \sim Normal(0, 1)$ was a (independent) random number generated from a Gaussian distribution with mean zero and variance 1, so that the variance of the added noise was $f(u_i(t))$, the same as the mean output. Here, $\Delta u_i(t) = u_i(t + \Delta t) - u_i(t)$.

Recurrent weight matrix. We designed a range of symmetric, translationally-invariant $W_{ij} = W_{|i-j|}$ recurrent weight matrices $\mathbf{W}$ by hand, all of which performed well with the change-based readout method. One key starting point for the design was the hand-crafted weights in Li and Dayan (2001), which involved overall inhibition, to control activity, and carefully positioned peaks to arrange for interactions among the bumps in the input depending on their relative locations.

Learning the recurrent weight matrix. For Section "Learning the weights", we employed discretized dynamics to be able to take advantage of the BPTT learning algorithm. We applied the Polack-Ribiere conjugate gradient method to minimize the cost function which for a single case was:

$$\begin{aligned} L &= \log(g(\Delta)) && \text{if } \varepsilon > 0 \\ L &= \log(1 - g(\Delta)) && \text{if } \varepsilon < 0, \end{aligned} \tag{5}$$

where Δ is the change in the centre of mass during the time between the first and the last iteration and $g(x) = 1/(1 + e^{-\gamma x})$, using $\gamma = 5$ and the activation function used for discretized dynamics was $f(x) = \log(1 + e^{\beta x})/\beta$, using $\beta = 50$.

Evaluation. The overall position of the array, y, was randomly selected from a uniform distribution between -2 and 2. The deviation of the middle bar from the centre of the two outer bars, ε, was between -0.05 and 0.05. The performance of the net was measured as the percentage of correct responses as a function of ε by the readout method described next. The performances measured throughout were based on averaging 400 trials. Thus, the standard deviations of the estimates here and in subsequent graphs are typically less than 0.025, making the error bars too small to see.

Change-based readout

In Li and Dayan (2001), inference was performed by letting the recurrent network evolve to a fixed point, and then mapping the static activity that resulted through a set of feedforward weights that were learned to make classification as accurate as possible. A central innovation of our work here is that rather than trying to readout

the final, converged, state of the network, we considered how a statistic of the activity (the population centre of mass)

$$\bar{\mu}(t) = \frac{\sum_{i=1}^{N} x_i u_i(t)}{\sum_{i=1}^{N} u_i(t)} \tag{6}$$

changed over the evolution of the activity toward the attractor, with the decision being based on whether the centre of mass increases or decreases over time. More formally, the sign of ε was estimated from $\bar{\mu}(t)$ at two different times $t = t_1$ and $t = t_2$ as follows:

$$\bar{\mu}(t_2) - \bar{\mu}(t_1) > 0 \Rightarrow \varepsilon > 0 \tag{7}$$

$$\bar{\mu}(t_2) - \bar{\mu}(t_1) < 0 \Rightarrow \varepsilon < 0 \tag{8}$$

Since the recurrent weights were translation invariant, this decision is completely invariant to the value of y, one of the basic desiderata for the network. The most straightforward way to compute the centre of mass would be to use divisive normalization (Carandini and Heeger 1994) to implement the denominator in a feedforward network that computes the weighted sum in the numerator of Equation 6.

Results

Temporal dynamics

Figure 3 cartoons our new method in the case that the network instantiated just a single line attractor (parameterized by y). In this case, it would be impossible to make any decision about the sign of ε based on the asymptotic activities because the stable pattern is completely invariant to ε. In other words, the initial pattern always converges to the same line attractor independent of the sign of ε and it would be impossible to disentangle the values of y and ε solely from the terminal position on the line.

The curves in Figure 4(a) show examples of the activity of the net 20 and 250 ms after onset, and the asymptotic activities of the network (the corresponding weight matrix is shown in Figure 4b). These indicate that the state of the network was far from equilibrium 250 ms after the activity onset, and, in fact, it only reached asymptote at around 1000 ms.

Further, we can observe the effect of the threshold non-linearity: As time evolves, new neurons that were not super-threshold at the beginning become active, while some of those that were active at the beginning become silent. The weight matrix in Figure 4(b) was handcrafted, starting originally from the design in Li and Dayan (2001). We took the following considerations into account in its design: such a matrix required at least three excitatory bumps to allow crosstalk between units that were activated by different bars. Having five rather than three excitatory bumps allowed those bumps to be at positions that the Fisher information (which depends on the slope of the hill of activity) was highest.

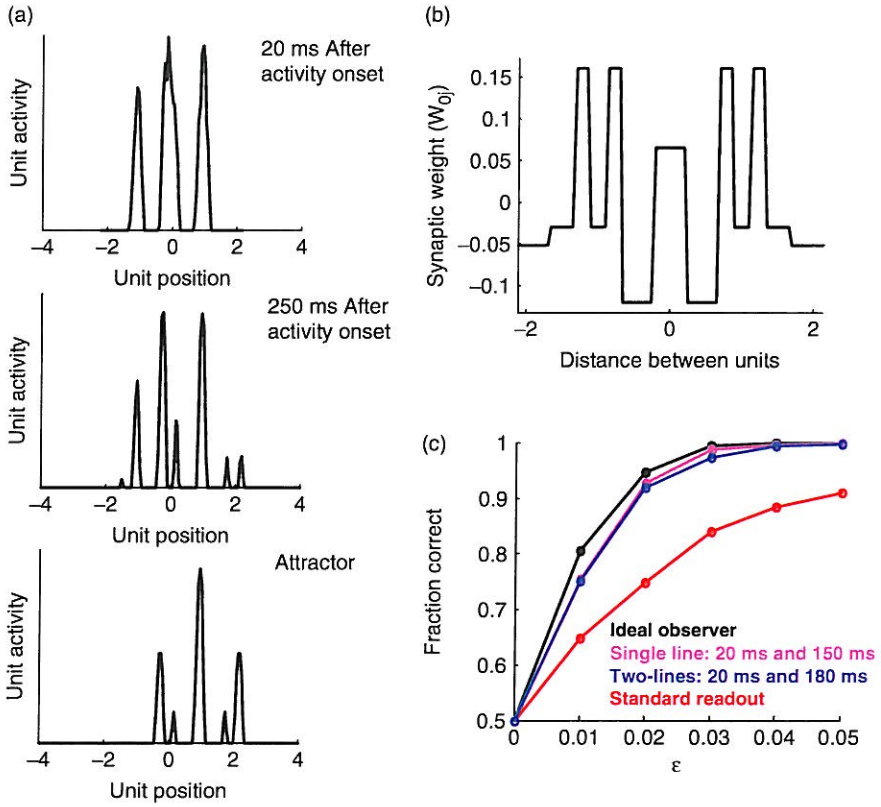

Figure 4. Performance of the network which has a single line attractor. (a) Normalized neural activities on the line attractor (the associated weight matrix is shown in B which has only one line attractor), and an example of the activity of the network after 20 and 250 ms. (b) Central row of the symmetric position invariant synaptic weight matrix $W_{ij} = W_{|i-j|}$ for the case of only one line attractor. For graphical convenience, we represent the weights (and associated attractors) as being continuous, even though only they are really discrete. (c) The performance of the ideal observer (black), previous recurrent network (Zhaoping et al. 2003) for a range of y between -0.6 and 0.6 (red) and the performance of the new line attractor network with only one line attractor (Figure 4b) with change-based readout (magenta). The performance of the latter is based on comparing the centre of mass of the neural activity at 20 and 150 ms after the onset of neural activity. Also, blue shows the performance of the network with two line attractors (Figure 7a) based on the comparing the centre of mass of the neural activity at 20 and 180 ms after the onset of neural activity. Performances are based on averaging 400 trials (the standard deviations of the estimates here and in subsequent graphs are less than 0.025). Note that the performance of the previous recurrent network (which is based on static readout) decreases as the range of y increases while the performance of the change-based readout is independent of the range of y. This figure appears in color in the online edition of this article.

Performance of the change-based readout

Figure 4(c) shows the discrimination performance of the single line attractor network (weights in Figure 4b) based on comparisons between the activity at 20 and 150 ms (magenta), and the ideal observer (black). Change-based readout performs very much better than the previous model (Li and Dayan 2001), which is based on a static readout method (red). One important question is the dependence of the

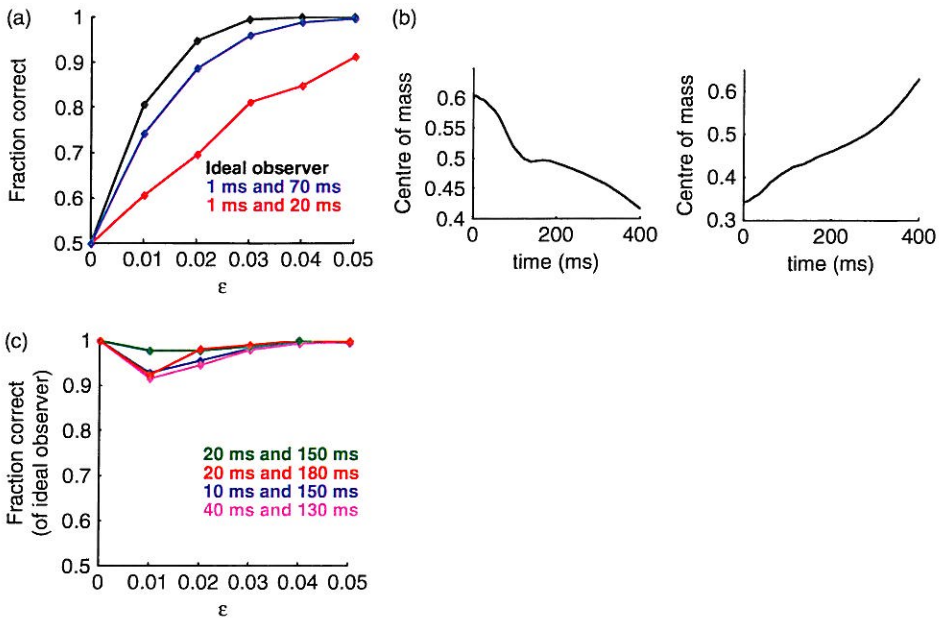

Figure 5. Performance improvement during evolution of neural activity of the network. (a) Performance based on the movement of the centre of mass of the neural activity immediately after (1 ms) the activity onset *vs.* t ms after the onset of neural activity. $t = 20$ ms (red), $t = 70$ ms (blue) and ideal observer (black). (b) Two example trajectories of the centre of mass for the network in Figure 4(b). (c) Performance of the same network comparing centre of mass 20 and 150 ms (green), 20 and 180 ms (brown), 40 and 130 ms (magenta) and 10 and 150 ms (blue) after the neural activity onset as a fraction of that of ideal observer. This figure appears in color in the online edition of this article.

performance level on the delay between the first and the second measurement of the centre of mass. Figure 5(a) shows the discrimination performance of the single line attractor network of Figure 4 based on comparisons between the activity at 1 and 20 ms (red); 70 ms (blue) and the ideal observer (black). The network's performance dramatically improved within 70 ms after the activity onset, and was quickly near optimal. Thus, the direction that the centre of mass moves is highly informative about the sign of ε within 70 ms after activity onset.

Figure 5(b) shows two examples of the evolution of the centre of mass for this network – change-based readout discriminated the sign of ε effectively while the recurrent network was far from the attractor (also clear in Figure 4a). Evidence for this distance from the attractor was that digital selection (Hahnloser et al. 2000) between the different units in the network was still occurring, i.e. the set of units that were supra-threshold was still changing. In fact, with our parameters, performance reached near optimal within the first 150 ms, even though the state did not reach the attractor for a further 850 ms.

Change-based readout performed well even for very different times of comparison of the positions of the centre of mass (performance shown as a proportion of that of the ideal observer in Figure 5c). The main constraints are that these two times of comparison be sufficiently far apart, so that there is measurable evolution between them, and, at least for the single attractor network, that the times are sufficiently

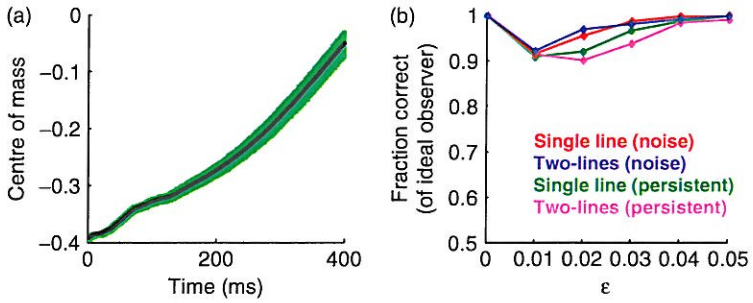

Figure 6. Effect of noise and persistent input during the dynamical evolution of the network. (a) An example of the trajectory of the centre of mass with no dynamical noise (black) and the standard deviation (green) from this mean trajectory as a result of Gaussian mean dependent noise (with equal mean and variance) added to the output of active units during the evolution of the network for the same input as for the case with no dynamical noise. Standard deviations are computed based on 200 trials. (b) Performance as a fraction of that of ideal observer of the one line attractor network (red) and two line attractor network which is shown in Figure 7(a) (blue) in the presence of Gaussian mean dependent noise with variance equal to the mean. Performances are based on averaging 400 trials. The green line shows the performance of the one line attractor network and the magenta line shows the performance of the two line attractor network in the presence of persistent informative input. Performances are based on averaging 400 trials. The difference between the cases of transient and persistent input is that the same input that previously just initialized the network is presented throughout the dynamical evolution in the latter case. This figure appears in color in the online edition of this article.

close to the onset transient so that the critical information has not been suppressed through the process of attraction. This is a key dimension of robustness of the network's performance.

Robustness to dynamical noise

So far, we have assumed deterministic, noise-free dynamics, and thus avoided the potential sensitivity of the change-based readout to perturbative dynamics. To test the effects of noise, we considered a stochastic difference equation version of the dynamics, using Gaussian (rather than Poisson) noise, but making the variance of this noise equal to the mean firing rate (Section "The model").

Figure 6(a) illustrates an example of the rather limited effect of noise on the early evolution of the centre of mass. Although the effect did accumulate as time progressed, the net influence remained quite small over the first few hundred milliseconds of activity that are used by the change-based readout, and therefore its overall performance was not greatly affected (red curve in Figure 6b).

Robustness to variations in weight matrix and number of line attractors

Change-based readout is not only effective in the particular case of a single line attractor (Figure 4b) that most clearly demonstrates its unique computational properties. Rather, it works well in more general cases, including those involving multiple line attractors. As an example, Figure 7(a) shows recurrent weights associated with two main mirror line attractor patterns (Figure 7b), for which

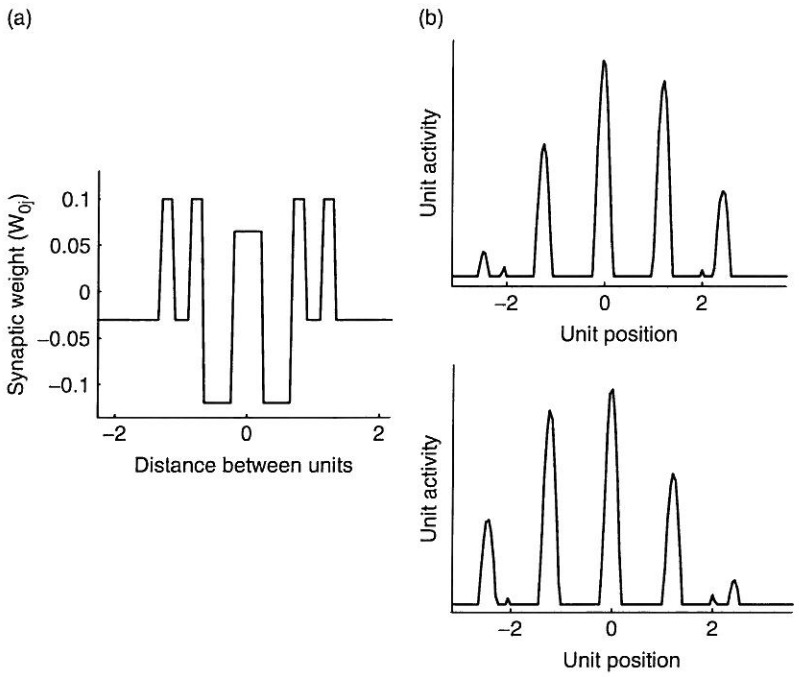

Figure 7. Recurrent network with two dominant line attractors. (a) Central row of the symmetric position invariant recurrent weight matrix with two line attractors. (b) The form of the two dominant line attractors of the network shown in (a).

change-based readout also performed near optimally (Figure 4c; blue). As for the case of the single line attractor, this network was also robust to high levels of noise (Figure 6b; blue).

Figure 8 shows the robustness in another way. Figure 8(a) depicts a further set of handcrafted weights that performed near optimally with and without noise; we were able to find many such. Figure 8(b) illustrates the fact that small changes to the weights also preserve overall performance. It is important to note that while these handcrafted weights are not robust to change in the spacings of the outer bars, it is possible to learn the network for different spacings using BPTT algorithm (data not shown). Recently Herzog and his colleagues (Parkosadze et al. 2008) have demonstrated that it is possible to learn different spacings.

Persistent input

Although these networks were designed specifically for the case of transient input, in which the feedforward input is only informative at the outset, we also tested performance in the case that the input was persistently presented. In this eventuality, the network can be seen as having an attractive (input-dependent) point rather than an attractive line, as the latter is destablized. This destabilization presents a problem for methods depending on the fact that the asymptotic activity of the population lies on the line. By contrast, the relative performance plots in Figure 6(b) (green and magenta lines), which are based on the weight matrix shown in Figure 4(b), show

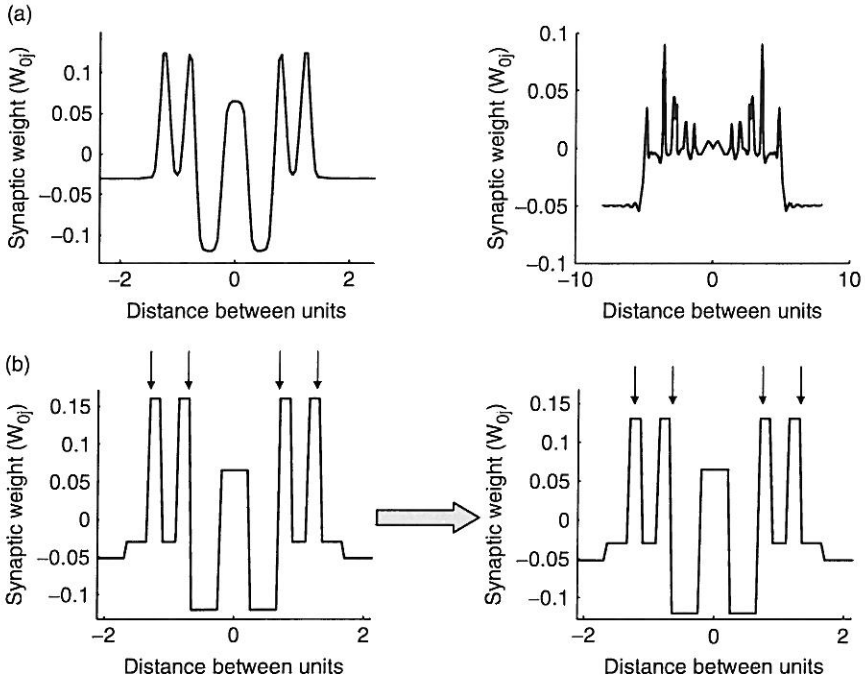

Figure 8. Robustness to variations in weight matrix. (a) Two example weight matrices with different number of line attractors that perform the task near optimally. (b) The two weight matrices shown are similar except that the long range excitation (indicated by the black arrows) of the right one is about 80% of that of the left one. Nevertheless, the performance levels of both of networks are near optimal. This figure appears in color in the online edition of this article.

that persistent presentation of the stimulus did not decrease the performance of change-based readout.

Learning the weights

One additional advantage of not having to wait until the activities asymptote is that the task of learning suitable recurrent weights is made somewhat easier. That good performance is relatively insensitive to small changes in the weights (Figure 8b) also argues that the use of the change-based readout will help smooth the course of gradient-based learning rules. Here we present preliminary results proving that learning such optimal weight matrices is possible for the change-based readout.

In order to explore the possibilities for learning, we considered the use of the non-biologically-realistic learning rule called BackPropagation Through Time (BPTT), in the context of a discrete-time version of the network dynamics. BPTT is a gradient-based method (we used a conjugate gradient minimizer) that operates by 'unfolding' recurrent networks across time-steps, and using the chain rule to work out the derivatives of the error with respect to the weights. For the case of change-based readout, it is obviously only necessary to unfold the network until the later comparison time for the centre of mass.

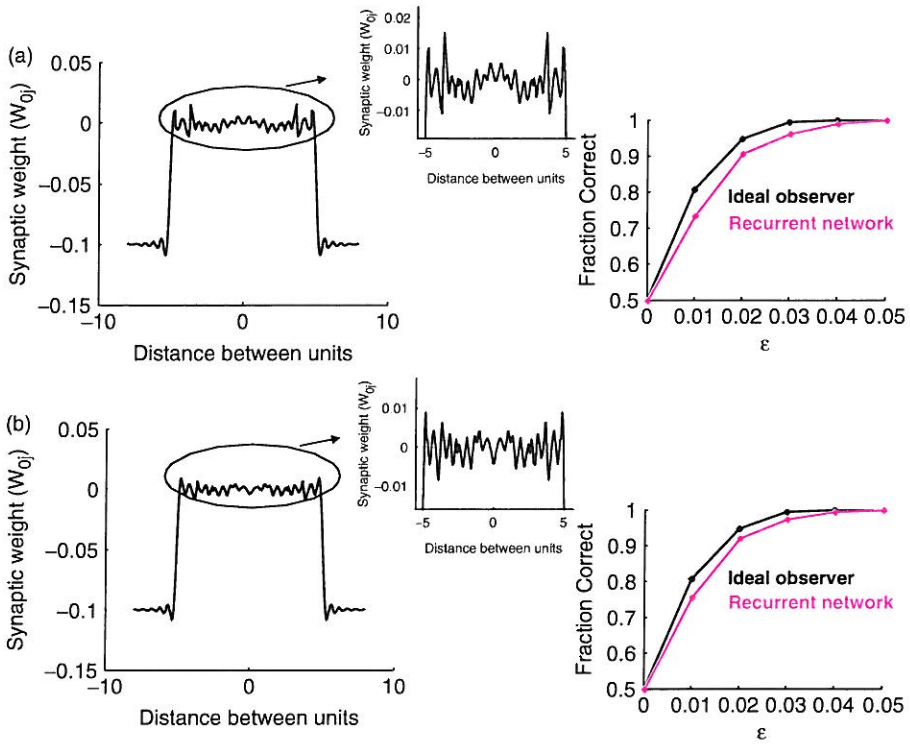

Figure 9. Two example weight matrices that are learned to perform the task near optimally after four (a) and eight (b) iterations. This figure appears in color in the online edition of this article.

We considered the case in which the centre of mass of the input was compared with the centre of mass at either four or eight iterations. Optimizing the weights of non-linear recurrent networks is notoriously difficult in general (particularly given threshold-based activation functions that allow neurons to 'drop out'), and so we imposed some constraints to help, notably starting from weights that achieved moderate (though far from optimal) performance or restricting the range of spatial frequencies in the weights, to make them smooth. Under these circumstances, Figure 9 shows that BPTT was readily able to learn to achieve near-optimal performance. The figure also shows two of the wide variety of examples of near-optimal weights.

Discussion

Non-linear dynamical systems are powerful candidates as neural information processors. The conventional view about such networks is that they operate as forms of point, surface or line attractors, representing information according to which attractor, or where on the continuous attractor, their activity state converges (Seung 1996; Zhang 1996; Camperi and Wang 1997; Pouget et al. 1998; Compte et al. 2000; Deneve et al. 2001; Wang 2001; Wu et al. 2001; Renart et al. 2003;

Wu and Amari 2005). In this article, we studied a different way in which such networks might compute far from the attractor. Precedents for this view come from the locust antennal lobe (Mazor and Laurent 2005). In this system, it has been shown that signal amplification is optimal at a point at which the state of the projection cells that form the output from the lobe is far from a fixed point. Further, the Kenyon cells, that readout the state of the projection cells, are not responsive during the time the state of the projection cells is at a fixed point. This has been shown both when the stimulus has been presented briefly or persistently.

In our case, using the visual rather than the olfactory modality, we examined a specific computational advantage associated with off-attractor computations, namely, the possibility of building invariance to a task-irrelevant dimension directly into the structure of the dynamics and the readout. Discrimination associated with one quantity in the face of invariance to others is a very common requirement for recognition; we used the bisection task as a paradigmatic example. In this task, we showed that by assessing how a statistic of the neural activity (the centre of mass of the underlying population code) changes over time during the evolution of the state of a translation invariant recurrent network, we could perform inferences about the structure of the input pattern (i.e. about the sign of ε) in a way that was invariant to the translation of the whole input array along the retina (i.e. the value of y). It is straightforward to calculate the centre of mass using divisive normalization (Carandini and Heeger 1994) in a feedforward network. However, it would be interesting to explore incorporating the divisive normalization directly into our recurrence, and thereby allow our network to remain appropriately normalized throughout.

We showed that the change-based readout led to near optimal behaviour which was robust against noise corrupting the ongoing activity of the network, noise in the synaptic weights (indeed that performance varied smoothly with the weights, permitted gradient-based learning to work well), different times at which the centre of mass was compared, and also the persistence or otherwise of the external input. We argued that these characteristics weigh in favour of change-based readout in comparison with the classical view of attractor-based computations.

Indeed, there are at least three such classical solutions for the bisection task against which change-based readout might be compared.

One solution is to use two separate line attractor networks, one for $\varepsilon > 0$, a second for $\varepsilon < 0$, both parameterized continuously by the value of y. The trouble with this is that it is difficult to decide to which line attractor the network has converged in the face of varying y. In response to this, a second idea, employed in the work that inspired ours, was to destabilize the two line attractors so that they converge to single points each (Li and Dayan 2001; Zhaoping et al. 2003), effectively attached to one single, special, value of y. In turn, the trouble with this is that as the initial value of y strays further from this special value, the network performs less well. Indeed, the performance reported by Li and Dayan (2001) and Zhaoping et al. (2003) is much further away from the ideal observer than is that of our change-based readout method.

A third idea (Latham and Beck, personal communication], based on the recent work on probabilistic population codes by Ma et al. (2006) and Beck et al. (2007)

would be to build a two dimensional, surface, attractor that represents the distribution of values of both y and ε at the attractor itself, and then marginalize across y to work out the likely value of ε. The operations to do this involve a form of divisive normalization. Under certain circumstances, at least for very small amounts of noise, this can be proved to work exactly as well as an ideal observer. This method is inferentially viable, but poses a harder learning problem, requires longer for inference for each case, cannot cope with persistent input, and is likely to be more sensitive to the effects of dynamical and weight-based noise.

In addition, an alternative idea to ours which is more closely associated with transience is to use the dynamics of an amorphously structured network to create a wealth of non-linear temporal and conjunctive filters over the input information (Maass et al. 2002; Jaeger and Haas 2004). The unstructured nature of such networks has the advantage of permitting pluripotentiality; but at the likely expense of chaotic dynamics. By contrast, the more restricted dynamics of our networks are better adapted to the particular computations required.

One obvious empirical direction in which to test change-based readout is its likely greater sensitivity to transient changes in the input. It would be interesting, for instance, to train subjects on the bisection task, and then present cases that are moving over time either along the axis of the array or perpendicular to it, and see whether this induces a short term bias in performance. The prediction of the conventional account is not completely clear; but we would certainly expect inferences based on change-based readout to be biased in the direction of movement (other factors having been excluded). Presenting parts of the pattern at slightly different times could also be illuminating. Finally, it would be worthwhile to study the effect of presenting patterns with different luminosities for the different components.

It has proven to be very difficult to analyse the way that the network achieves its near-optimal performance. Linearization methods are of limited use given that the processing happens very far from the equilibrium state of the network (on the line attractor) and whilst strongly non-linear activations are evident in the activity patterns. We are hopeful that the inferential success of the more limited dynamics used for the learned collections of weights will open new possibilities for analysis.

In general, stable patterns of activity are the exception in the brain rather than the rule, and so it is important to understand the range of possible computations created by the transient evolution of neural activities. Recurrent networks offer an attractive metaphor for many neural computations. By showing a different way in which they can be seen as processing information, we hope to open up a wealth of new possibilities.

Acknowledgements

This work was funded by the Gatsby Charitable Foundation (RM, PD) and the BBSRC, the Wellcome Trust and the EPSRC (PD). We are very grateful to Peter Latham and Li Zhaoping for helpful discussions and comments.

Declaration of interest: The authors report no conflicts of interest. The authors alone are responsible for the content and writing of the paper.

References

Alpern M. 1972. Eye movements. In: Jameson D, Hurvich LM, editors. Handbook of sensory physiology vol VII/4. Berlin: Springer. pp 303–330.

Beck J, Ma WJ, Latham PE, Pouget A. 2007. Probabilistic population codes and the exponential family of distributions. Progress in Brain Research. 165:509–19.

Brody CD, Hernandez A, Zainos A, Romo R. 2003. Timing and neural encoding of somatosensory parametric working memory in macaque prefrontal cortex. Cerebral Cortex (New York, N.Y.: 1991) 13: 1196–1207.

Camperi M, Wang X-J. 1997. Modeling delay-period activity in the prefrontal cortex during working memory tasks. In: Bower J, editor. Computational Neuroscience: Trends in Research. New York & London: Plenum Press. pp 273–279.

Carandini M, Heeger DJ. 1994. Summation and division by neurons in primate visual cortex. Science 264:1333–1336.

Compte A, Brunel N, Goldman-Rakic PS, Wang X-J. 2000. Synaptic mechanisms and network dynamics underlying spatial working memory in a cortical network model, Cerebsal Cortex (New York, N.Y.: 1991)10: 910–923.

Crist RE, Kapadia MK, Westheimer G, Gilbert CD. 1997. Perceptual learning of spatial localization: Specificity for orientation, position, and context. Journal of Neurophysiology 78:2889–2894.

Deneve S, Latham PE, Pouget A. 2001. Efficient computation and cue integration with noisy population codes. Nature Neuroscience 4:826–831.

Douglas R, Martin K, Whitteridge D. 1989. A canonical microcircuit for neocortex. Neural Computation 1:480–488.

Fahle M, Poggio T. 2002. Perceptual learning. Cambridge, MA: MIT Press.

Hahnloser RHR, Sarpeshkar R, Mahowald MA, Douglas RJ, Seung HS. 2000. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. Nature 405:947–51.

Jaeger H, Haas H. 2004. Harnessing Nonlinearity: Predicting chaotic systems and saving energy in wireless communication. Science 304:78–80.

Li Z, Dayan P. 2001. In: Leen TK, Dietterich TG, Tresp V, editors. Position variance, recurrence and perceptual learning NIPS 2000. Cambridge, MA: MIT Press. pp 31–7.

Ma WJ, Beck JM, Latham PE, Pouget A. 2006. Bayesian inference with probabilistic population codes. Nature Neuroscience 11:1432–8.

Maass W, Natschläger T, Markram H. 2002. Real-time computing without stable states: A new framework for neural computation based on perturbations. Neural Computation 14: 2531–2560.

Mazor O, Laurent G. 2005. Transient dynamics versus fixed points in odor representations by locust antennal lobe projection neurons. Neuron 48:661–673.

Parkosadze K, Otto TU, Malania M, Kezeli A, Herzog MH. 2008. Perceptual learning of bisection stimuli under roving: Slow and largely specific. Journal of Vision 8(1):5.1–8.

Pouget A, Zhang K, Deneve S, Latham PE. 1998. Statistically efficient estimation using population code. Neural Computation 10:373–401.

Reinagel P. 2001. How do visual neurons respond in the real world?. Current opinion in Neurobiology 11:437–442.

Renart A, Song P, Wang X-J. 2003. Robust spatial working memory through homeostatic synaptic scaling in heterogeneous cortical networks. Neuron 38:473–485.

Seung HS. 1996. How the brain keeps the eyes still. Proceedings of the National Academy of Sciences of the united states of America 93: 13339–44.

Vinje WE, Gallant JL. 2000. Sparse coding and decorrelation in primary visual cortex during natural vision. Science 287:1273–1276.

Wang X-J. 2001. Synaptic reverberation underlying mnemonic persistent activity. Trends in Neurosciences 24:455–463.

Wu S, Amari S. 2005. Computing with continuous attractors: Stability and online aspects. Neural Computation 17:2215–2239.

Wu S, Nakahara H, Amari S. 2001. Population coding with correlation and an unfaithful model. Neural Computation 13:775–797.

Zhang K. 1996. Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: A theory. The Journal of Neuroscience 16:2112–2126.

Zhaoping L, Herzog MH, Dayan P. 2003. Quadratic ideal observation and recurrent preprocessing in perceptual learning. Network 14:233–247.