Fast population coding

Quentin JM Huys [1], Richard S Zemel [2], Rama Natarajan [2] and Peter Dayan [1]

[1] Gatsby Computational Neuroscience Unit, University College London,
Alexandra House, 17 Queen Square, London WC1N 3AR, UK
[2] Department of Computer Science, University of Toronto
6 King's College Road, Toronto, Ontario, Canada M5S 3H5
{qhuys, dayan}@gatsby.ucl.ac.uk, {zemel, rama}@cs.toronto.edu

Preliminary, submitted draft

October 10, 2005

Abstract

Uncertainty arises in neural computations from noisy processing elements and the formally ill-posed nature of many tasks. Taking appropriate decisions requires that uncertainty be represented and manipulated in a self-consistent manner, likely in standard cortical structures such as population codes. There is a rich literature on the capability of populations of neurons to support computations in the face of the two types of uncertainty. However, one major facet of uncertainty has received rather little attention, namely time, as in a dynamic, rapidly changing world, data is only temporarily relevant. Here, we analyse the computational consequences of encoding stimulus trajectories in the activity of populations of neurons. For a simple, instantaneous, analytically tractable encoder, we show how the correlations induced by natural, smooth stimuli lead to a decoding problem that can only be resolved by access to information that is non-local both in time and across neurons. Such encodings are computationally ruinous; we show that there is an alternative, computationally and representationally powerful, code in which each spike contributes independent information, *ie* is independently decodeable.

1 Intro: Representation and computation

Sensory and motor information is represented in the joint activity of large populations of neurons (Barlow, 1953; Georgopoulos et al., 1983). There are by now substantial ideas and data about how these representations are formed (Rao et al., 2002), how information can be decoded from recordings of this activity (Paradiso, 1988; Snippe and Koenderinck, 1992; Seung and Sompolinsky, 1993), and how various sorts of computations, including uncertainty-sensitive, Bayesian optimal statistical processing can be performed through the medium of feedforward and recurrent connections amongst the populations (Pouget et al., 1998; Deneve et al., 2001). Critical issues have emerged from these analyses, notably the existence and significance of correlations between neurons for decoding and computation (Shamir and Sompolinsky, 2004; Seriès et al., 2005), and the importance of various sorts of prior information.

However, albeit with some important exceptions, many theoretical investigations into population coding have so far somewhat neglected a major dimension of coding, namely time. This is despite the beautiful and influential analyses of circumstances in which individual spikes contribute importantly to the representation of rapidly varying stimuli (Bialek et al., 1991; Reinagel and Reid, 2000; Rieke et al., 1997; Johansson and Birznieks, 2004), and the importance accorded to fast-timescale spiking by some practical investigations into population coding (Wilson and McNaughton, 1993; Schwartz, 1994; Brown et al., 1998; Zhang et al., 1998; Brown et al., 1998). The assumption is often made that encoded objects do not vary quickly with time, and that therefore firing *rates* in the population suffice. Even some approaches that consider fast decoding (Brunel and Nadal, 1998; Van Rullen and Thorpe, 2001), treat stimuli as being discrete and separate, rather than as evolving along whole trajectories.

In this paper, we study the coding and decoding (Brown et al., 1998; Zhang et al., 1998) of trajectories in populations of spiking neurons. We consider a regime in which stimuli change rapidly and create a sparse train of spikes; we thus analyse the extension to the case of trajectories of one of the simplest ideas about population codes for static stimuli (Snippe and Koenderinck, 1992). Decoding trajectory information is the most comprehensive computation that can be performed, and is therefore our canonical test. When spiking is sparse, decoding becomes a thoroughly ill-posed problem. Probabilistic prior information about the likely trajectories is critical for solving this problem, and we consider naturally realistic, smooth, Gaussian process priors. Unlike some previous work on decoding in time (Brown et al., 1998; Zhang et al., 1998; Smith and Brown, 2003) we do not confine ourselves to recursively specifiable priors, and can therefore treat smoother cases. Smooth priors render decoding, and likely other computations, intractable, by formally coupling spikes together. This effectively forces decoders to interpret exponentially many spike combinations. We thus consider the possibility of an energy-based (Products of Expert; (Hinton, 1999;

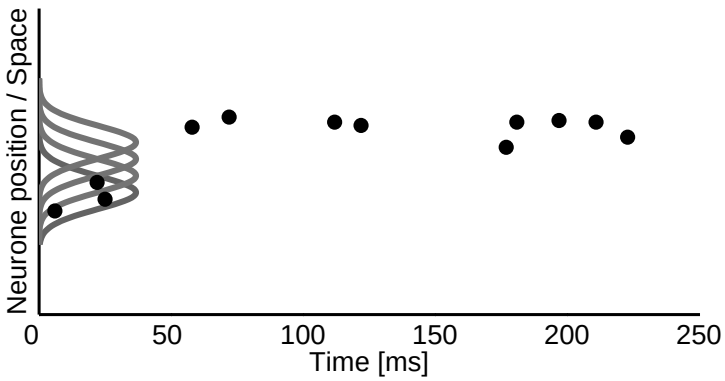

Figure 1: The problem: reconstructing the stimulus as a function of time given the spikes emitted by a population of neurons. If a neuron with preferred stimulus s_i emits a spike at time t, a black dot is plotted at (t, s_i). A few example tuning functions are shown in grey, indicating that the ordinate represents stimulus space and the position of the neurons in that space according to their preferred stimulus s_i.

Zemel et al., 2005)) spike-based *recoding* of the trajectory into a form that more readily supports computations.

Section 2 starts with a simple encoding model. It introduces the need for priors, their shape, and analytical results for decoding in time. Section 3 shows what aspects of the priors determine the availability of information to downstream neurons. We show that the decoder corresponding to the simple encoder can, in time, be complex, meaning that the encoded information is not readily available to downstream neurons. We find that realistic priors lead to a code in which information is not readily available. Finally, in section 4 we propose a representation that has comparable power, but is computationally advantageous.

2 A Gaussian process prior approach

Figure 1 illustrates the general setting of the paper: an array of neurons with partially overlapping tuning functions that emit spikes in response to a changing stimulus. Real-world examples of such a setting include hippocampal neurons with place fields firing as a rat explores an environment, or V1 neurons responding to a target as it moves through their receptive fields. We would like to decode the spikes over time, *ie* recover the trajectory of the stimulus (the animal's position, say) based on the spikes and a knowledge of the neuronal tuning functions (cf Brown et al., 1998; Zhang et al., 1998, for hippocampal examples). In figure 1, the ordinate represents the (1-dimensional) stimulus space, the abscissa time. When a neuron with preferred stimulus s_i emits a spike ξ_t^i at time t, we draw a dot at position (t, s_i). The dots in figure 1 thus represent the spiking activity of an entire population of neurons over time. Our aim is then to find, for each observation time T, a distribution over likely stimulus positions s_T given all the spikes previous to that time. This is related to fitting a line representing the trajectory of the stimulus through the points and is a thoroughly ill-posed problem, given that between the spikes we are not given any information about the stimulus at all.

To solve this ill-posed problem, we have to incorporate additional knowledge in the form of a prior distribution about the stimulus *trajectory*. The prior distribution specifies the temporal characteristics of the trajectories (*eg* how smooth they are), but also whether they live within some constrained part of the stimulus space. Subjects can acquire such prior information from previous exposures to trajectories.

With the aim of gaining analytical insight into the structure of decoding in the temporal scenario, we consider a very simple spiking model $p(\xi_t^i|s_t)$ (Snippe and Koenderinck (1992, cf.) for the static case), augmented with a simple prior over stimulus trajectories $p(\mathbf{s})$. We thereafter follow standard approaches (Zhang et al., 1998; Brown et al., 1998) by doing causal decoding and recovering $p(s_T|\boldsymbol{\xi}_{[0,T)})$ over the current stimulus s_T at time T given all the J past spikes $\boldsymbol{\xi}_{[0,T)} define \{\xi_{t_j}^i\}$, $j = 1 \cdots J$, $i = 1 \cdots N$ emitted at times $0 < \{t_j\}_{j=1}^J < T$ by the population in the observation period ($[0, T)$).

To state the problem in mathematical terms, let $\mathbf{s}_{[0,T)}$ be a vector containing the stimulus at all the J times $\{t_j\}_{j=1}^J$, $t_j \in [0, T)$ at which a spike was was emitted by a neuron in the population, let $p(\boldsymbol{\xi}_{[0,T)}|\mathbf{s}_{[0,T)})$ be the likelihood of observing an entire population spike train $\boldsymbol{\xi}_{[0,T)}$ conditional on the stimulus trajectory $\mathbf{s}_{[0,T)}$ (the spiking model) and let $p(\mathbf{s}_{[0,T)}, s_T)$ be the prior over stimulus trajectories. Using Bayes theorem and an expansion in terms of joint probability allows us to write the distribution of interest as a posterior distribution

$$p(s_T|\boldsymbol{\xi}_{[0,T)}) \propto p(s_T)p(\boldsymbol{\xi}_{[0,T)}|s_T) = p(s_T) \int d\mathbf{s}_{[0,T)}\, p(\boldsymbol{\xi}_{[0,T)}|\mathbf{s}_{[0,T)})p(\mathbf{s}_{[0,T)}|s_T) \tag{1}$$

2.1 Poisson-Gaussian spiking model

The spiking model is as follows: Let $\phi_i(s)$ be the tuning function of neuron i and assume independent, inhomogeneous and instantaneous Poisson neurons (Snippe and Koenderinck, 1992; Brown et al., 1998; Barbieri et al., 2004). The likelihood of a particular population spike train $\boldsymbol{\xi}_{[0,T)}$ given the stimulus trajectory $\mathbf{s}_{[0,T)}$ is then

$$p(\boldsymbol{\xi}_{[0,T)}|\mathbf{s}_{[0,T)}) = \prod_j p(\xi_{t_j}^i|\mathbf{s}_{[0,T)}) = \left(\prod_j \phi_i(s_{t_j})\right) \exp(-\sum_j \phi_i(s_{t_j})) \propto \left(\prod_j \phi_i(s_{t_j})\right) \tag{2}$$

The first equality stems from the assumption that all spiking events (across neurons) are independent given the stimulus. The product over j implies a factorisation both across neurons i and across time (via the instantaneous, inhomogeneous Poisson assumption). Associated to each spike j there is a neuron i according to which neuron emitted the spike $\xi_{t_j}^i$ at time t_j. The final proportionality stems from the assumption of dense tuning function coverage (the sum of the tuning functions is constant for all s at all times). Finally, let us assume squared-exponential (Gaussian) tuning functions

$$\phi_i(s_{t_j}) = \phi_{max} \exp\left(-\frac{(s_{t_j} - s_i)^2}{2\sigma^2}\right)$$

where ϕ_{max} is the maximal firing rate of a neuron and s_i the i^{th} neuron's preferred stimulus. Combining this with our previous assumptions (equation 2) allows us to write the spiking model as

$$p(\boldsymbol{\xi}_{[0,T)}|\mathbf{s}_{[0,T)}) = \phi_{max} \exp\left(-\frac{(\mathbf{s}_{[0,T)} - \boldsymbol{\theta})^T(\mathbf{s}_{[0,T)} - \boldsymbol{\theta})}{2\sigma^2}\right) \tag{3}$$

where the spikes from the entire population have been ordered in time; the j^{th} component of both $\mathbf{s}_{[0,T)}$ and $\boldsymbol{\theta}$ correspond to the j^{th} spike and are, respectively, the stimulus at that spike's time t_j and the preferred stimulus s_i of the neuron that produced it. Note that time is continuous here.

2.2 Gaussian process prior

The prior defines a density over possible stimulus trajectories, and thus a joint distribution over the stimulus values at those times at which spikes are observed in the population. For a finite set of spikes, this will be a finite set of values which can be represented by a vector, as mentioned above.

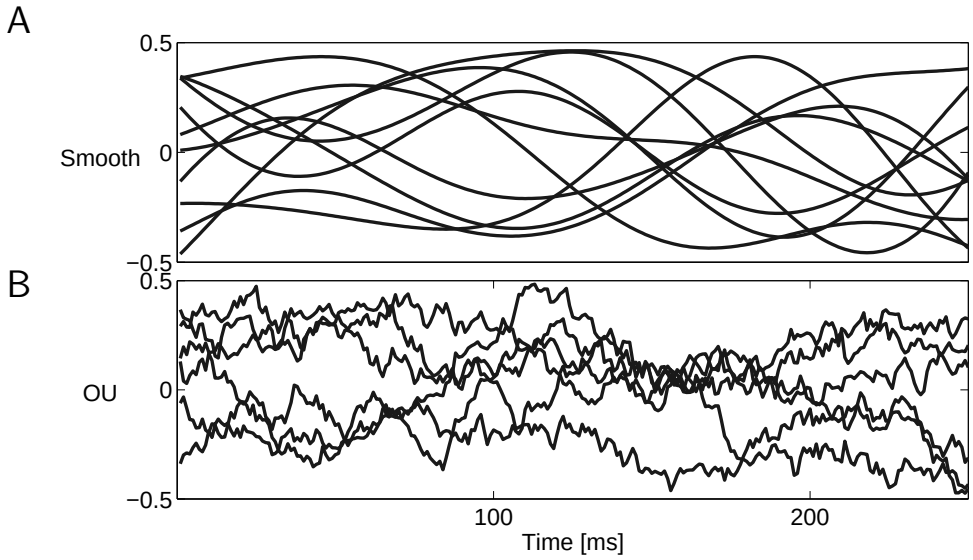

Figure 2: Example trajectories drawn from the prior distribution in equation 4. **A** shows examples for the smooth covariance matrix with $\zeta = 2$, and **B** for the OU covariance matrix, $\zeta = 1$.

One popular prior is a Gaussian process (GP) MacKay (2003), for which the joint distribution of the stimulus at times at which spikes were observed ($\mathbf{s}_{[0,T)}$) and at the observation time (s_T) is a multivariate Gaussian with mean $\mathbf{m}$ and covariance matrix $\mathcal{C}$

$$p(\mathbf{s}_{[0,T)}, s_T) \sim \mathcal{N}(\mathbf{m}, \mathcal{C}) \qquad \mathcal{C}_{t_j t_{j'}} = c \exp\left(-\alpha \|t_j - t_{j'}\|^{\zeta}\right) \tag{4}$$

Note that $\mathbf{s}_{[0,T)}$ is a vector because it contains the stimulus at the discrete set of times at which we have observed spikes; time itself is still treated as being a continuous variable. The parameter $\zeta \geq 0$ dictates the smoothness and the correlation structure of the process. $\zeta = 0$ is the static case which assumes the stimulus does not vary over time. Setting $\zeta = 1$ corresponds to assuming that the stimulus evolves as a Ornstein-Uhlenbeck (OU) or first-order autoregressive process. This is the generative model underlying Kalman filters (Twum-Danso and Brockett, 2001) and generates an autocorrelation with the Fourier spectrum $1/f^2$. We will generalise this to n^{th} order autoregressive processes. At the opposite end of the spectrum is $\zeta = 2$, for which trajectories are smooth and non-Markovian. The parameter α dictates the temporal extent of the correlations and c their overall size (c also parametrises the scale of the overall process). Example trajectories drawn from these priors for $\zeta = \{1, 2\}$ are shown in figure 2. For most of the paper, we will let $\mathbf{m} = \mathbf{0}$ without loss of generality. Assuming a GP prior with a particular covariance matrix is equivalent to regularising the autocorrelation of the trajectory.

2.3 Posterior

With these assumptions we can write down the posterior distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$ analytically by solving equation 1. It is a simple Gaussian distribution with mean $\mu(T)$ and variance $\nu^2(T)$ given in terms of tuning function widths σ, the vector $\boldsymbol{\theta}$ and the covariance matrix $\mathcal{C}$. For clarity, we suppress the subscript $_{[0,T)}$ in this section.

All three terms inside the integral of equation 1 are now known. The conditional distribution $p(\mathbf{s}|s_T)$ is given in terms of the partitioned covariance matrix $\mathcal{C}$:

$$p(\mathbf{s}|s_T) = \mathcal{N}(\mathbf{s}|\mathcal{C}_{[0,T)T}\mathcal{C}_{TT}^{-1}s_T, (\mathcal{C}_{[0,T)[0,T)} - \mathcal{C}_{[0,T)T}\mathcal{C}_{TT}^{-1}\mathcal{C}_{T[0,T)}))$$

where we abuse notation and let $\mathcal{C}_{[0,T)[0,T)}$ be the covariance matrix between all the spike times, $\mathcal{C}_{T[0,T)}$ and $\mathcal{C}_{[0,T)T}$ are vectors with the covariances between the spike times and the observation

time T and $\mathcal{C}_{TT}$ is the marginal (static) stimulus prior at the observation time (constant for the stationary processes considered here). The corresponding partitioning of the matrix $\mathcal{C}$ is

$$\mathcal{C} = \left(\begin{array}{c|c} \mathcal{C}_{[0,T)[0,T)} & \mathcal{C}_{[0,T)T} \\ \hline \mathcal{C}_{T[0,T)} & \mathcal{C}_{TT} \end{array} \right) \tag{5}$$

The remaining two terms in equation 1 are given by $p(s_T) = \mathcal{N}(s|0, \mathcal{C}_{TT})$ and equation 3. As the integral of equation 1 is a convolution, the variances add and the integral evaluates to

$$p(\boldsymbol{\xi}|s_T) \quad = \quad \mathcal{N}(\boldsymbol{\theta}|\mathcal{C}_{[0,T)T}\mathcal{C}_{TT}^{-1}s_T, (\mathcal{C}_{[0,T)[0,T)} - \mathcal{C}_{[0,T)T}\mathcal{C}_{TT}^{-1}\mathcal{C}_{T[0,T)}) + \mathbf{I}\sigma^2)$$

and we only need to calculate the final product with $p(s_T)$ and then renormalise. Application of the Sherman-Morrison-Woodbury formula (the matrix inversion lemma) then leads to

$$\mu(T) \quad = \quad \mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T) \cdot \boldsymbol{\theta}(T) \tag{6}$$

$$\nu^2(T) \quad = \quad \mathcal{C}_{TT} - \mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T) \cdot \mathcal{C}_{[0,T)T} \tag{7}$$

$$\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T) \quad = \quad \mathcal{C}_{T[0,T)}(\mathcal{C}_{[0,T)[0,T)} + \mathbf{I}\sigma^2)^{-1} \tag{8}$$

The mean $\mu(T)$ of the posterior is thus a weighted sum of the preferred positions of those neurons that emitted particular spikes. The weights are given by what we term the *temporal kernel* $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$. As we will see, the weight given to each spike will depend strongly on the time at which it occurred. A spike that occurred in the distant past will be given small weight. As for conventional Kalman filters, the posterior variance depends only on the spike times, not on their identities. That this is true depends on the squared exponential nature of the tuning functions ϕ and other tuning functions may not lead to this quality. However, it will not affect most of the conclusions reached below. This posterior distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$ is well-known in the GP literature as the predictive distribution (MacKay, 2003, chapter 45).

2.4 Structure of the code

The operations needed to obtain the posterior $p(s_T|\boldsymbol{\xi}_{[0,T)})$ provide insight into the structure of the code, and how it depends on the prior. If the posterior is a function of combinations of spikes, postsynaptic neurons have to have simultaneous access to all those spikes. This point will be critical in temporal codes, as the spikes to which access is required are spread out in time. Only if spikes are interpretable independently, can they be forgotten once they have been used for inference. All information the spikes contribute to some future time $T' > T$ is then contained within $p(s_T|\boldsymbol{\xi}_{[0,T)})$. If the posterior depends on combinations of spikes (as will be the case for natural, smooth priors), information that can be extracted from a spike about times $T' > T$ is *not* entirely contained within $p(s_T|\boldsymbol{\xi}_{[0,T)})$. As a result, past spikes have to be stored and the posterior recomputed using them – an operation that is nonlocal in time. We will show that under natural priors the posterior depends on spike combinations and is thus complex. Decoding for the simple encoder (the spiking model) is thus hard. In section 4, we will illustrate the type of computations ("recoding") a network has to perform to access all the information. This will be equivalent to finding a new, complex encoder in time for which decoding is simple.

3 Effect of the prior

We next study how the temporal kernels $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ from equation 8 and the structure of the code depend on the prior. We analyse the behaviour of the temporal kernels and the structure of the code for a representative set of priors, including those that generate constant, varyingly rough and entirely smooth trajectories.

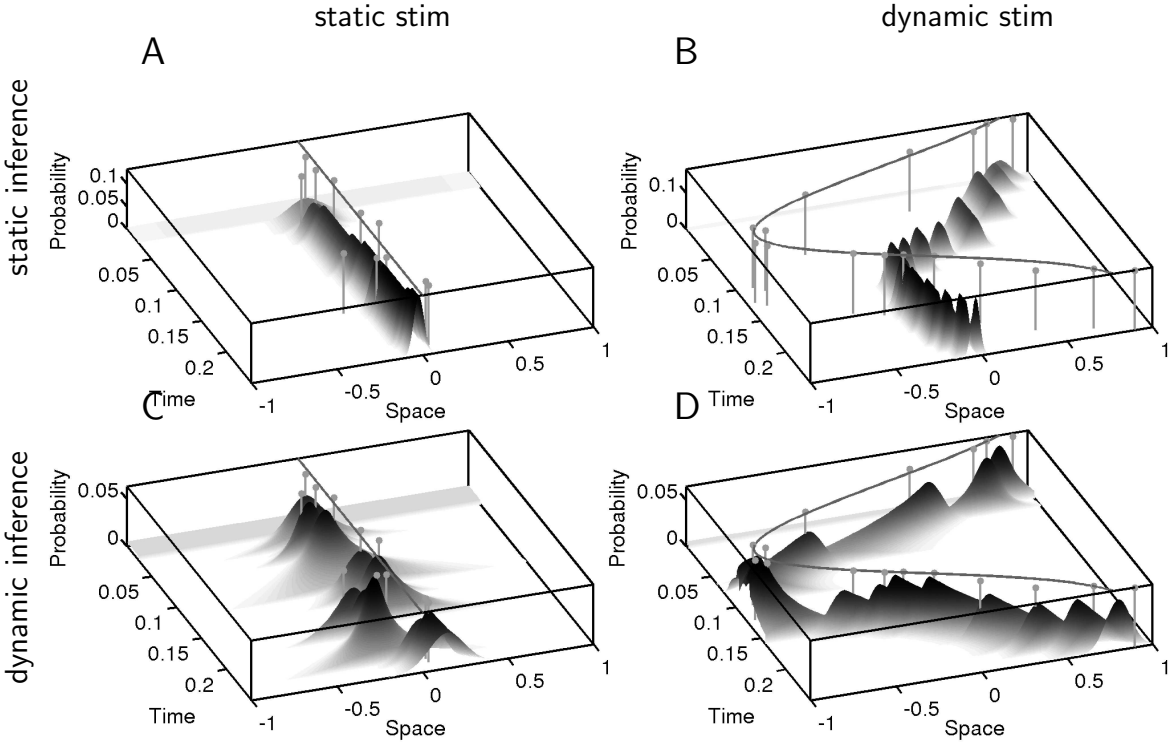

Figure 3: Comparison of static and dynamic inferences. Throughout, the posterior distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$ is indicated by gray shading, the spikes are vertical (gray) lines with dots and the true stimulus is the line at the top of each plot. **A** Static stimulus, constant temporal kernel **B** Moving stimulus, constant temporal kernel. **C** Static stimulus, decaying temporal kernel. **D** Moving stimulus, decaying temporal kernel.

3.1 Constant stimulus prior $\zeta = 0$

We first show that our treatment of the time-varying case is an exact generalisation of the static stimulus case by re-deriving classical results for static stimuli. Snippe and Koenderinck (1992) have shown that the posterior mean and variance (under a flat prior) is given by a weighted spike count

$$\mu(T) = \frac{\sum_i n_i(T) s_i}{J(T)} \qquad \nu^2(T) = \frac{\sigma^2}{J(T)} \tag{9}$$

where $n_i(T) = \int_0^T dt\, \xi_t^i$ is the i^{th} neuron's spike count and $J(T) = \sum_i n_i(T)$ is the total population spike count at time T.

If we let $\zeta = 0$, the matrix $\mathcal{C}_{[0,T)[0,T)} = c\mathbf{n}\mathbf{n}^T$ where $\mathbf{n}$ is a $J(T) \times 1$ vector of ones. Equations 6 can then be solved analytically:

$$\begin{aligned}
\left((\mathcal{C}_{[0,T)[0,T)} + \mathbf{I}\sigma^2)^{-1}\right)_{ij} &= \frac{(\sigma^2 + cJ(T))\delta_{ij} - c}{\sigma^2(\sigma^2 + cJ(T))} \\
\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T) &= \frac{c}{\sigma^2 + cJ(T)}\mathbf{n} \\
\mu(T) &= \frac{c\sum_i n_i(T) s_i}{\sigma^2 + cJ(T)} \\
\nu^2(T) &= \frac{c\sigma^2}{\sigma^2 + cJ(T)}
\end{aligned}$$

which is exactly analogous to equation 9 with an informative prior. The temporal kernel $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ is now constant and $\propto 1/J$. The contribution of each neuron to the mean $\mu(T)$ is given by its spike

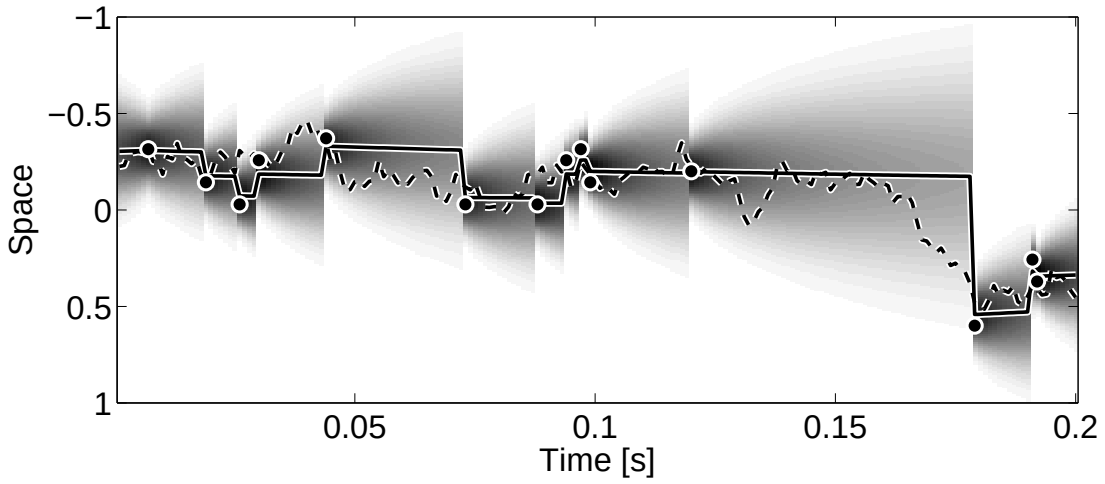

Figure 4: Posterior distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$ for OU prior. Same representation as in figure 1. The dashed line shows the stimulus trajectory used to generate the spikes, the dots are the spikes, the posterior distribution is in grey and the solid line shows the posterior mean.

count $n_i(T)$. Each spike is given the same weight, which is only a sensible approach if spikes are eternally informative about the stimulus. If the stimulus is a varying function of time $s(t)$, spikes at time t' are only informative about the stimulus at times t close to t' and the influence of each spike on the posterior should fade away with time. This is illustrated in figure 3. Figure 3A shows the present static case, where the stimulus does indeed not move. Over time, the posterior $p(s_T|\boldsymbol{\xi}_{[0,T)})$ sharpens up around the true value, but if the stimulus does move, the posterior ends up at the wrong value (figure 3B). Only if the stimulus is static, is never forgetting about spikes the right approach. Static inference corresponds to a constant kernel.

Imagine now that the temporal kernel $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ decays and we forget about spikes in the more distant past. Figure 3C shows that this leads to a posterior that widens inbetween spikes. The posterior is wider than it should be. On the other hand, figure 3D shows how such a decaying temporal kernel would, in contraposition to figure 3B, allow $p(s_T|\boldsymbol{\xi}_{[0,T)})$ to nicely track the stimulus. Dynamic inference corresponds to decaying kernels. In the following, we analyse the behaviour of $p(s_T|\boldsymbol{\xi}_{[0,T)})$ and the optimal temporal kernel $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ for various stimulus autocorrelation functions.

3.2 Non-smooth (Ornstein-Uhlenbeck) prior $\zeta = 1$

Setting $\zeta = 1$ in the definition of the prior (equation 4) corresponds to assuming that the stimulus evolves as a random walk with drift to zero (an Ornstein-Uhlenbeck process):

$$\frac{ds}{dt} = -\beta s(t) + c\sqrt{dt}\, d\eta(t) \tag{10}$$

with Gaussian noise $\frac{d\eta}{dt} \sim \mathcal{N}(0,1)$ and $0 \leq \beta \leq 1$. The Ornstein-Uhlenbeck process is the underlying generative process assumed by standard Kalman filters. The simplicity of Kalman-filter like formulations explains some of its wide applicability and success (*eg* Brown et al., 1998; Barbieri et al., 2004). However, as indicated visually by the example trajectories in figure 2, the rough trajectories this prior imposes are not a good model of smooth biological movements (see also Discussion).

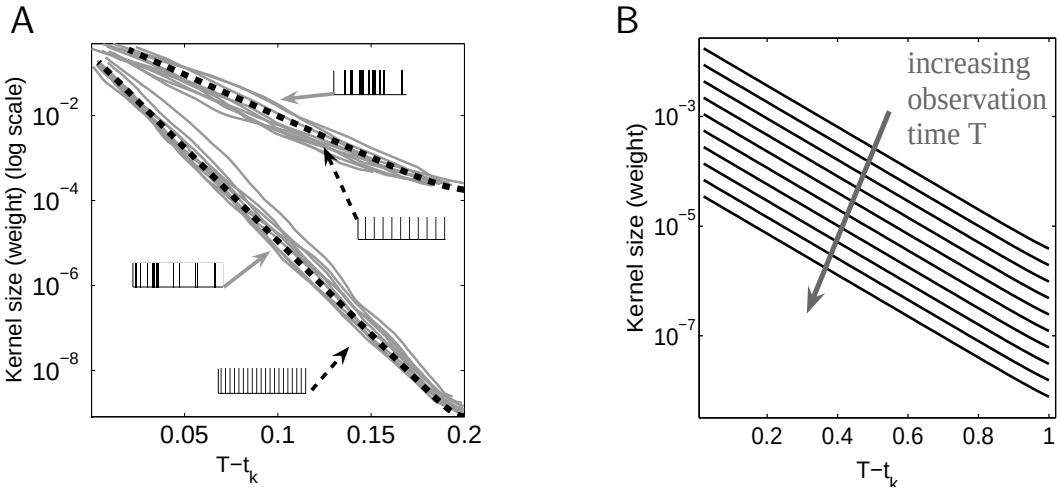

Figure 5: OU temporal kernels, $\zeta = 1$. **A** Example temporal kernels, top traces are for low, bottom for higher average firing rate. The gray traces show temporal kernels for Poisson spike trains. The components of the vector $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ are plotted against the corresponding spike time. The dashed black traces show temporal kernels for regular spike arrivals (metronomic temporal kernels). The true (gray) temporal kernels are relatively closely bunched around the metronomic temporal kernel. **B** The effect of the time since the last spike on the temporal kernel is an overall scaling.

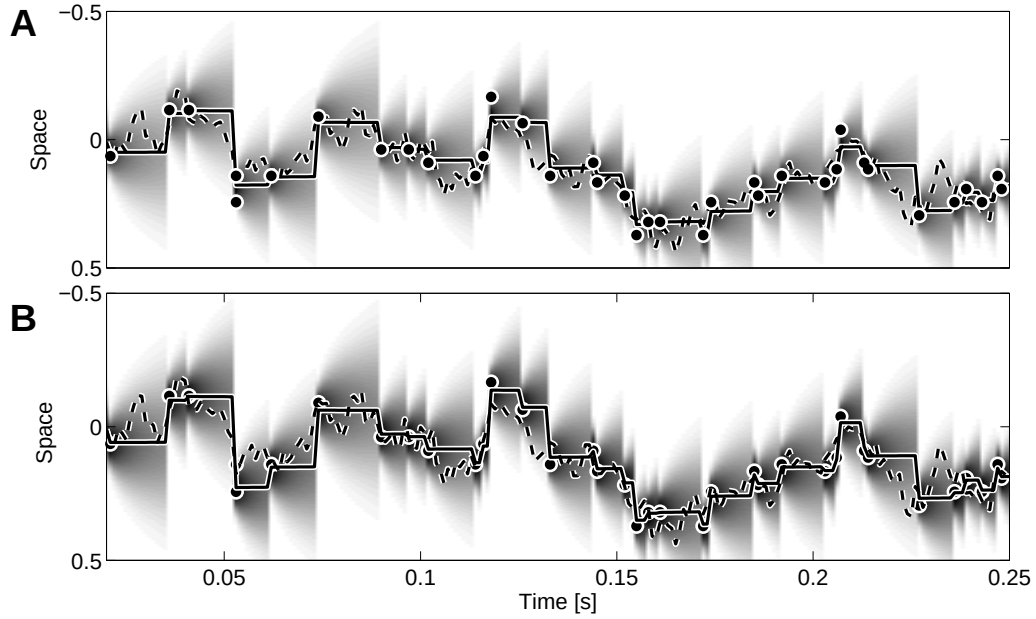

Figure 6: Comparison between exact and metronomic kernels. Same representation as in figure 4. **Top** Exact posterior $p(s_T|\boldsymbol{\xi}_{[0,T)})$; **Bottom** Approximate posterior derived by replacing all ISI's by Δ, but keeping $T - t_J$.

Figure 4 shows the whole set of stimulus trajectory, spikes and posterior distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$. The mean of the posterior does a good job of tracking the true underlying stimulus trajectory and is never more than two standard deviations away from it. Between spikes, the mean simply moves back to zero (albeit slowly in the figure shown).

Figure 5A displays example temporal kernels $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ for inference in this process. They are very close to exponentials (note the logarithmic ordinate). This makes intuitive sense as an OU process is a first-order Markov process as it can be rewritten as a first-order difference equation. In fact, assuming the spikes arrive regularly (*ie* replacing each of the inter-spike intervals (ISI) by their average value $\Delta = 1/J \sum_j t_j - t_{j-1}$) allows us to write the j^{th} component of $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ as

$$k_j \approx d_1 \lambda_1^{j-1}$$

where d_1 is some constant and $\lambda = c\exp(-1/\tau)$ (see appendix A). For such metronomic spiking, $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ is thus really simply a decaying exponential. Similar expressions can be obtained for the original case of Poisson distributed ISI's (appendix A). Figure 5A shows that the metronomic approximation provides a generally good fit, capturing especially the slope of the true temporal kernels, which depends mostly on the correlation length α and on the maximal (or average) firing rate ϕ_{max}. The remaining quality of the fit is influenced most strongly by the match between Δ and the time since the last spike $T - t_J$ (which takes its effects through $\mathcal{C}_{T[0,T)}$ in equation 5 and 6-8), which determines the overall size of the temporal kernel.

The factors influencing the slope of the temporal kernel and its overall size do not interact much, *ie* $T - t_J$ does not affect the slope (shape) of the temporal kernel, only its size, as shown in figure 5B (metronomic temporal kernels are used for clarity, but the argument applies equally to the exact kernel). Conversely though less importantly, Δ affects mostly the slope. Replacing the true temporal kernels by metronomic temporal kernels, *ie* replacing all ISI's by Δ but keeping the time since the last spike $T - t_J$ does not degrade $p(s_T|\boldsymbol{\xi}_{[0,T)})$ much (cf. figure 6A and figure 6B).

To understand the dependence in figure 5B, we write out the integrand of equation 1 in detail for the OU prior and find that it factorises over potentials involving duplets of spikes because $\mathcal{C}^{-1}$ is tridiagonal and the elements of $\mathcal{C}^{-1}$ only involve two spikes.

$$\begin{aligned} p(\mathbf{s}_{[0,T)}, s_T) &\propto \exp\left(-\frac{1}{2}\left[\mathbf{s}_{[0,T)} s_T\right]^T \mathcal{C}^{-1} \begin{bmatrix} \mathbf{s}_{[0,T)} \\ s_T \end{bmatrix}\right) \\ &= \exp\left(-\frac{1}{2}\left(\sum_{j=1}^{J+1} s_{t_j}^2 \mathcal{C}^{-1}_{t_j t_j} + \sum_{j=1}^{J} s_{t_j} \mathcal{C}^{-1}_{t_j, t_{j+1}} s_{t_{j+1}}\right)\right) \\ p(\mathbf{s}_{[0,T)}, s_T) &= \psi(s_T) \prod_{j=1}^{J} \psi(s_{t_j}, s_{t_{j+1}}) \end{aligned} \tag{11}$$

where t_j stands for the time of the last spike, t_{J-1} the time of the penultimate one etc., and the observation time $T = t_{J+1}$. Note that the last equality implies that the determinant also factors over spike pairs. This means that the integrations over each spike in the main equation 1 can be pulled into the integral and the equation can be written in a recursive form akin to that used in message passing algorithms (MacKay, 2003):

$$\begin{aligned} p(s_T|\boldsymbol{\xi}_{[0,T)}) &\propto \psi(s_T) \int ds_{t_j}\, p(\rho_{t_j}|s_{t_j}) \psi(s_T, s_{t_j}) \int ds_{t_{J-1}}\, p(\xi_{t_{J-1}}|s_{t_{J-1}}) \psi(s_{t_j}, s_{t_{J-1}}) \cdots \\ &= \psi(s_T) \int ds_{t_j}\, p(\xi_{t_j}|s_{t_j}) \psi(s_T, s_{t_j}) m_{t_j}(t_{J-1}) \end{aligned} \tag{12}$$

Here $m_{t_j}(t_{J-1})$ is the "message" passed from all spikes up to the penultimate spike to be incorporated into the posterior by multiplying it with $\psi(s_T, s_{t_j})$, *ie* by a simple scaling of the entire temporal kernel. This formulation again reminds us of the Kalman filter equations.

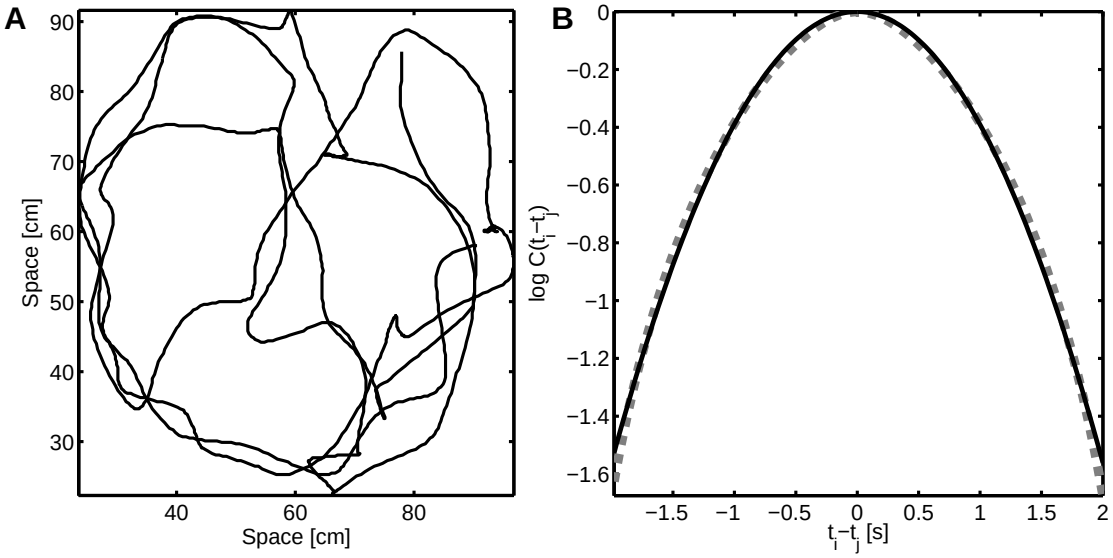

Figure 7: Natural trajectories are smooth. **A** Position of a rat freely exploring a square environment. **B** Covariance function of the position along the ordinate (gray, dashed line) and a quadratic approximation (black, solid line). Note the logarithmic ordinate. The smoothing applied to eliminate artefacts was of a timescale short enough not to interfere with the overall shape of the covariance function.

3.3 Smooth prior $\zeta = 2$

Setting $\zeta = 2$ in the definition of the prior (equation 4) corresponds to assuming that the stimulus evolves as a non-Markov random walk. Trajectories with this autocovariance function are smooth (cf figure 2A shows some trajectories generated from the prior) and infinitely differentiable. The smoothness makes it a more natural and informative prior for Bayesian decoding from movement-related trajectories than non-smooth priors. Figure 7A shows trajectories of a rat exploring a square environment (data kindly provided by (Lever et al., 2002)). Not only are these natural trajectories smooth, but figure 7B also shows that a squared exponential covariance function closely approximates the real covariance function. [1]

Figure 8 shows the equivalent of figure 4 for the smooth case. The posterior $p(s_T|\boldsymbol{\xi}_{[0,T)})$ is shown in the top panel of the figure. The main dynamical difference between inference in this smooth case and inference in the OU case is indicated by the arrows in the figure. While the OU process simply decays back to the mean (here zero for simplicity), the dynamics of the smooth posterior mean are much richer in that, in the absence of spikes, the mean continues in its current direction for a while before reversing back. As can be seen, this gives a better fit to the underlying trajectory (black dotted line) than would otherwise have been achieved. It arises directly from the fact that the correlations extend beyond the last spike (into the entire past in fact). For comparison, figure 9 shows the posterior when the wrong prior is used. The stimulus was generated from the smooth prior, but the OU prior was used to build the posterior. The arrow indicates where the posterior behaves suboptimally, falling back to zero instead of predicting that the stimulus will continue to move further away from zero. In terms of difference equations, the larger extent of correlations intuitively mean that the higher order derivatives of the process are also "constrained" by the covariance $\mathcal{C}$.

[1] Only the centre of the covariance function is shown here. Due to the small size of the environment, the rat runs back and forth the entire available length and there are oscillating flanks to the covariance function for delays larger than those shown.

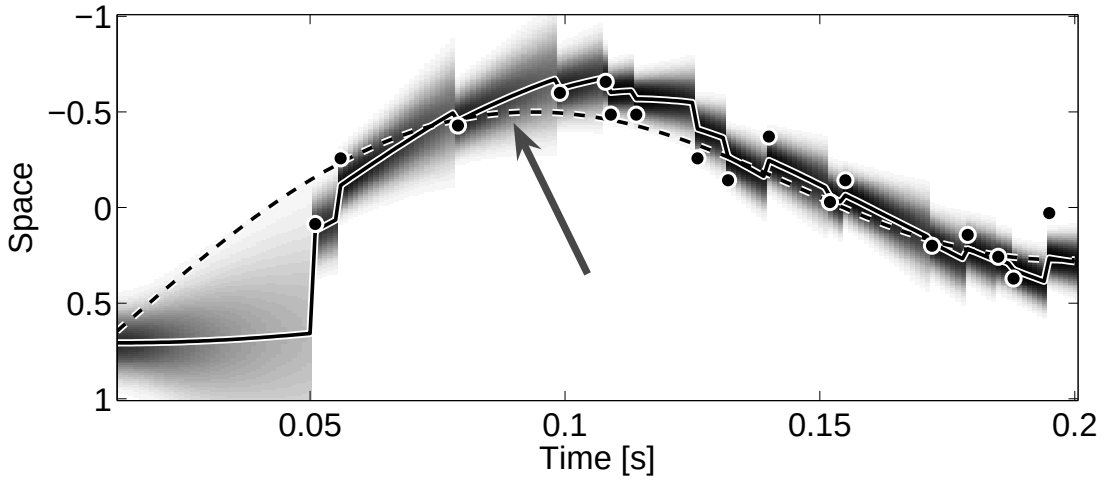

Figure 8: Posterior distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$ for smooth prior. Same representation as in figure 4. The arrow is explained in the main text.

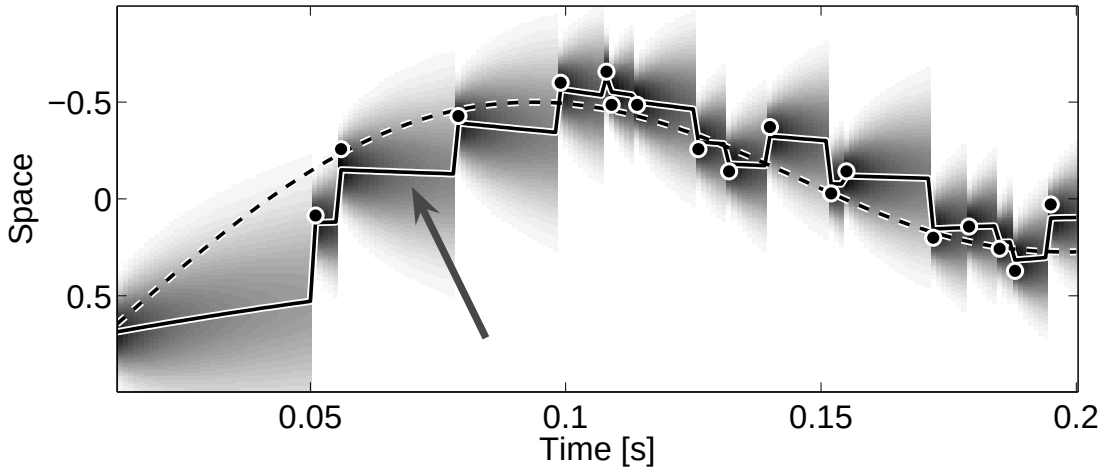

Figure 9: Posterior distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$ for smooth prior. Same representation as in figure 4. The arrow is explained in the main text.

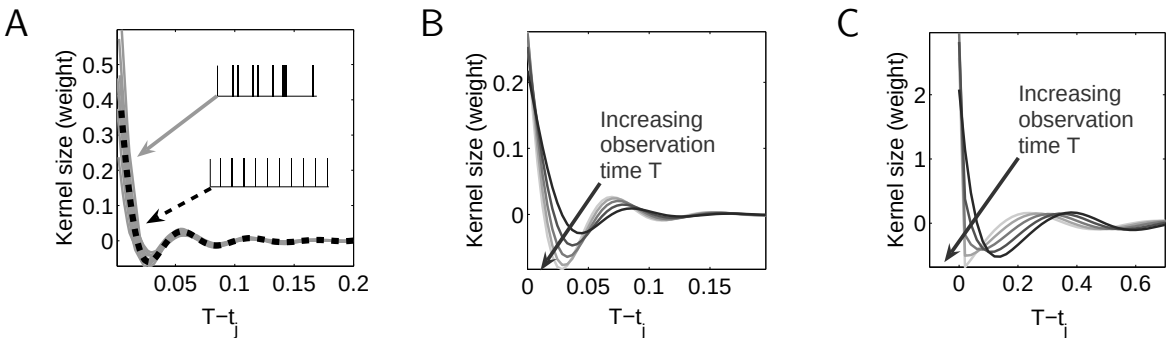

Figure 10: Temporal kernels for the smooth prior **A** shows exact (gray solid) and metronomic (black dashed) temporal kernels for the smooth prior with $\zeta = 2$. **B** shows how the metronomic temporal kernels change as the observation time T is moved away from the last spike. **C** shows the same as panel B, but for the empirical covariance function derived from the rat trajectories.

The simple exponential temporal kernels observed in the OU process cannot give rise to the reversals observed in the smooth process. Figure 10A shows the temporal kernels for the smooth process, which have a distinctively different flavour from the OU temporal kernels, including oscillating terms multiplying the exponential decay. Most importantly, the oscillating terms allow the weight assigned to a spike to dip below zero, *ie* a spike initially signifies proximity of the stimulus to the neuron's preferred position, but later on swaps over, signalling that the stimulus is *not* there any more etc. This feature of the temporal kernels gives rise to the reversals seen in the posterior mean.

As in the OU case, the metronomic temporal kernel based on equal ISI's gives a good description of the temporal kernel mostly for spikes in the more distant past. Replacing the true temporal kernels by metronomic temporal kernels (but keeping the exact time since the last spike $T - t_J$) again does not affect the posterior strongly, but the KL-divergence between the true posterior and the metronomic posterior is larger in the smooth than in the OU case (data not shown), indicating that the exact timing of spikes has greater weight in the smooth inference.

Unlike in the OU case, there is no simple analytic expression for the metronomic temporal kernel (let alone the true temporal kernel). Critically, as shown in figure 10B, changing the time since the last observed spike $T - t_J$ does not simply scale the temporal kernel, but also changes the shape of the temporal kernel (it produces a complicated phase shift of the oscillating component). Again, for clarity, the metronomic kernels are used as an illustration. The same argument applies to the exact kernels. Thus, local structure has complex global consequences in the smooth, but not the OU case. The simple rescaling of the OU temporal kernel by the time since the last spike can be achieved without a memory of all past spikes, by simply scaling the products $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)\boldsymbol{\theta}$ for $\mu(T)$ or $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)\mathcal{C}_{[0,T)T}$ for $\nu^2(T)$. Conversely, for the smooth process, all spikes need to be re-weighted individually. A memory of all past spikes needs to be kept at all times. Figure 10C shows that this temporal kernel complexity is also a feature of the temporal kernel derived from the covariance function of the empirical rat trajectories in figure 7.

The fundamental difference between the OU and the smooth temporal kernels arises from the difference in the factorisation properties of the prior. As the inverse of the covariance matrix for $\zeta \notin \{0, 1\}$, and specifically for $\zeta = 2$, is dense, it does not factorise over spike combinations and therefore does not allow a recursive form as in equation 12. A recurrence relation as in equation 25

is only possible for the OU prior which factories across duplets of spikes. To see this, write

$$\begin{aligned} p(s_T|\boldsymbol{\xi}_{[0,T)}) &= \int ds_{t_J} p(s_T, s_{t_J}|\boldsymbol{\xi}_{0:T}) \\ &\propto \int ds_{t_J} p(s_T, s_{t_J}) p(s_{t_J}|\xi_{t_J}) \int ds_{0:t_{J-1}} p(s_{0:t_{J-1}}, \boldsymbol{\xi}_{0:T}|s_T, s_{t_J}) \\ &= \int ds_{t_J} p(s_T, s_{t_J}) p(s_{t_J}|\xi_{t_J}) \int ds_{0:t_{J-1}} p(\boldsymbol{\xi}_{0:t_{J-1}}|s_{0:t_{J-1}}) p(s_{0:t_{J-1}}|s_T, s_{t_J}) \\ &= \int ds_{t_J} p(s_T, s_{t_J}) p(s_{t_J}|\xi_{t_J}) m_T(s_T, s_{t_J}) \end{aligned} \tag{13}$$

where t_J stands for the time at which the last spike was observed. The first equality is just an expansion, the proportionality follows from Bayes rule and an expansion and the final equality follows from the assumption of instantaneous spiking. This now looks rather like an update equation. $p(s_T, s_{t_J})$ is a transition probability from the last observed spike to the inference time T, $p(s_{t_J}|\xi_{t_J})$ is the innovation due to the last observation (the likelihood of the last observed spike). Inside the integral however we have, next to the likelihood of all past spikes $p(\boldsymbol{\xi}_{0:t_{J-1}}|s_{0:t_{J-1}})$, also the term $p(s_{0:t_{J-1}}|s_T, s_{t_J})$, which does depend on s_T and means that the spikes are reweighted as a function of s_T: the message $m_T(s_T, s_{t_J})$ is a function of s_T. All spikes have to be used to infer the posterior at each time T. To make the integral independent of s_T, the prior has to be Markov in individual spikes, which is only the case for the OU process:

$$\begin{aligned} p_{\text{OU}}(s_T|\boldsymbol{\xi}_{[0,T)}) &= \int ds_{t_J} p(s_T, s_{t_J}) p(s_{t_J}|\xi_{t_J}) \int ds_{0:t_{J-1}} p(\boldsymbol{\xi}_{0:t_{J-1}}|s_{0:t_{J-1}}) p(s_{0:t_{J-1}}|s_{t_J}) \\ &= \int ds_{t_J} p(s_T, s_{t_J}) p(s_{t_J}|\xi_{t_J}) m_T(s_{t_J}) \end{aligned} \tag{14}$$

Thus, while the time to the last spike simply multiplies the temporal kernel in the OU process, (the message $m_T(s_{t_j})$ is multiplied by the transition probability, see equations 12 and 14 and figure 5, right panel), the smooth temporal kernel changes shape in a complex way (corresponding to the dependence of the message $m_T(s_T, s_{t_j})$ in equation 13 on s_T). Again, this means that all spikes have to be kept in memory for full inference.

3.4 Intermediate (autoregressive) processes

There are intermediate cases between the smooth and the OU process that allow a partially recursive formulation. For illustrative purposes, let us generalise the metronomic OU process to an autoregressive model of n^{th} order we write

$$s_t = \sum_{i=1}^{n} \beta_i s_{t-i\Delta} + c\sqrt{\Delta}\eta_t \tag{15}$$

The set of β_i directly specifies an inverse covariance matrix $\mathcal{C}^{-1}$ (see appendix B), which is $(2n+1)$-diagonal. This implies that the posterior factorises over cliques ψ involving $n+1$ spikes (see equation 11), and that a recursive formulation similar to that in equation 14 is possible. Here, however, the inference will be Markov in *groups of n spikes*. Zhang et al. (1998) find that a 2-step Bayesian decoder, which is an AR(2) process in our terms, aids decoding from hippocampal cell data significantly.

Figure 11A shows samples from such processes of increasing order. The coefficients β where here set such that the n^{th} difference of the process evolved as an OU process (see appendix B). The higher the order, the smoother the processes that can be generated and the more oscillations are apparent in the temporal kernels. The OU and the smooth processes (see section 3.3) are at the opposite end of this spectrum, with tridiagonal and dense inverse matrices respectively.

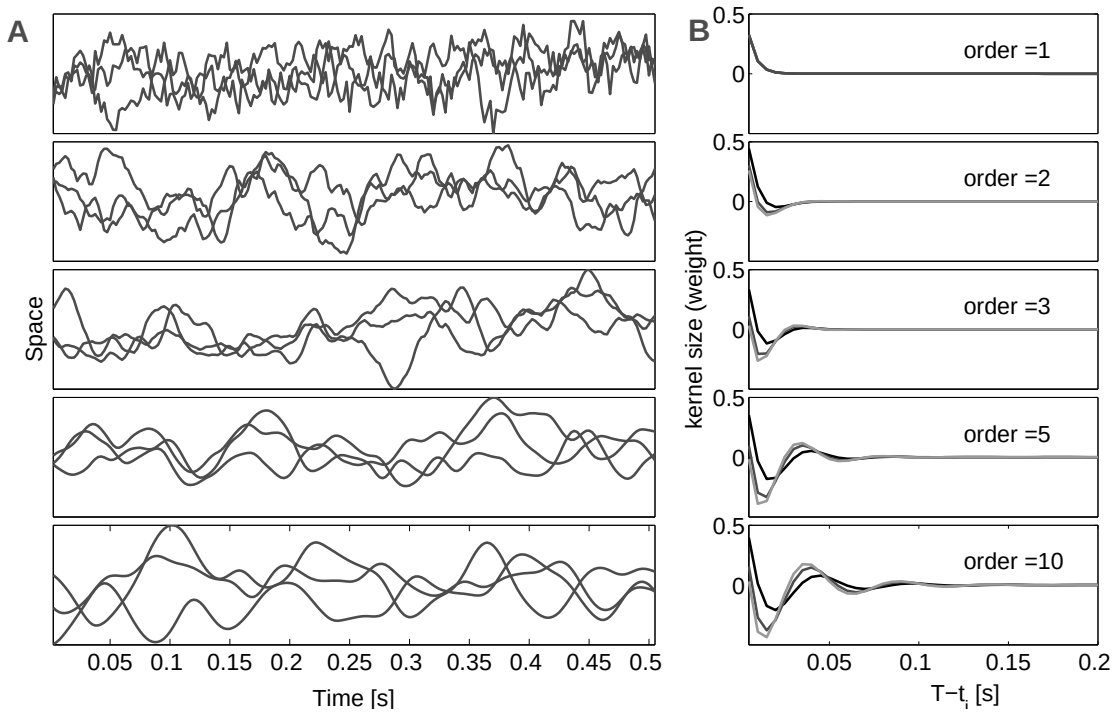

Figure 11: Autoregressive processes of increasing order. **A** Samples from processes of order $n = \{1, 2, 3, 5, 10\}$ from top to bottom. The top process corresponds to an OU process. **B** Metronomic temporal kernels $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ corresponding to the processes in panel A. The different lines correspond to an increasing observation time T as in figures 5 and 10.

The higher the order, the higher the complexity of the code. That is, to decode, it becomes necessary to remember larger numbers of spikes, and also compute or approximate the inverses of larger covariance matrices. While natural stimulus trajectories may not be indefinitely differentiable, the increase in complexity implies that any smoothness has significant potential to render the code complex.

4 Expert spikes for efficient computation

Complex codes, such as that following from the assumption of natural smooth priors, have detrimental effects on the computational availability of information. Efficient computation in time requires access to all encoded information, and thus requires that the complex temporal structure of the code be taken into account. Here, we show that information present in the complex codes can be re-represented by codes that are straightforward to decode and to use in key probabilistic computations.

Specifically, we propose to treat each spike as an independent expert in a product of experts (PoE) setting (Hinton, 1999; Zemel et al., 2005)

$$\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)}) = \frac{1}{Z(T)} \prod_i \exp\left(\sum_t g_i(s,t)\xi^i_{T-t}\right) \tag{16}$$

ie each time a spike ξ^i occurs, it contributes its same *projection kernel* $\exp(g_i(s,t))$ to the posterior distribution $\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)})$ (for each spike, we add the same, stereotyped contribution to the log posterior and then renormalise).

From the discussion in the preceding sections, it is immediately apparent that the PoE approximation is a better approximation for the OU case than for the smooth case. In the following we first derive an approximate analytical expression for separable projection kernels $g_i(s,t) = f_i(s)h(t)$ based on metronomic spikes and the OU prior. We then remove any restrictions and derive non-parametric, non-separable $g_i(s,t)$ for both the OU and the smooth temporal kernel and show that these perform better for the OU process than for the smooth process. Finally we infer a new set of spikes $\boldsymbol{\rho}_{[0,T)}$ such that decoding according to the PoE model produces a posterior distribution $\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)})$ that matches the true posterior distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$ well both for OU and smooth priors.

4.1 Approximate projection kernels

4.1.1 Metronomic projection kernels

We have seen in section 3.2 that the weight given to a spike is approximately a decreasing exponential function of the time elapsed since its occurrence, and that replacing the true temporal kernels by the metronomic temporal kernels (without fixing the time since the last spike at Δ) gives a qualitatively good approximation (bottom panel, figure 4). This suggests writing an approximate distribution with spatiotemporally separable projective kernels

$$\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)}) \propto \prod_i \phi_i(s)^{\sum_t \xi^i_{T-t}\kappa_t} = \prod_i \phi_i(s)^{a_i(T)} \qquad \kappa_t \propto \exp(-\beta t) \tag{17}$$

where $a_i(T)$ is an equivalent "activity" of each neuron. The performance of this approximation is shown in figure 12 for the OU process (see also Zemel et al., 2005). There are a few differences between figure 4B and 12. Keeping the $\phi_i(s)$ as before, the variance of this approximation is $\hat{\nu}^2(T) = \sigma^2/\sum_i a_i(T)$. As the last observed spike recedes into the past this approaches infinity (black arrows in figure 12) and the mean returns to zero (gray dashed arrows in figure 12). This is different in the exact inference, which approaches the static prior with variance $\mathcal{C}_{TT}$. The mean

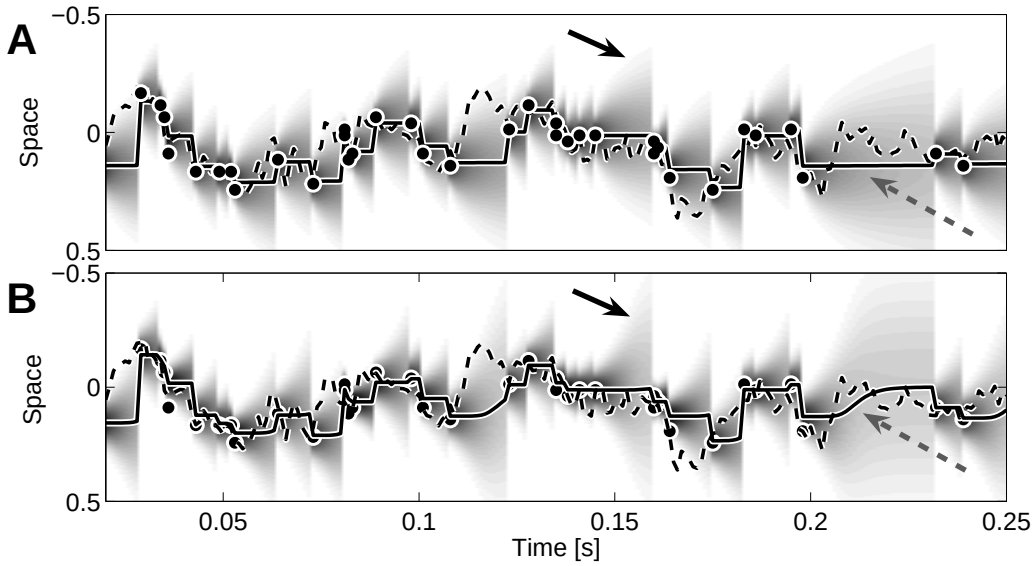

Figure 12: Separable projection kernel for OU process: comparison of true $p(s_T|\boldsymbol{\xi}_{[0,T)})$ (panel **A**) and $\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)})$ from equation 17 (panel **B**). Arrows are explained in the text.

$\hat{\mu}(T) = \sum_i s_i b_i(T)$ where $b_i(T) = a_i(T)/\sum_j a_j(T)$, which is always normalised and returns to zero more slowly. However, $\sum_t k_t(\boldsymbol{\xi}_{[0,T)}, T) < 1$ due to the implicit presence of a spatial prior (some weight is given to the prior, which means that the sum of the weights of the evidence ≤ 1). Because the smooth temporal kernels dip below zero however, this formulation is not applicable to the smooth case.

4.1.2 Inferring full spatiotemporal projection kernels $g_i(s,t)$

To apply expression 16 to the smooth case, we inferred $g_i(s,t)$ in a nonparametric way by discretizing time and space over which the distributions are defined and minimising the Kullback-Leibler divergence between the discretized versions $p(s_T|\boldsymbol{\xi}_{[0,T)})$ and $\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)})$ wrt. the projection kernels

$$g_i(s,t) \leftarrow g_i(s,t) - \varepsilon \nabla_{g_i(s,t)} D_{KL}(p(s_T|\boldsymbol{\xi}_{[0,T)})||\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)})) \tag{18}$$

Given that our approximation 16 is related to restricted Boltzmann machines (RBM), it is not surprising that the gradient has a form akin to the wake-sleep algorithm (Hinton et al., 1995):

$$\nabla_{g_i(s,t)} D_{KL}(p(s_T|\boldsymbol{\xi}_{[0,T)})||\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)})) = \sum_T \left[\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)}) - p(s_T|\boldsymbol{\xi}_{[0,T)})\right] \xi_i(T-t) \tag{19}$$

Figure 13 shows the projection kernels inferred for the OU prior (figure 13A) and the smooth prior (figure 13B). Both fall off as exponentials of time and start out with a spatial profile similar to a difference of Gaussians (DOG). The projection kernels shown are for the same parameter settings as figures 4 and 8, and the faster decay of the smooth projection kernels is due to its shorter correlation timescale. Overall, both projection kernels are approximately separable, indicating that the analytically derived motivation above may be close to optimal and that, in the PoE framework of equation 16, separable projection kernels may be the optimal choice even for the smooth prior. However, simply using these projection kernels to interpret the original spikes $\boldsymbol{\xi}_{[0,T)}$ results in an approximation that is far from perfect, especially in the smooth case: Figure 14 compares the true posterior distribution and that given by the approximation with the

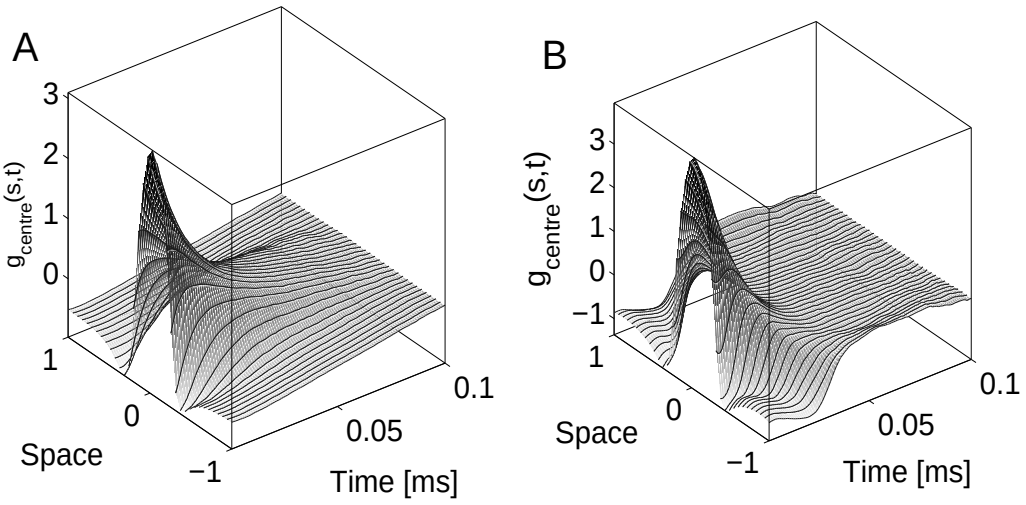

Figure 13: Projection kernels inferred by equation 18 for OU (**A**) and smooth (**B**) priors. Both are initially difference of Gaussians and fall of exponentially with time. There is only slightly more nonseparable structure in the smooth than in the OU case.

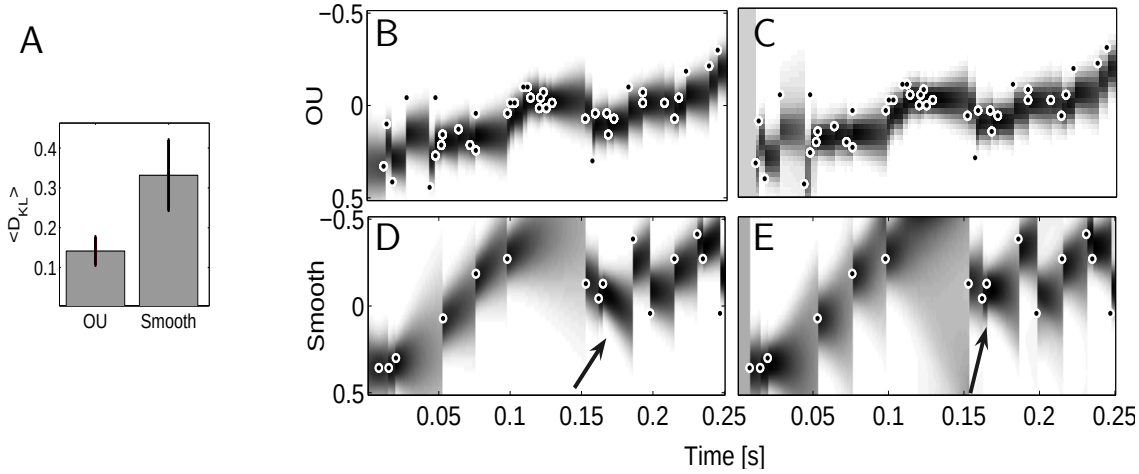

Figure 14: Comparison of true distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$ and approximate distribution $\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)})$ given by equation 16 with projection kernels inferred by equation 18 and shown in figure 13. Organization same as in previous figures. **A** shows $\langle \frac{1}{T}\sum_t D(p(s_T|\boldsymbol{\xi}_{[0,T)})||\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)}))\rangle_{p(s)} \pm 1$ S.D. for both priors. **B,D** show $p(s_T|\boldsymbol{\xi}_{[0,T)})$ and **C,E** the corresponding $\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)})$ for the same spikes. **B,C** are for a stimulus generated from the OU prior and **D,E** for the smooth prior. Arrows: see text.

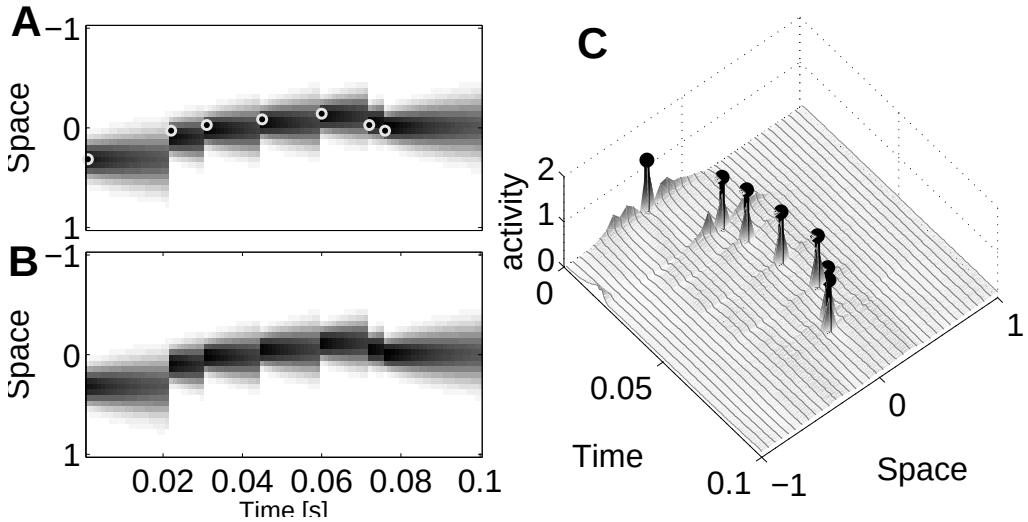

Figure 15: Inferring activities $\mathbf{a}$ for the OU prior. **A** true posterior $p(s_T|\boldsymbol{\xi}_{[0,T)})$ **B** approximate posterior $\hat{p}(s_T|\mathbf{a}_{[0,T)})$, which matches perfectly ($\langle D_{KL}\rangle < 10^{-5}$). **C** activities $\mathbf{a}_{[0,T)}$ for all neurons. The vertical black lines with dots indicate the original spike times $\boldsymbol{\xi}_{[0,T)}$. Each thin line along the gray surface is the "activity" of one neuron as a function of time.

above projection kernels. The cost of independent decoding is quantified in figure 14A using $\langle \frac{1}{T}\sum_t D(p(s_T|\boldsymbol{\xi}_{[0,T)})||\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)}))\rangle_{p(s)}$ and is alrger for the smooth than for the OU process. Visually, there are no gross differences between $p(s_T|\boldsymbol{\xi}_{[0,T)})$ and $\hat{p}(s_T|\boldsymbol{\xi}_{[0,T)})$ for the OU prior (figure 14B and C) but for the smooth prior the arrows in figures 14D and E indicate areas where a large and qualitatively expected mismatch is introduced by the PoE treatment of the spikes.

4.2 Recoding: Finding expert spikes

The previous section has shown that an independent interpretation of spikes is more costly for the smooth than the OU prior. In this section we show that it is possible to find a new set of "expert" spikes $\boldsymbol{\rho}_{[0,T)}$, such that each spike can be interpreted independently and the posterior distribution is matched closely for both the OU and the smooth prior. This recoding thus takes spikes $\boldsymbol{\xi}_{[0,T)}$ that are redundant in a decoding sense and produces a new set of spikes $\boldsymbol{\rho}_{[0,T)}$ that can be easily used for efficient neural computation because the decoding redundancy has been eliminated. We first infer real-valued activities $\mathbf{a}_{[0,T)}$ and then proceed to infer actual spikes $\boldsymbol{\rho}_{[0,T)}$. We here use neurally implausible methods to infer the new set of spikes $\boldsymbol{\rho}$. In a companion paper (Natarajan et al., 2005), we explore the capability of neurally plausible spiking networks to both do this recoding, and to use the resulting simple code for probabilistic computations in time.

4.2.1 Activities

Given a set of projection kernels $g_i(s,t)$ from the previous section, we can go back and infer the optimal activities $\mathbf{a}_{[0,T)} \geq 0$ of neurons by writing

$$\hat{p}(s_T|\mathbf{a}_{[0,T)}) \propto \exp\left(\sum_{i,t} a_i(T-t)g_i(s,t)\right). \tag{20}$$

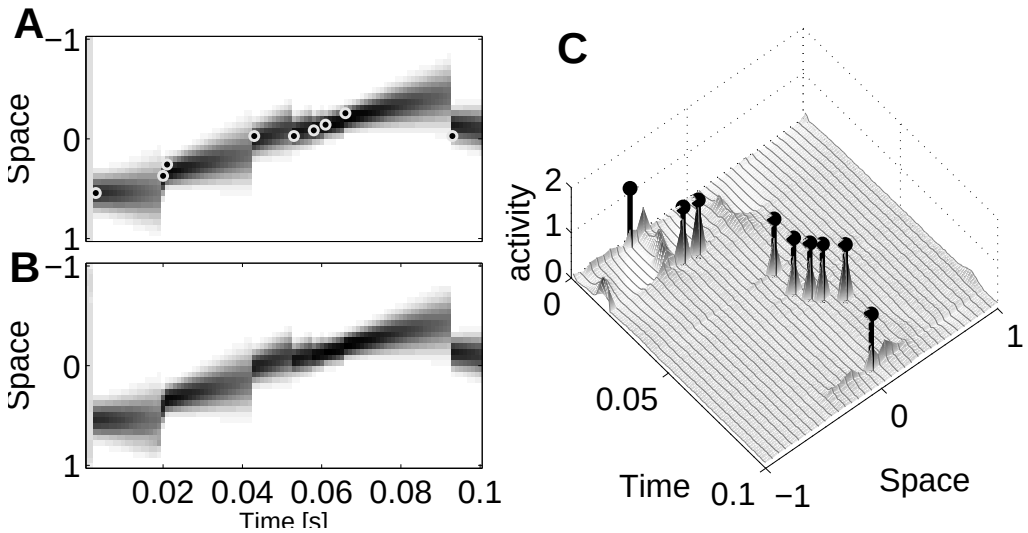

Figure 16: Inferring activities $\mathbf{a}$ for the smooth prior. Same organisation as in figure 15.

If we let $a_i(T-t) = \exp(b_i(T-t))$ and minimise the D_{KL} wrt. $\mathbf{b}_{[0,T)}$, we simultaneously enforce $\mathbf{a}_{[0,T)} \geq 0$:

$$b_i(t) \leftarrow b_i(t) - \varepsilon \nabla_{b_i(t)} D_{KL}(p(s_T|\boldsymbol{\xi}_{[0,T)})||\hat{p}(s_T|\mathbf{a}_{[0,T)})) \tag{21}$$

The results of this procedure are shown for both the OU process (figure 15) and for the smooth process (figure 16). Figure 15A and 16A show the true spikes $\boldsymbol{\xi}_{[0,T)}$ and the corresponding distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$. Figures 15B and 16B show the approximate distributions $\hat{p}(s_T|\mathbf{a}_{[0,T)})$ defined in equation 20 for the optimal activities $\mathbf{a}_{[0,T)}$ inferred with equation 21. They match extremely closely with $\langle D_{KL} \rangle \sim 10^{-5}$. Figures 15C and 16C finally show the inferred activties $\mathbf{a}_{[0,T)}$. Most importantly, we see that the inferred activities (one grey line for each neuron) are very sparse (in time and across neurons), suggesting that there might indeed be a set of (zero-one) spikes that leads to a good approximation via 16. On the other hand, the activities line up closely with the original spikes $\boldsymbol{\xi}_{[0,T)}$ (vertical black lines with dots) and it may be that the approximations with the original spikes in the previous paragraph already gave us the best possible approximation. For the OU prior (top row), the activities in spikeless times are extremely small and zeroing them does not significantly worsen the approximation with $\hat{p}(s_T|\mathbf{a}_{[0,T)})$. However, for the smooth prior (bottom row), there is residual activity between the peaks, the zeroing of which significantly worsens the quality of approximation by $\hat{p}(s_T|\mathbf{a}_{[0,T)})$ (data not shown).

4.2.2 Spikes via simulated annealing

To check whether there exists in fact a set of spikes $\boldsymbol{\rho}_{[0,T)}$ such that decoding according to equation 16 results in a posterior distribution $\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)})$ that matches $p(s_T|\boldsymbol{\xi}_{[0,T)})$ closely for the smooth prior, we assume, as before, the projection kernels $g_i(s,t)$ inferred above and find a set of spikes $\boldsymbol{\rho}_{[0,T)}$ that minimises $D(p(s_T|\boldsymbol{\xi}_{[0,T)})||\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)}))$, where $\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)})$ given by

$$\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)}) = \frac{1}{Z(T)} \prod_i \exp\left(\sum_t g_i(s,t)\rho^i_{T-t}\right) \tag{22}$$

which is the same interpretation we gave the original spikes in equation 16. Our aim is thus to find a new set of spikes that satisfies

$$\boldsymbol{\rho}_{[0,T)} = \arg\min_{\boldsymbol{\rho}_{[0,T)}} D\left(p(s_T|\boldsymbol{\xi}_{[0,T)})||\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)})\right) \tag{23}$$

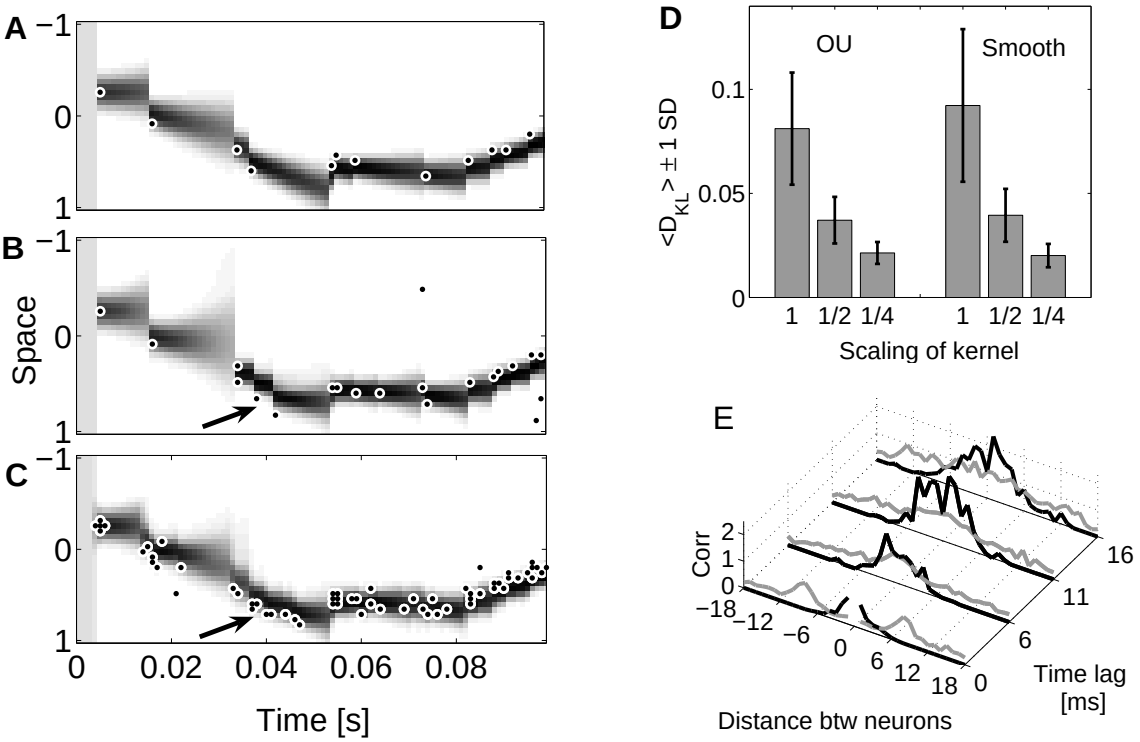

Figure 17: Inferring new spikes for smooth prior. **A** original spikes with $p(s_T|\boldsymbol{\xi}_{[0,T)})$. **B** $\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)})$ with projection kernels $g_i(s,t)$ given by equation 18 **C** $\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)})$ with scaled projection kernels $g_i(s,t)/4$. Note the increased firing rate. Arrows explained in the text. **D** shows the KL-divergence between $p(s_T|\boldsymbol{\xi}_{[0,T)})$ and $\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)})$ for projection kernels scaled by factors of 1, 1/2 and 1/4 respectively. **E** Cross-correlations between neurons for a few different time lags. Black lines: original spikes $\boldsymbol{\xi}$. Grey lines: recoded spikes $\boldsymbol{\rho}$ for unscaled projection kernels. The autocorrelation has been scaled to unity, except at zero lag, where the autocorrelation was excluded and scaling was done wrt. the maximal crosscorrelation.

This is a highly non-convex discrete problem, so we applied standard simulated annealing techniques[2]. Figure 17 shows the results. Figure 17A shows the original spikes distribution $p(s_T|\boldsymbol{\xi}_{[0,T)})$. Figure 17B shows $\hat{p}(s_T|\boldsymbol{\rho}_{[0,T)})$ using the projection kernels shown in figure 13. The arrow in 17B indicates where the new set of spikes performs better than the original, independently interpreted spikes and matches the shifting distribution by adding a new spike. (Remember from figure 14 that simply interpreting the original spikes according to 16 would not allow us to match the downward slope in between spikes.) Overall, the improvement achieved by the new set of spikes $\boldsymbol{\rho}_{[0,T)}$ is much greater for the smooth than the OU process (compare figure 14A and the bars for the unscaled projection kernels in figure 17D). From the close match between $p(s_T|\boldsymbol{\xi}_{[0,T)})$ and $\hat{p}(s_T|\mathbf{a}_{[0,T)})$, we expect the ultimately achievable KL-divergence to dependent strongly on the projective kernels. Figure 17C shows the effect of scaling the inferred projection kernel $g_i(s,t)$. There are more spikes but the match is better. Figure 17D shows average KL-divergencies over 100 sets of new spike trains, for different scalings of the $g_i(s,t)$. In general, the projection kernels found here form an overcomplete basis set. By scaling them down and allowing more spikes, we come closer to the setting in the previous section where we allowed continuous activities rather than 0-1 spikes.

Note the different coding strategy indicated by the arrows, especially in figure 17C: here, spikes are positioned such that they take into account what has already been expressed by previous spikes – spikes are positioned wrt. the already encoded distribution, *ie* there are explicit relations amongst the spikes that are not explained by the stimulus. Figure 17E shows this more clearly. The black traces show the correlations of the original spikes $\boldsymbol{\xi}$, which are purely due to stimulus correlations. The grey lines show the correlations of the recoded spikes $\boldsymbol{\rho}$. At lag 0 (bottom of the figure), flanks appear in the crosscorrelations functions, but at greater lags the crosscorrelations are flatter for the recoded than for the original spikes. Requiring independently decodeable spikes has introduced instantaneous correlations and flattened the spatial profile of crosscorrelations over time.

5 Discussion

In the present work we have analysed the structure of a Bayesian, optimal decoder in a simple, analytically tractable model. The results are a direct generalisation of decoding in the static Gaussian-Poisson encoding model (Snippe and Koenderinck, 1992). We showed that the structure of the decoder depends on the prior over stimulus trajectories in time; that realistic priors render decoding hard (nonlocal in time and space) and that an independent recoding exists in which information is readily available for computational purposes. We are currently working on a biologically plausible network that approximates this recoding and uses the resulting code for flexible probabilistic computations Zemel et al. (2005); Natarajan et al. (2005).

The main innovation in our work is the nature of the informative prior over stimulus *trajectories*. Figure 7 indicates that the exponential prior with $\zeta = 2$ is a good model of natural movements as they tend to be smooth. Smooth trajectories have power spectra that roll off with (temporal) frequency f more like a square exponential $\propto \exp(-f^2)$ than the power law $\propto 1/f^b$ which is a common finding for less structured natural inputs.

Partially as a result of recursive formulations, which do generate power-law spectra, most previous work has assumed rougth priors within the OU class (Brown et al., 1998; Smith and Brown, 2003; Barbieri et al., 2004; Kemere et al., 2004; Gao et al., 2002). However, Zhang et al. (1998) use a 2-step Bayesian decoder corresponding to a second-order autoregressive process (AR(2)) with coefficients that fall off as a squared exponentials (their equation 43). This 2-step decoder performs much better than a 1-step decoder (corresponding to an AR(1) process) on hippocampal

[2]From the very strong sensitivity of our simulated annealing results to the procedure used to reduce the temperature, we infer that the optima are not very well-separated, with a number of similar sets of spike trains doing approximately equally well. We rendered the procedure more global by evaluating, at each step, the decrease in cost that would accompany switching every spike, and accepting one of the best switches probabilistically.

place cell data. In terms of applications, such as brain-machine-interfaces, Kemere et al. (2004) decode from the motor cortex of monkeys making arm movements to one of seven targets. They show that use of a rich, modular prior consisting of separate priors for movements to each of the targets greatly improves decoding. In a similar Bayesian vein, we have shown here that correct treatment of prior temporal structure significantly ameliorates decoding. Figure 14A illustrates the cost of treating all spikes independently. The differences between inference in the smooth and OU case (*eg* the overshoot in figure 8 which is not seen in 4) also indicate qualitatively what information is lost by applying Kalman-filter like formulations to decoding. How large this effect is in quantitative terms depends on the exact specifics of the true model. If spikes are dense (*ie* the likelihood term in equation 1 dominates), the difference may not be large and an approximate prior such as the recursive priors used by Brown et al. (1998); Zhang et al. (1998) may suffice. If spikes are sparse however, the reliance on the prior will be more marked and inference with the wrong prior more costly. Furthermore, the gains from inference with the correct prior have to outweigh the cost of estimating the correct prior. The priors here are presumably empirical priors, inferred from previous experience of the stimulus statistics. While it is sensible to expect nervous systems to acquire detailed and correct informative priors (Körding and Wolpert, 2004; Körding et al., 2004), it remains to be seen whether this is a generally feasible for decoding applications like brain-machine-interfaces.

A number of recent decoding approaches to population coding have looked at Fisher information. Fisher information arises from notions of asymptotic normality where there is a great deal of "data", *ie* long spike counting windows and many neurons. In the asymptotic limit, the posterior distribution is well-approximated by a Gaussian with width $(JI_F)^{-1}$ where J is the number of data points or spikes in our case. This is a linear expansion where each data point (spike) contributes the same amount $1/I_F$ to the variance of the posterior. By contrast, more like Brunel and Nadal (1998); Bethge et al. (2002), we have considered the regime far from the asymptotic limit (albeit with a model which has a permanently Gaussian posterior), where spikes contribute very varied amounts. Remember that spikes contribute varied amounts because they can contribute large amounts. As $J \to \infty$, each spike contributes infinitesimally small amounts. Nevertheless, by analogy with the Fisher information, it is possible to study the dependence of the posterior variance $\nu^2(T)$ on the width of the encoding tuning functions σ^2 and the dimensionality. In our simple model, we find results similar to previously reported ones (Snippe and Koenderinck, 1992; Zhang and Sejnowski, 1999) (data not shown). However, as we are always in the sparse spike limit, only the information per spike is of relevance, and the posterior variance is strictly increasing in σ, the width of the encoding tuning functions, independent of the dimensionality. If there were dense spiking, the population firing rate (Zhang and Sejnowski, 1999; Silberberg et al., 2004; DeCharms and Zador, 2000; Knight, 1972) might carry enough information to overwhelm any prior.

Two assumptions about the encoding model need to be discussed. Firstly, the bell-shaped form of the tuning functions is only very roughly realistic. However, the structure of the code depends on the prior only (see section 3.3), and the shape of the tuning functions does not affect our argument. The most fundamental change would be that the variance of the posterior would depend not only on spike timing but also on which neurons emit the spikes, and the posteriors could also become multimodal in some regimes. Secondly, we assume an instantaneous relationship between the hazard rate of the inhomogeneous Poisson process and the stimulus. There are two aspects to this instantaneity: Dependence only on the current stimulus s_t and independence from the spike history. The first is easily relaxed if the dependence on the stimulus history can be approximated by a linear filter (a discrete sum) as generally done in standard spike-triggered averaging. In that case, the likelihood term 2 is a function of the stimulus at a number of times each of which enters equation 1, *ie* each spike contributes as many entries to the covariance matrix as its linear filter extends in time. However, future work will need to evaluate the impact of history dependence.

Whether brains preferentially utilise independent or more elaborate codes such as correlational ones is a hotly debated topic (Pouget et al., 2003; DeCharms and Zador, 2000). The focus has mostly been on noise correlations (which are absent in the present work as we assume indepen-

dent, inhomogeneous Poisson spiking). Here the correlational nature of the code arises through the temporal correlations in the input signal, *ie* we show that natural, smooth, priors induce codes that have certain computational properties akin to some found in static correlational codes. Foremost amongst these is that decoding becomes hard (nonlocal in time) and may impair an animal's capacity to perform efficient computation (Zohary et al., 1994; Shadlen et al., 1996).

With efficient coding arguments (Barlow, 1961, 1989; Atick, 1992), this suggest that the early sensory system should take the prior into account to produce an *independently interpretable* code. Nirenberg et al. (2001) analyse the cost of neglecting noise correlations present and thus ask whether the code is independently interpretable – a question of interest with respect to stimulus correlations too. As a first step, we have here shown that there exist sets of spikes $\boldsymbol{\rho}_{[0,T)}$ independently interpretable in time that can encode the same distributions as the original spikes $\boldsymbol{\xi}_{[0,T)}$. Once each spike can be interpreted according to equation 16, combining information from *eg* different modalities (as in multisensory integration (Ernst and Banks, 2002; Hillis et al., 2002) or sensorimotor integration (Zemel et al., 2005; Körding and Wolpert, 2004)) becomes straightforward and only requires a multiplication.

We will present neurally plausible implementations of this recoding in a companion paper (Natarajan et al., 2005). This recoding has the aim of producing spike trains that lack *temporal* redundancy and is the dynamic analogue of efficient coding efforts that produce population activities lacking *eg* spatial correlations (Srinivasan et al., 1982; Atick, 1992; Nirenberg et al., 2001). While we here motivate the recoding as a generic manipulation that requires full access to the encoded information and is implied by any computation efficient in the data, producing independently interpretable spikes may arguably be a good objective for an early sensory system. In our companion paper we also show how the recoded spikes allow neurally plausible spiking networks to implement probabilistic computations in a straightforward manner (see also Zemel et al., 2005).

Our most pressing lacuna is that we have considered only a simple form of uncertainty – that arising from sparse and partial observation. In cases such as the aperture problem; (Weiss and Adelson, 1998), ill-posedness leads to a more fundamental form of uncertainty which appears to require that distributions be explicitly encoded. While a number of approaches have successfully addressed these issues in the static case (Anderson, 1994; Barber et al., 2003; Zemel et al., 1998; Sahani and Dayan, 2003), they are still beyond the current dynamic framework.

5.1 Acknowledgements

We thank Sophie Denève, Peter Latham, Máté Lengyel, Liam Paninski and Peggy Seriès for helpful discussions and for reading versions of the manuscript. This work was supported by the Gatsby Charitable Foundation (PD, QH), the EU BIBA consortium (QH, PD), the UCL Medical School MB/PhD program (QH), NSERC and CIHRC NET program (RN, RZ).

A OU process

Replacing each of the ISI's by the average value Δ, we get a Kac-Murdock-Szego Toeplitz matrix for which the analytical inverse is (Dow, 2003):

$$\mathcal{C} = c \begin{bmatrix} 1 & \rho & \rho^2 & \rho^3 \\ \rho & 1 & \rho & \rho^2 \\ \rho^2 & \rho & 1 & \rho \\ \rho^3 & \rho^2 & \rho & 1 \end{bmatrix} \qquad \mathcal{C}^{-1} = c \begin{bmatrix} 1 & -\rho & 0 & 0 \\ -\rho & 1+\rho^2 & -\rho & 0 \\ 0 & -\rho & 1+\rho^2 & -\rho \\ 0 & 0 & -\rho & 1 \end{bmatrix}$$

where $\rho = \exp(-\alpha\Delta)$. rewriting equation 8 as $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T) = \mathcal{C}_{T\tau}\mathcal{C}_{\tau\tau}^{-1}/\sigma^2(\mathcal{C}_{\tau\tau} + \mathbf{I}/\sigma^2)^{-1}$, we note that $\mathcal{C}_{T\tau}\mathcal{C}_{\tau\tau}^{-1} \approx \delta_{i-1}$, *ie* only the first component of this vector is one, all others are zero. The

second factor

$$\mathbf{A}^{-1} = (\mathcal{C} + \mathbf{I}/\sigma^2) = (a-1)\sigma^2 \begin{bmatrix} a & -\rho & 0 & 0 & 0 \\ -\rho & a+\rho^2 & -\rho & 0 & 0 \\ 0 & -\rho & a+\rho^2 & -\rho & 0 \\ 0 & 0 & -\rho & a+\rho^2 & -\rho \\ 0 & 0 & 0 & -\rho & a \end{bmatrix}$$

where $a = \frac{c}{\sigma^2}e^{\alpha\Delta} + 1$. We know $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$. Neglecting the pre-factor for a moment, the first row of $\mathbf{A}$ (which is the one of interest) therefore has to satisfy the following recurrence relation:

$$\begin{aligned} A_{2,1} &= (aA_{1,1} - 1)/\rho & (24) \\ A_{k+2,1} &= (a/\rho + \rho)A_{k+1,1} - A_{k,1} \text{ for } n > 3 & (25) \\ A_{N,1}/A_{N-1,1} &= \rho/a & (26) \end{aligned}$$

Equation 25 is a simple two-term linear recurrence equation and can be solved, and equations 24 and 26 give the boundary conditions. The characteristic equation of equation 25 is

$$r^2 - (a/\rho + \rho)r + 1 = 0 \quad \text{with real roots} \quad \lambda_{1,2} = \frac{1}{2}\left(a/\rho + \rho \pm \sqrt{(a/\rho+\rho)^2 - 4}\right)$$

Including the boundary conditions leads to a solution

$$\begin{aligned} A_{n,1} &= d_1\lambda_1^{n-1} + d_2\lambda_2^{n-1} \\ d_1 &= \left(a - \lambda_1\rho - (a-\lambda_2\rho)\frac{(a\lambda_1 - \rho)}{(a\lambda_2 - \rho)}\left(\frac{\lambda_1}{\lambda_2}\right)^{N-2}\right)^{-1} \\ d_2 &= \frac{1 - d_1(a - \lambda_1\rho)}{a - \lambda_2\rho} \end{aligned}$$

One of the eigenvalues will always be > 1, the other < 1, but both positive. As $\mathcal{C}_{\tau\tau}$ is symmetric, so are $\mathbf{A}^{-1}$ and $\mathbf{A}$, and the first column of $\mathbf{A}$ is equal to its first row which we pick out by premultiplying with $\mathcal{C}_{T\tau}\mathcal{C}_{\tau\tau}^{-1}$. This vector $\mathbf{A}_{1,1:N}$ is exactly the sum of two exponentials we saw when using regular spikes to infer the temporal kernel $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$ and the n^{th} component of $\mathbf{k}(\boldsymbol{\xi}_{[0,T)}, T)$, k_n is given by

$$k_n = [\mathcal{C}_{T\tau}(\mathcal{C}_{\tau\tau} + \mathbf{I}\sigma^2)^{-1}]_n = (a-1)\sigma^2 A_{n,1} = (a-1)\sigma^2(d_1\lambda_1^{n-1} + d_2\lambda_2^{n-1}). \quad (27)$$

If λ_1 is the larger eigenvalue, we see that the corresponding coefficient d_1 will be $\approx (\lambda_2/\lambda_1)^N$ which is very small. The contribution of the larger λ will grow only very slowly and only be seen for the very distant spikes. On the other hand, d_2 will be $\approx 1/(a - \lambda_2\rho)$. For all intents and purposes, the temporal kernel will be decaying exponentially with a negative 'spike time constant' $\log\lambda_2$. Furthermore, if the second boundary condition (for time 0) is moved to $-\infty$, the result is a pure exponential. Both the analytical and numerical kernels are plotted in figure 5.

Relaxing the assumption of metronomic spiking, gives a matrix $\mathbf{A}^{-1}$ which is still tridiagonal, but the elements of which are not equal. Writing matrix $\mathcal{C}$ as

$$\mathcal{C} = \begin{bmatrix} 1 & a & ab & abd \\ a & 1 & b & bd \\ ab & b & 1 & d \\ abd & bd & d & 1 \end{bmatrix} \qquad \mathcal{C}^{-1} = \begin{bmatrix} \frac{1}{1-a^2} & -\frac{a}{1-a^2} & 0 & 0 \\ -\frac{a}{1-a^2} & \frac{1-a^2b^2}{(1-a^2)(1-b^2)} & -\frac{b}{1-b^2} & 0 \\ 0 & -\frac{b}{1-b^2} & \frac{1-b^2d^2}{(1-b^2)(1-d^2)} & -\frac{d}{1-d^2} \\ 0 & 0 & -\frac{d}{1-d^2} & \frac{1}{1-d^2} \end{bmatrix}$$

where $a = ce^{-\alpha|t_1-t_2|}$, $b = ce^{-\alpha|t_2-t_3|}$ etc. This lead to a set of equations similar to 25-26, but including more terms.

B Autoregressive processes of second and higher order

An n^{th} order Gaussian autoregressive sequence of length T as produced by equation 15 can be written as a sample from a multivariate normal distribution in the following way: Let $\mathbf{b} = [1, -\beta_1, -\beta_2, -\beta_3, \cdots, \beta_N]$ and let $\mathcal{B}_t = [\mathbf{0}_t, \boldsymbol{\beta}, \mathbf{0}_{T-n-t}]$, where $\mathbf{0}_t$ stands for a vector of zeros of length t. The inverse covariance matrix of the process is given by

$$\mathcal{C}^{-1} = \sum_{t=0}^{T-n-1} \mathcal{B}_t \mathcal{B}_t^T \tag{28}$$

For the coefficients of $\mathbf{b}$ to define a stationary and finite process, $\mathcal{C}$ must be Toeplitz. One way of generating a finite process from the $\mathbf{b}$ is by letting the n^{th} derivative of the process evolve as an OU process

$$s_t^{(n)} = \beta_0 s_{t-1}^{(n)} + c\sqrt{\Delta}\eta_t \tag{29}$$

in which case the coefficients of the vector $\mathbf{b}$ are given by

$$\boldsymbol{\beta}_i = {}^nC_i(-\beta_0)^{i-1} \tag{30}$$

where ${}^nC_i$ is the binomial coefficient. To enforce stationarity, we have to finally perform a subtraction:

$$\mathcal{C}^{-1} = \left(\sum_{t=0}^{T-1} \mathcal{B}_t \mathcal{B}_t^T\right) - \sum_{t'=T-n}^{T} \mathcal{B}_t^{-1} \mathcal{B}_t^{-T} \tag{31}$$

where we abuse notation and $\mathcal{B}^{-1}$ stands for $\mathcal{B}_t^{-1} = [\mathbf{0}_t, \beta_N, \beta_{N-1}, \cdots, \beta_1, \mathbf{0}_{T-n-t}]$

References

Anderson, C. H. (1994). Basic elements of biological computational systems. *Int. J. Modern Physics C*, 5(2):135–7.

Atick, J. J. (1992). Could information theory provide an ecological theory of sensory processing? *Network*, 3:213–51.

Barber, M. J., Clark, J. W., and Anderson, C. H. (2003). Neural representation of probabilistic information. *Neural Comp.*, 15:1843–64.

Barbieri, R., Frank, L. M., Nguyen, D. P., Quirk, M. C., Solo, V., Wilson, M. A., and Brown, E. N. (2004). Dynamic analyses of information encoding in neural ensembles. *Neural Comp.*, 16:277–307.

Barlow, H. B. (1953). Summation and inhibition in the frog's retina. *J. Physiol.*, 137:69–88.

Barlow, H. B. (1961). Possible principles underlying the transformation of sensory messages. In Rosenblith, W., editor, *Sensory Communicaton*. MIT Press, Cambride, MA.

Barlow, H. B. (1989). Unsupervised learning. *Neural Comput.*, 1:295–311.

Bethge, M., Rotermund, D., and Pawelzik, K. (2002). Optimal short-term population coding: when Fisher information fails. *Neural Comp.*, 14(2):303–319.

Bialek, W., Rieke, F., de Ruyter van Steveninck, R. R., and Warland, D. (1991). Reading a neural code. *Science*, 252(5014):1854–7.

Brown, E. N., Frank, L. M., Tang, D., Quirk, M. C., and Wilson, M. A. (1998). A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *J. Neurosci.*, 18(18):7411–25.

Brunel, N. and Nadal, J.-P. (1998). Mutual information, Fisher information, and population coding. *Neural Comp.*, 10:1731–57.

DeCharms, C. and Zador, A. (2000). Neural representation and the cortical code. *Annu. Rev. Neurosci.*, 23:613–647.

Deneve, S., Latham, P. E., and Pouget, A. (2001). Efficient computation and cue integration with noisy population codes. *Nat. Neurosci.*, 4(8):826–31.

Dow, M. (2003). Explicit inverses of Toeplitz and associated matrices. *Austr. New Zealand Industr. Appl. Math. J. E (ANZIAMJ E)*, 44:E185–E215.

Ernst, M. O. and Banks, M. S. (2002). Humans integrate visual and haptic information in a statistically optimal fashion. *Nature*, 415(6870):429–33.

Gao, Y., Black, M. J., Bienenstock, E., Shoham, S., and Donoghue, J. P. (2002). Probabilistic inference of hand motion from neural activity in motor cortex. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems (NIPS) 14*. MIT Press, Cambridge, MA.

Georgopoulos, A. P., Schwartz, A. B., and Kettner, R. E. (1983). Neuronal population coding of movement direction. *Science*, 233(4771):1416–9.

Hillis, J. M., Ernst, M. O., Banks, M. S., and Landy, M. S. (2002). Combining sensory information: mandatory fusion within, but not between, senses. *Science*, 298(5598):1627–30.

Hinton, G. E. (1999). Products of experts. In *Ninth International Conference on Artificial Neural Networks, (ICANN 9)*, volume 1, pages 1–6. IEEE Conf. publ.

Hinton, G. E., Dayan, P., Frey, B. J., and Neal, R. M. (1995). The wake-sleep algorithm for unsupervised neural networks. *Science*, 268(5214):1158–61.

Johansson, R. S. and Birznieks, I. (2004). First spikes in ensembles of human tactile afferents code complex spatial fingertip events. *Nat. Neurosci.*, 7:170–7.

Kemere, C., Santhaman, G., Yu, B. M., Ryu, S., Meng, T., and Shenoy, K. V. (2004). Model-based decoding of reaching movements for prosthetic systems. In *Proceedings of the 26th Annual International conference of the IEEE EMBS*, pages 4524–4528. IEEE EMBS, IEEE.

Knight, B. W. (1972). Dynamics of encoding in a population of neurons. *J. Gen. Physiol.*, 59:734–766.

Körding, K. and Wolpert, D. M. (2004). Bayesian integration in sensorimotor learning. *Nature*, 427(6971):244–7.

Körding, K. P., Ku, S. P., and Wolpert, D. M. (2004). Bayesian integration in force estimation. *J Neurophysiol*, 92(5):3161–5.

Lever, C., Wills, T., Cacucci, F., Burgess, N., and O'Keefe, J. (2002). Long-term plasticity in the hippocampal place cell representation of environmental geometry. *Nature*, 426:90–94.

MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge University Press, Cambridge, UK.

Natarajan, R., Huys, Q. J., Dayan, P., and Zemel, R. S. (2005). Online learning and inference in spiking populations. In preparation.

Nirenberg, S., Carcieri, S. M., Jacobs, A. L., and Latham, P. E. (2001). Retinal ganglion cells act largely as independent encoders. *Nature*, 411:698–701.

Paradiso, M. A. (1988). A theory for the use of visual orientation information which exploits the columnar structure of striate cortex. *Biol. Cybern.*, 58(1):35–49.

Pouget, A., Dayan, P., and Zemel, R. S. (2003). Inference and computation with population codes. *Annu. Rev. Neurosci.*, 26:318–410.

Pouget, A., Zhang, K., Deneve, S., and Latham, P. E. (1998). Statistically efficient estimation using population coding. *Neural Comp.*, 10(2):373–401.

Rao, R. P. N., Olshausen, B. A., and Lewicki, M. S., editors (2002). *Probabilistic Models of the Brain: Perception and Neural Function*. MIT Press, Cambridge, MA, USA.

Reinagel, P. and Reid, R. C. (2000). Temporal coding of visual information in the thalamus. *J. Neurosci.*, 20(14):5392–5400.

Rieke, F., Warland, D., de Ruyter van Steveninck, R., and Bialek, W. (1997). *Spikes. Exploring the neural code*. MIT Press, Cambridge, MA.

Sahani, M. and Dayan, P. (2003). Doubly distributional population codes: simultaneous representation of uncertainty and multiplicity. *Neural Comp.*, 15:2255–79.

Schwartz, A. B. (1994). Direct cortical representation of drawing. *Science*, 265:540–3.

Seriès, P., Latham, P. E., and Pouget, A. (2005). Tuning curve sharpening for orientation selectivity: coding efficiency and the impact of correlations. *Nat. Neurosci.*, 7(10):1129–35.

Seung, S. H. and Sompolinsky, H. (1993). Simple models for reading neuronal population codes. *Proc. Natl. Acad. Sci. USA*, 90(22):10749–53.

Shadlen, M. N., Britten, K., Newsome, W. T., and Movshon, T. (1996). A computational analysis of the relationship between neuronal and behavioural responses to visual motion. *J. Neurosci.*, 16:1486–510.

Shamir, M. and Sompolinsky, H. (2004). Nonlinear population codes. *Neural Comput.*, 16(6):1105–36.

Silberberg, G., Bethge, M., Markram, H., Pawelzik, K., and Tsodyks, M. (2004). Dynamics of population rate codes in ensembles of neocortical neurons. *J Neurophysiol*, 91(2):704–9.

Smith, A. C. and Brown, E. N. (2003). Estimating a state-space model from point process observations. *Neural Comp.*, 15:965–991.

Snippe, H. P. and Koenderinck, J. J. (1992). Discrimination thresholds for channel-coded systems. *Bio. Cybern.*, 66:543–551.

Srinivasan, M. V., Laughlin, S. B., and Dubs, A. (1982). Predictive coding: a fresh view of inhibition in the retina. *Proc R Soc Lond B Biol Sci.*, 216(1253):427–59.

Twum-Danso, N. and Brockett, R. (2001). Trajectory estimation from place cell data. *Neural Networks*, 14:835–44.

Van Rullen, R. and Thorpe, S. J. (2001). Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex. *Neural Comp.*, 13(6):1255–83.

Weiss, Y. and Adelson, E. H. (1998). Slow and smooth: a Bayesian theory for the combination of local motion signals in human vision. A.I.Memo 1624, Massachusetts Institute of Technology, Artificial Intelligence Laboratory.

Wilson, M. A. and McNaughton, B. L. (1993). Dynamics of the hippocampal ensemble code for space. *Science*, 261(5124):1055–8.

Zemel, R. S., Dayan, P., and Pouget, A. (1998). Probabilistic interpretation of population codes. *Neural Comp.*, 10(2):403–430.

Zemel, R. S., Huys, Q. J. M., Natarajan, R., and Dayan, P. (2005). Probabilistic computation in spiking populations. In Saul, L. K., Weiss, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems (NIPS) 17*, pages 1609–1616. MIT Press, Cambridge, MA.

Zhang, K., Ginzburg, I., McNaughton, B. L., and Sejnowski, T. J. (1998). Interpreting neuronal population activity by reconstruction: unified framework with application to hippocampal place cells. *J. Neurophysiol.*, 79:1017–1044.

Zhang, K. and Sejnowski, T. J. (1999). Neuronal tuning: to sharpen or to broaden. *Neural Comp.*, 11(1):75–84.

Zohary, E., Shadlen, M. N., and Newsome, W. T. (1994). Correlated neuronal discharge rate and its implications for psychophysical performance. *Nature*, 370:140–43.