Decoding spikes in a spiking neuronal network

2004 J. Phys. A: Math. Gen. 37 5713

(http://iopscience.iop.org/0305-4470/37/22/001)

INSTITUTE OF PHYSICS PUBLISHING JOURNAL OF PHYSICS A: MATHEMATICAL AND GENERAL

J. Phys. A: Math. Gen. **37** (2004) 5713–5727 PII: S0305-4470(04)72724-9

Decoding spikes in a spiking neuronal network

Jianfeng Feng[1] **and Mingzhou Ding**[2]

[1] Department of Informatics, University of Sussex, Brighton BN1 9QH, UK
[2] Department of Mathematics, Florida Atlantic University, Boca Raton, FL 33431, USA

Received 27 November 2003
Published 18 May 2004
Online at stacks.iop.org/JPhysA/37/5713
DOI: 10.1088/0305-4470/37/22/001

Abstract

We investigate how to reliably decode the input information from the output of a spiking neuronal network. A maximum likelihood estimator of the input signal, together with its Fisher information, is rigorously calculated. The advantage of the maximum likelihood estimation over the 'brute-force rate coding' estimate is clearly demonstrated. It is pointed out that the ergodic assumption in neuroscience, i.e. a temporal average is equivalent to an ensemble average, is in general not true. Averaging over an ensemble of neurons usually gives a biased estimate of the input information. A method on how to compensate for the bias is proposed. Reconstruction of dynamical input signals with a group of spiking neurons is extensively studied and our results show that less than a spike is sufficient to accurately decode dynamical inputs.

PACS numbers: 89.70.+c, 87.19.La

(Some figures in this article are in colour only in the electronic version)

1. Introduction

In a spiking neuronal network, how to reliably decode its input information in terms of observed neuronal output activity? This is a long-standing and fundamental issue in (computational) neuroscience [11, 5]. Even in the simplest form of neuronal models, the integrate-and-fire model, the answer is not known [16]. The difficulty lies in the fact that we do not know the exact input and output relationship in any 'realistic' neuron (model) and hence a parametric estimate approach is hard to be implemented in practice. In the current paper, we tackle the issue in a network of integrate-and-fire models. We expect that our approach will open a pathway into the study of how the biological nervous system works.

To implement a decoding scheme for a spiking network, we first need the exact relationship between the input and the output of the integrate-and-fire model. To this end, we begin by considering spiking models with exactly balanced inputs. By exactly balanced input we mean that the mean input equals the threshold. In fact, such a model has been extensively studied

in the literature, see for example [19]. The output interspike interval distribution is then rigorously calculated. With the help of such a distribution, the maximum likelihood estimator is explicitly constructed. Surprisingly, the obtained maximum likelihood estimate of the input rate is quite simple and we can further manage to calculate the related Fisher information as well. With a static (constant) input, the accuracy of the maximum likelihood estimate is explored and we assert that the input information can be reliably decoded. In [2], the author has carried out a similar approach with experimental data. Unfortunately, in [2] an exact distribution density of the interspike intervals is not available.

We then go a step further to decode dynamical input signals. Here, we find that the often made assumption of ergodicity (averaging over a lone time window of a single neuron activity is equivalent to averaging over a large group of neurons' activity), does not hold true. The ensemble average usually overestimates the input. A method to adjust for the overestimate is then proposed. We examine how long a time window of a population of neurons is needed to carry out a reliable decoding. The issue is related to the currently hotly debated 'coding problem': whether the nervous system employs a 'time coding' strategy or a 'rate coding' strategy. The usual argument against the 'rate coding' strategy is with respect to its information processing speed; it is too slow to reconstruct the input information from the rate function (see for example [14]). In the population coding framework as we developed below, a neuron only needs a single spike to read out the input information. How many spikes are actually needed to decode the input information in the 'rate coding' framework? We show that in a population of spiking neurons (100 neurons are used in our simulations), we can reliably decode a dynamical input signal even with less than a single spike per neuron. Our results clearly demonstrate that if time is the critical requirement of information processing in the nervous system, the 'rate coding' is faster than the 'time coding'.

We emphasize here that due to technical difficulties (see section 3), we are only able to work out our results under the assumption of exactly balanced inputs. Although we believe that all our main results below are qualitatively true for general cases, i.e., without the assumption of exactly balanced inputs, it remains an interesting question to exactly find the maximum likelihood estimate of the input. A movie showing our results with dynamical input signals can be downloaded at http://www.cogs.susx.ac.uk/users/jianfeng.

2. Models

The model neuron we use here is the classical integrate-and-fire model [15]. When the membrane potential $V(t)$ is below the threshold V_{thre}, its evolution is determined by

$$\mathrm{d}V(t) = -\left[\frac{V(t) - V_{\text{rest}}}{\gamma}\right]\mathrm{d}t + \mathrm{d}I_{\text{syn}}(t) \qquad t > 0 \tag{1}$$

with $V(0) = V_{\text{rest}} < V_{\text{thre}}$ and where γ is the decay time constant. The synaptic input current is

$$I_{\text{syn}}(t) = a\sum_{i=1}^{p} E_i(t) - b\sum_{j=1}^{q} I_j(t)$$

with $E_i = \{E_i(t), t \geqslant 0\}$, $I_j = \{I_j(t), t \geqslant 0\}$ as nonhomogeneous Poisson processes with rates $\lambda_{E,i}(t)$ and $\lambda_{I,j}(t)$, respectively [13], $a > 0, b > 0$ being the magnitudes of each EPSP and IPSP and p and q are the total number of active excitatory and inhibitory synapses. Once $V(t)$ crosses V_{thre} from below a spike is generated and V is reset to V_{rest}, the resting potential. This model is termed the leaky integrate and fire model. The interspike interval of efferent

spikes is the random variable

$$T = \inf\{t : V(t) \geqslant V_{\text{thre}} | V(0) = V_{\text{rest}}\}.$$

We use $p_\lambda(t)$ to denote the distribution density of T. More precisely we should define

$$\begin{aligned} &\tau_i = \inf\{t > \tau_{i-1} : V(t) \geqslant V_{\text{thre}} | V(\tau_{i-1}) = V_{\text{rest}}\} \qquad i \geqslant 1 \\ &\tau_0 = 0 \end{aligned} \tag{2}$$

and $T_i = \tau_i - \tau_{i-1}$ for $i \geqslant 1$. It is readily seen that $\{T_i, i \geqslant 1\}$ is an i.i.d. sequence and has a common distribution density as T.

In the following, we further assume that $V_{\text{rest}} = 0, a = b, p = q$ and use diffusion approximations to approximate synaptic inputs [15]. However, the assumption of equal strengths for the excitatory and inhibitory synaptic drives may be easily relaxed. Thus, we put

$$\mathrm{d}I_{\text{syn}}(t) = \mu(t)\,\mathrm{d}t + \sigma(t)\,\mathrm{d}B(t)$$

where $B = \{B(t), t \geqslant 0\}$ is the standard (mean zero, variance t at time t) Brownian motion, and

$$\begin{cases} \sigma^2(t) = a^2\lambda(t)(1+r) \\ \mu(t) = a\lambda(t)(1-r) \end{cases} \tag{3}$$

with $\lambda(t) = \sum \lambda_{E,i}(t)$ and $r\lambda(t) = \sum \lambda_{I,i}(t)$, r being the ratio between the inhibitory input and excitatory input. In general r could be a function of the input λ, i.e. $r(t) = r(\lambda(t))$.

In the current paper, we focus on the case of *exactly balanced inputs*, which means $\mu(t)\gamma = V_{\text{thre}}$, i.e.

$$r(t) = 1 - \frac{V_{\text{thre}}}{a\lambda(t)\gamma}. \tag{4}$$

Biologically this condition says: inhibitory inputs act as a feedback to maintain the balanced inputs. The relationship between r and λ roughly describes the well-known 'push–pull' effect of the inhibitory input: the stronger the input (larger λ), the stronger the inhibitory input (larger r). Hence,

$$\sigma^2(t) = 2a^2\lambda(t) - \frac{aV_{\text{thre}}}{\gamma}. \tag{5}$$

We fix a few parameters in the simulations: $\gamma = 20$ ms, $a = 0.5$ mV and $V_{\text{thre}} = 20$ mV, these values being appropriate for some pyramidal neurons in the mammalian neocortex. We consider input rates within the regions of 2–10 kHz, which is roughly equivalent to 300 neurons firing with a rate between 6 and 30 Hz.

3. Maximum likelihood estimate

We first consider static inputs. Equation (3) in the previous section is now reduced to

$$\begin{cases} \sigma^2 = a^2\lambda(1+r) \\ \mu = a\lambda(1-r) \end{cases} \tag{6}$$

with

$$r = 1 - \frac{V_{\text{thre}}}{a\lambda\gamma}$$

and hence

$$\sigma^2 = 2a^2\lambda - \frac{aV_{\text{thre}}}{\gamma}.$$

We have the following lemma on the probability distribution of $p_\lambda(t)$.

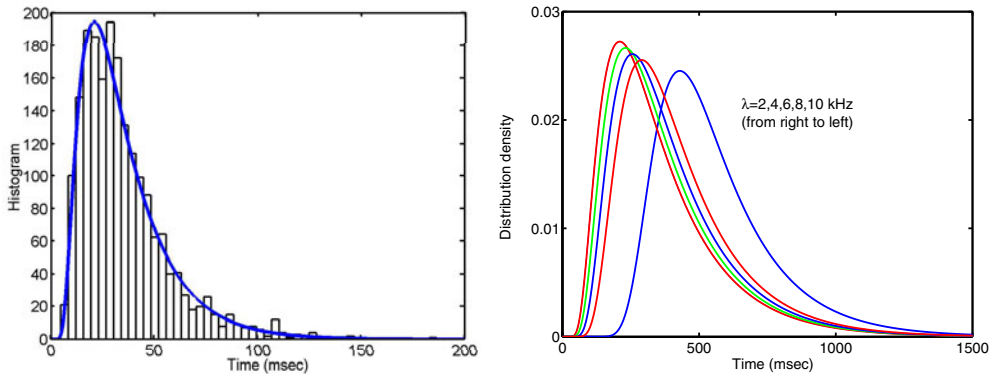

Figure 1. Left: histogram versus time and $p_\lambda 194/0.027$ (solid, smooth line) versus time. Histogram is obtained with $a = 0.5, \lambda = 10\,\text{kHz}, \gamma = 20$ ms and $r = 0.8$. Two thousand ISIs are generated to produce the histogram. Right: $p_\lambda(t)$ versus t with $\lambda = 2, 4, 6, 8$ and 10 kHz.

Lemma 1. *When $V_{\text{thre}} = \gamma\mu$, i.e. the exactly balanced input case, we have*

$$p_\lambda(t) = \frac{2\sigma^2 V_{\text{thre}} \exp(-t/\gamma)}{\sqrt{\pi[\sigma^2\gamma(1-\exp(-2t/\gamma))]^3}} \exp\left\{-\frac{(V_{\text{thre}})^2 \exp(-2t/\gamma)}{\sigma^2\gamma(1-\exp(-2t/\gamma))}\right\}. \quad (7)$$

Proof. From equation (9.222) in [15], we see that the Laplace transformation of $p_\lambda(t)$ is given by

$$p_{\text{L},\lambda}(s) = \frac{\exp\left[\left(\frac{\gamma\mu}{\sigma\sqrt{2}}\right)^2\right] D_{-s}\left(\frac{\gamma\mu}{\sigma/\sqrt{2}}\right)}{\exp\left[\left(\frac{\gamma\mu - V_{\text{thre}}}{\sigma\sqrt{2}}\right)^2\right] D_{-s}\left(\frac{\gamma\mu - V_{\text{thre}}}{\sigma/\sqrt{2}}\right)} \quad (8)$$

where

$$D_{-s}(x) = \frac{\exp(-x^2/4)}{2^{s/2}\sqrt{\pi}}\left[\cos\left(\frac{\pi s}{2}\right)\Gamma\left(\frac{1}{2}-\frac{s}{2}\right)\Phi\left(\frac{s}{2},\frac{1}{2};\frac{x^2}{2}\right) - 2^{1/2}\sin\left(\frac{\pi s}{2}\right)\Gamma\left(1-\frac{s}{2}\right)x\Phi\left(\frac{s}{2}+\frac{1}{2},\frac{3}{2};\frac{x^2}{2}\right)\right]$$

and Φ is the confluent hypergeometric function of the first kind, i.e.

$$\Phi(\xi,\eta;z) = 1 + \frac{\xi z}{\eta} + \frac{\xi(\xi+1)}{\eta(\eta+1)}\frac{z^2}{2!} + \cdots + \frac{\Gamma(\xi+n)}{\Gamma(\eta+n)}\frac{\Gamma(\eta)}{\Gamma(\xi)}\frac{z^n}{n!} + \cdots.$$

Up to now, it seems the inverse Laplace of equation (8) remains open and it is hard to obtain a close form of p_λ [7, 15]. However, under the assumption $\gamma\mu = V_{\text{thre}}$, equation (8) becomes

$$p_{\text{L},\lambda}(s) = \frac{\exp\left[\left(\frac{\gamma\mu}{\sigma\sqrt{2}}\right)^2\right] D_{-s}\left(\frac{\gamma\mu}{\sigma/\sqrt{2}}\right)}{\frac{1}{2^{s/2}\sqrt{\pi}}\cos\left(\frac{\pi s}{2}\right)\Gamma\left(\frac{1}{2}-\frac{s}{2}\right)}. \quad (9)$$

From equation (9), we can perform the inverse Laplace transformation, and the conclusion of the current lemma thus follows. □

In figure 1, the distribution density p_λ of equation (7) and numerically simulated distribution density are plotted. It is easily seen that the distribution density moves to the left (figure 1, right) when the input becomes stronger.

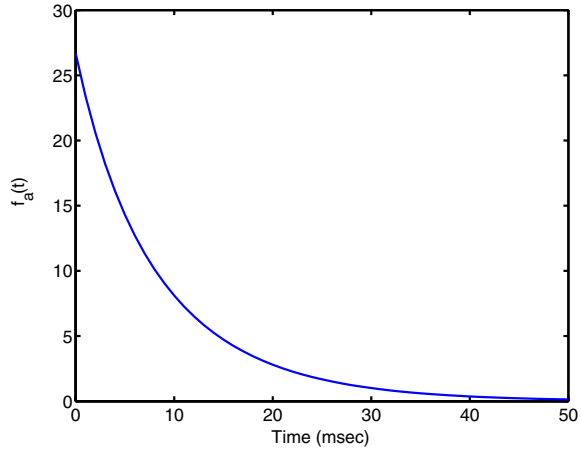

Figure 2. $f_a(t)$ versus time t. It is interesting to see that shorter intervals play a much important role in the estimation of $\hat{\lambda}_N$.

Now we turn our attention to the maximum likelihood estimate of the input. Let us denote

$$f_1(t) = \frac{2V_{\text{thre}} \exp(-t/\gamma)}{\sqrt{\pi[\gamma(1-\exp(-2t/\gamma))]^3}}$$

and

$$f_2(t) = \frac{(V_{\text{thre}})^2 \exp(-2t/\gamma)}{\gamma(1-\exp(-2t/\gamma))}.$$

We then have

$$p_\lambda(t) = \frac{f_1(t)}{\sigma} \exp\left(-\frac{f_2(t)}{\sigma^2}\right).$$

Therefore, the likelihood function is given by

$$L = \frac{\prod_{i=1}^N f_1(t_i)}{\sigma^N} \exp\left(-\frac{\sum_{i=1}^N f_2(t_i)}{\sigma^2}\right)$$

where N is the total number of spikes. It is easily seen that the maximum likelihood estimate of σ is

$$\hat{\sigma}_N^2 = \frac{2\sum_{i=1}^N f_2(t_i)}{N} = \sum_{i=1}^N \frac{2(V_{\text{thre}})^2 \exp(-2t_i/\gamma)}{N\gamma(1-\exp(-2t_i/\gamma))}. \tag{10}$$

We arrive at the final conclusion

$$\begin{aligned}\hat{\lambda}_N &= \sum_{i=1}^N \frac{(V_{\text{thre}})^2 \exp(-2t_i/\gamma)}{Na^2\gamma(1-\exp(-2t_i/\gamma))} + \frac{V_{\text{thre}}}{2a\gamma} \\ &= \frac{\sum_{i=1}^N f_a(t_i)}{N} + \frac{V_{\text{thre}}}{2a\gamma}\end{aligned} \tag{11}$$

where $f_a(t) = f_2(t)/a^2$ (see figure 2). Hence, we have

Theorem 1. *For the IF model with exactly balanced inputs, the maximum likelihood estimate of the input rate is given by equation (11).*

We call $f_a(t)$ the MLE function. It is interesting to note that the MLE function is a decreasing function. When t is small, its value is larger. In other words, the small t will play a much important role in the maximum likelihood estimate. Via f_a, different interspike interval lengths are weighted very differently. The functional meaning of the MLE function is clear. It helps the nervous systems improve the speed of decoding: ignoring long interspike intervals without affecting too much the decoding accuracy.

4. Fisher information and decoding of static inputs

In statistics, the Fisher information is of fundamental importance. Via the Cramer–Rao inequality, it sets the lowest bound for an unbiased estimate. Furthermore, it gives the confidence interval of the MLE when the sampling size is larger enough (asymptotically).

According to the definition of the Fisher information $I(\lambda)$, we have

$$I(\lambda) = \int \left(\frac{p'_\lambda(t)}{p_\lambda(t)}\right)^2 p_\lambda(t)\,\mathrm{d}t. \tag{12}$$

A simple calculation tells us that

$$p'_\lambda(t) = \left(-\frac{a^2}{\sigma^2} + \frac{2a^2 f_2(t)}{\sigma^3}\right) p_\lambda(t).$$

Hence,

$$\begin{aligned} I(\lambda) &= \frac{4a^4\langle f_2^2\rangle}{\sigma^6} - \frac{4a^4\langle f_2\rangle}{\sigma^5} + \frac{a^4}{\sigma^4} \\ &= \frac{4a^6\langle f_a^2\rangle}{\sigma^6} - \frac{4a^5\langle f_a\rangle}{\sigma^5} + \frac{a^4}{\sigma^4} \end{aligned}$$

which in turn yields the following conclusion.

For an estimate $\hat{\lambda}_N$ of a parameter λ, let us assume that $\sqrt{N}(\lambda - \hat{\lambda}_N) \to \mathrm{Nor}(0, \sigma^2)$ in distribution where $\mathrm{Nor}(\cdot,\cdot)$ is the normal distribution. We call $(\lambda - \sigma/\sqrt{N}, \lambda + \sigma/\sqrt{N})$ the $(\sigma-)$ confidence interval of the estimate $\hat{\lambda}_N$.

Theorem 2. *For a given N, the confidence interval of the maximum likelihood estimate of the input λ is given by*

$$\left[\lambda - \frac{1}{\sqrt{NI(\lambda)}}, \lambda + \frac{1}{\sqrt{NI(\lambda)}}\right]. \tag{13}$$

Proof. It is easy to check that all conditions in theorem 3.10 on p 449 of [10] are true and so we have

$$\sqrt{N}(\hat{\lambda}_N - \lambda) \to \mathrm{Nor}(0, 1/I(\lambda))$$

in distribution where $\mathrm{Nor}(\cdot,\cdot)$ is the normal distribution with corresponding mean and variance. Hence, the conclusions in theorem 2 follow. □

In figure 3, the coefficient of variation (CV) of the interspike intervals (ISIs) and Fisher information with respect to λ are plotted. The property of the obtained Fisher information and CV is interesting. We note that the higher the input rate, the more variable the output spikes (the larger the CV or the smaller the Fisher information). As we all know (see theorem 2), the lower the Fisher information, the less accurate the estimate of the input rate.

Another interesting question we might ask is why we intend to develop the MLE. There are many possible ways to estimate (decode) the input firing rate. The simplest way is to

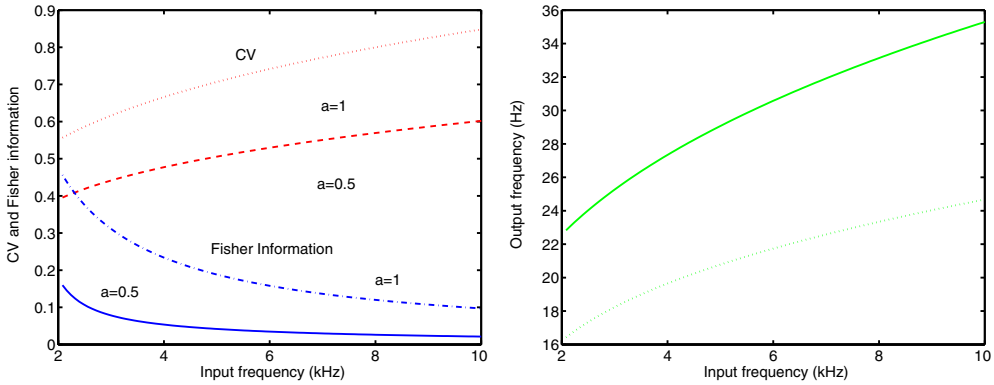

Figure 3. Left: CV (dotted and dashed lines) and Fisher information (solid and dot-dashed lines) versus input frequency. Right: output frequency versus input frequency. A refractory period of 5 ms is added to calculate the output frequency. $a = 1$ mV (solid lines) and $a = 0.5$ mV (dotted lines).

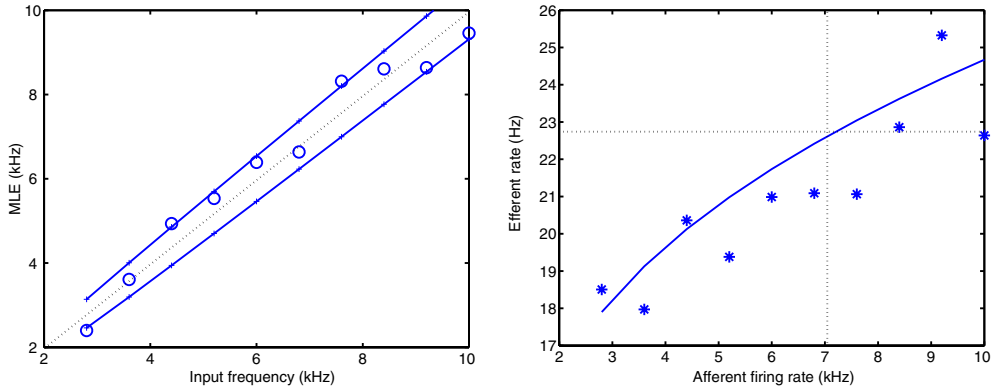

Figure 4. Left: MLE versus input frequency (kHz) with $N = 100$. Solid lines are confidence intervals. Right: theoretical efferent firing rate (solid line) and simulated efferent firing rate ($*$) versus afferent inputs.

determine the input rates via moments of ISIs and the approach has been extensively discussed in the literature. The term 'rate coding' is usually used to refer the approach. In figure 4(right), we also plot theoretical efferent firing rates versus afferent firing rates and simulated efferent firing rates versus afferent firing rates. In terms of figure 4(right), we (the nervous system) can read out the input rate. However, a direct comparison of figure 4(left) with figure 4(right) tells us that the MLE achieves a much better accuracy. For example, when the input rate is 10 kHz, the confidence interval for the MLE is around 0.5 kHz. Nevertheless, for the 'rate coding' we see that the estimated input rate is 7 kHz (see dotted lines in figure 4), a much worse estimate. In fact from the general theory of statistics, we know that the MLE asymptotically attains the optimal bound determined by the Fisher information (see equation (11)). Hence, it is not too surprising to see that the MLE gives a better estimate.

5. Decoding of dynamical inputs

Now we turn our attention to decode dynamic input signals. The network is depicted in figure 5(left). There are 100 neurons, each of them receives an exactly balanced input with

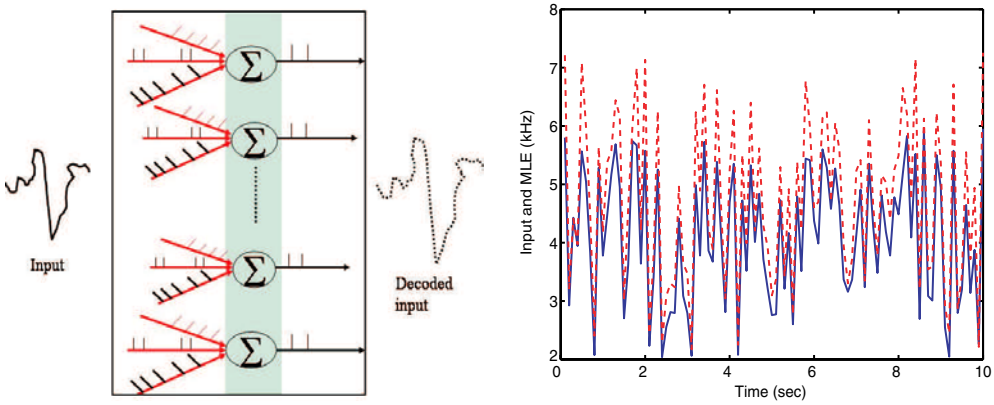

Figure 5. Left: a schematic plot of the network. Right: MLE (dotted lines) and input frequency (solid lines) (kHz) versus time. The time window used is 100 ms.

$\sigma(t)$ as described before. We assume that all inputs are independent (see section 7) and collect spikes in a fixed length of time window T_w. For example, if $T_w = 100$ ms, then we record 100 neurons with a 100 ms time window to decode the input information (figure 5(right)).

Surprisingly, data obtained from the MLE are always higher than inputs (figure 5(right)), in contrast with figure 4 where both higher and lower estimates than the actual data are obtained. It is then natural to ask what is going wrong here.

The problem lies in the assumption of ergodicity in Neuroscience which states that averaging over a long time window of a single neuron recording is equivalent to average over an ensemble of neurons, i.e. averaging over a larger number of identical neurons within a short time window. However, figure 5 clearly demonstrates that this assumption is problematic. Let us first understand why it happens. Consider the ith neuron in the group. We assume that within the time window of 100 ms, it fires no spikes. Hence, in the ensemble average, we will not count the neuron. However, in temporal average, it counts as a long interspike interval. As discussed before, a long interspike interval contributes to almost nothing to the numerator in the MLE, but increases the denominator. Hence, the temporal average in general gives a lower value than the ensemble average of a fixed time window.

Now let us formulate the problem mathematically. Define

$$N_i = \#\left\{ j, \sum_{k=1}^{j} t_{i,k} \leqslant T_w,\ j = 1, 2, \ldots \right\} \qquad i = 1, \ldots, \mathcal{N}$$

where T_w is the time window to collect spikes for each neuron, $t_{i,k}$ is the kth interspike interval collected from the ith neuron and $\mathcal{N}$ is the total number of neurones recorded with a time window $[0, T_w]$. $N = \sum_{i=1}^{\mathcal{N}} N_i$ is the total number of spikes. Let us define

$$\hat{\lambda}_{e,N} \frac{\sum_{i=1}^{\mathcal{N}} \sum_{j=1}^{\infty} f_a(t_{i,j}) I_{\{\sum_{k=1}^{j} t_{i,k} \leqslant T_w\}}}{N} \tag{14}$$

where $I_A(x)$ is the indicator function, i.e. $I_A(x) = 1$ if $x \in A$ and 0 otherwise. From equation (3.6), it is easily seen that $\hat{\lambda}_{e,N}$, omitting the constant $V_{thre}/2a\gamma$, is actually a truncated estimate of $\hat{\lambda}_N$.

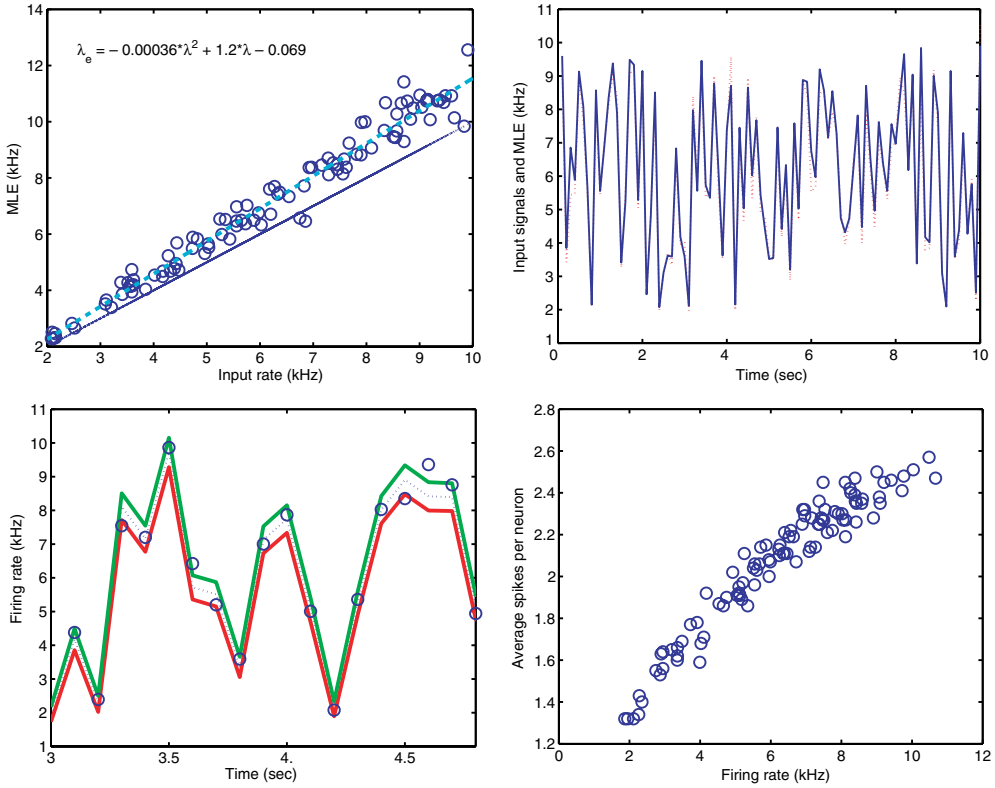

Figure 6. Upper panel: left, relationship between λ_e and λ; right, λ (solid lines) and MLE (dotted lines) versus time after readjustment according the relationship between λ_e and λ. Bottom panel: left, MLE and confidence intervals with thick lines indicating confidence intervals, '○' being true values and dotted lines MLE; right, average number of spikes per neuron within 100 ms window. Figures are generated using different realizations.

With a given random variable $t_{i,k}$ and under the assumption of the existence of the limit $\lim_N \hat{\lambda}_{e,N} = \lambda_e$, we can calculate the relationship between λ_e and λ explicitly (censored data in statistics). Hence,

$$\lambda = h(\lambda_e, T_w)$$

for a function h. Therefore, from biased ensemble average λ_e, we can recover the true value λ. Nevertheless, in the current paper, we do not intend to carry out a detailed calculation of the function h which is quite mathematically involved. In figure 6(left) (upper panel), we plot the relationship between λ_e and λ, i.e. h^{-1}, with $T_w = 100$ ms. The dashed line is obtained via data fitting. As claimed before, the ensemble average within a fixed time window is a biased estimate of λ, i.e. $\lambda_e > \lambda$. In figure 6, the relationship between λ_e and λ is

$$\lambda_e = -0.00036\lambda^2 + 1.2\lambda - 0.069.$$

Comparing figure 5 with figure 6 (upper panel), we see that a dramatic improvement of the MLE is obtained. In figure 5, in fact the range of inputs is smaller than the range of inputs in figure 6 (upper panel).

In conclusion, we have developed an algorithm which can reliably read out the input information based upon an ensemble average. To gain further information on the MLE

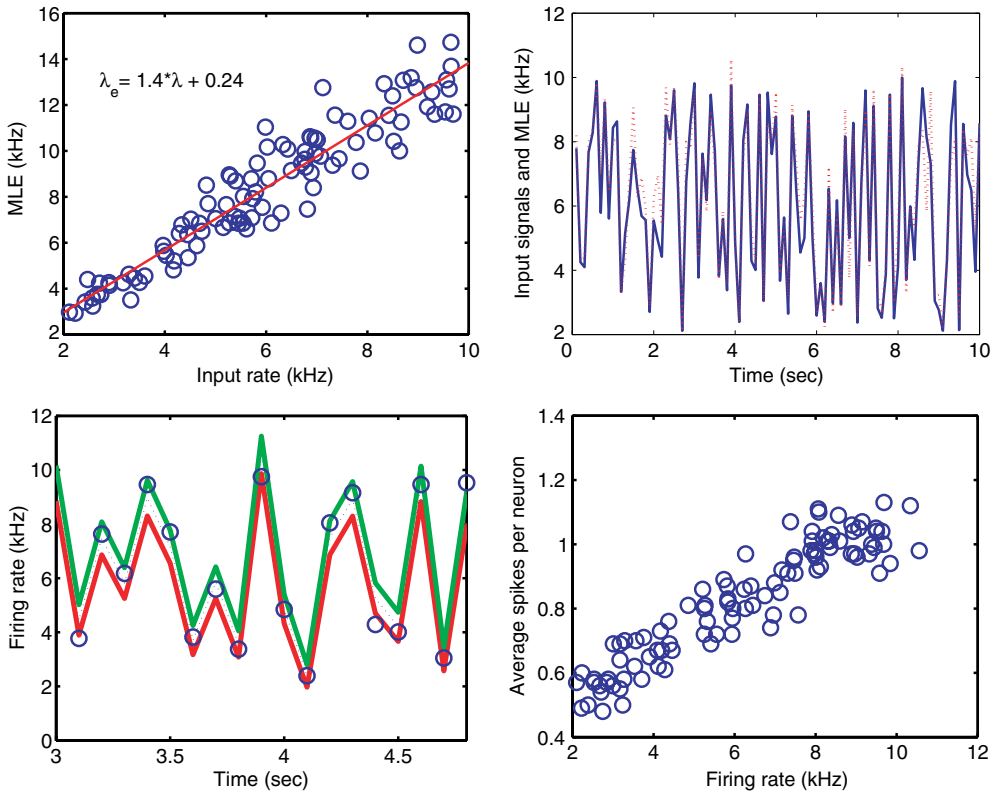

Figure 7. Upper panel: left, relationship between λ_e and λ; right, λ (solid lines) and MLE (dotted lines) versus time after readjustment according the relationship between λ_e and λ. Bottom panel: left, MLE and confidence intervals with thick lines indicating confidence intervals, '○' being true values and dotted lines MLE; right, average number of spikes per neuron within *50 ms window*. All figures are generated using different realizations.

approach, in figure 6 bottom panel (left), the confidence intervals according to theorem 2 are plotted. In figure 6 bottom panel (right), it is interesting to see that actually we have only used no more than three spikes in the MLE. In the extreme case, when neuronal input is around 2000 Hz, we need 1.3 spikes per neuron to decode the input.

The next and critical question is how the MLE depends on the length of the recording window T_w.

In figure 7, numerical results with $T_w = 50$ are depicted. It is illuminating to see when $T_w = 50$ ms, the decoding of dynamical signals is still quite good. Most surprisingly, figure 7 middle panel (right) shows that in general less than one spike per neuron is used in the MLE. Remembering that one of the main arguments to again rate coding is the speed of information processing, it requires hundreds of spikes to reliably decode the input information. In time coding, a single spike is enough to decode the input information. However, our results in figure 7 middle panel (right) clearly indicate that in population (rate) coding less than a single spike per neuron is sufficient to decode the input information.

To further confirm our conclusion, less than a spike is needed to decode the input information. In figure 8, we use $T_w = 29$ ms. It is clearly shown that at most 0.7 spike is needed to accurately decode the input information (see middle panel, right). Therefore, rate

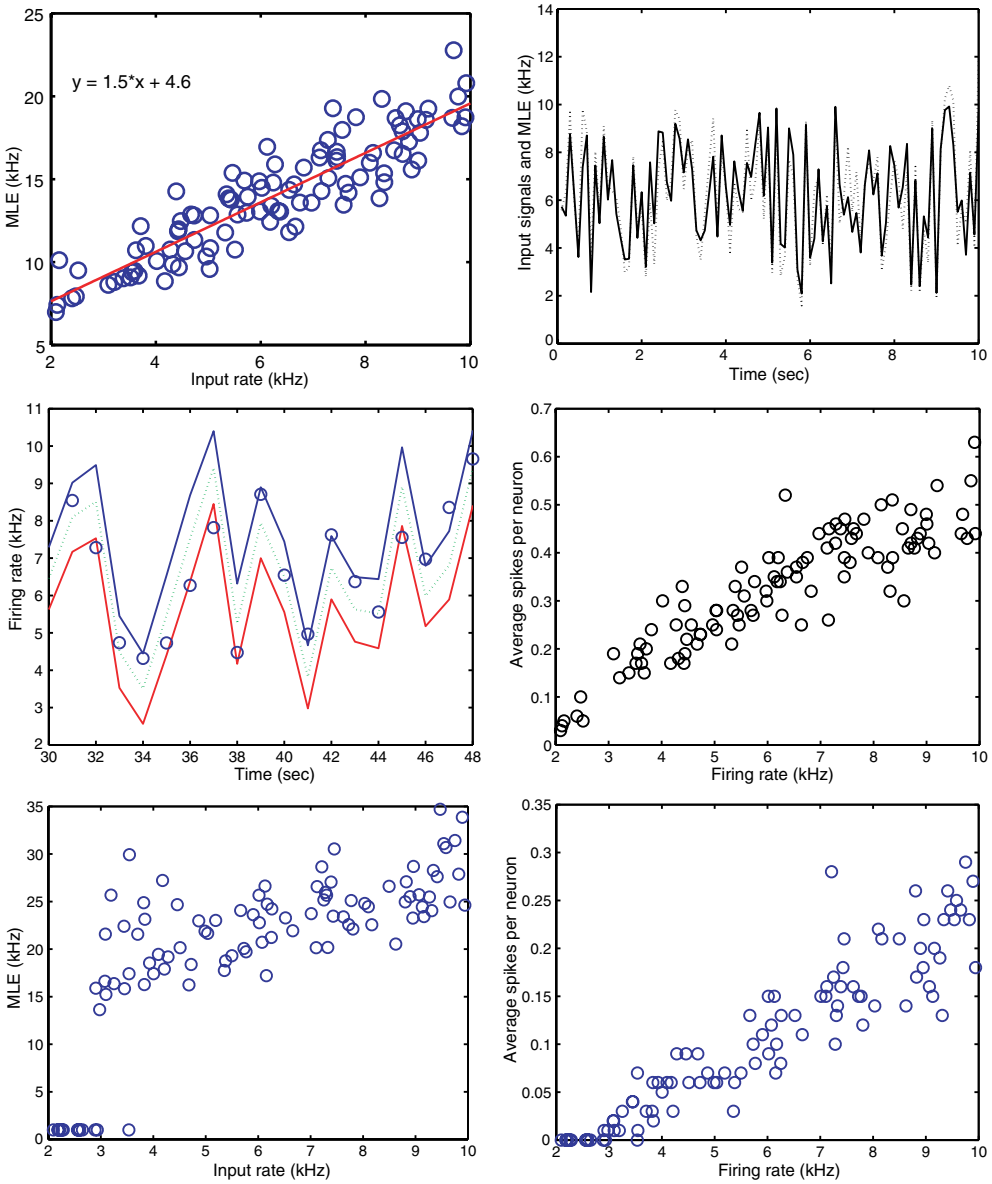

Figure 8. Upper panel: left, relationship between λ_e and λ; right, λ (solid lines) and MLE (dotted lines) versus time after readjustment according the relationship between λ_e and λ. Middle panel: left, MLE and confidence intervals with thick lines indicating confidence intervals, '○' being true values and dotted lines MLE; right, average number of spikes per neuron within *29 ms window*. All figures are generated using different realizations. Bottom panel: left, MLE versus input rate; right, average number of spikes per neuron with *20 ms window*.

coding could be even faster than time coding. In figure 8 (bottom panel), we further reduce T_w to 20 ms. However, in such a short time window, the whole population of neurons might be totally silent (no spike at all) and a decoding of the input information is surely impossible. It is interesting to compare figures 6 and 7 with figure 8. In figure 6, more than one spike is

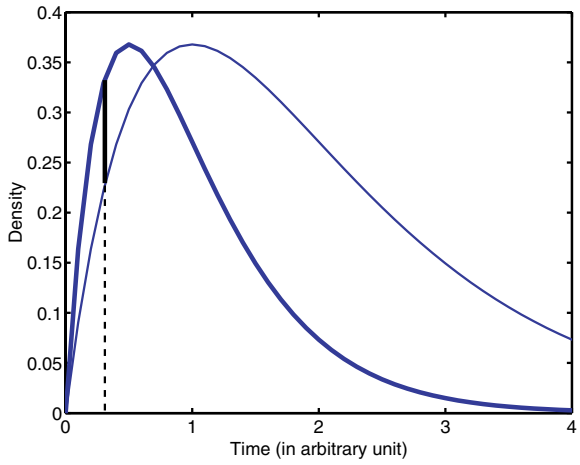

Figure 9. A schematic plot of reading spike distribution within a short time window. The time window on the left side of thick lines (dashed and solid) will be enough for the nervous system to read out the input information.

used to decode. As a result, the decoding accuracy is better than that of figure 7 (intermediate case) and figure 8 (less than a spike is employed to decode).

We can easily understand why this is the case. Assume that two received signals (spike trains) are distributed as depicted in figure 9. If we have a short time window but with a large number of samples, the samples would be enough for us to predict the actual distribution.

6. Bayesian estimate

It is known that Jeffreys prior [10] is defined by

$$p_{\mathrm{J}}(\lambda) \propto I^{1/2}(\lambda).$$

It is not difficult to check that the Jeffreys prior is proper, i.e. $\int p_{\mathrm{J}}(\lambda)\,\mathrm{d}\lambda < \infty$. Hence, the posterior distribution is

$$p_{\mathrm{P}}(\lambda) \propto p(x|\lambda)p_{\mathrm{J}}(\lambda).$$

To find λ which maximizes the posterior distribution is equivalent to finding the solution of the following equation:

$$2Np_{\mathrm{J}}'\sigma^4 - 2\sigma^2 a^2 p_{\mathrm{J}} + 4a^2 p_{\mathrm{J}} \sum_{i=1}^{N} f_2(t_i) = 0$$

or

$$a^2\sigma^2 - (\log p_{\mathrm{J}})'\sigma^4 = \frac{2a^2 \sum_{i=1}^{N} f_2(t_i)}{N}.$$

The optimal Bayesian estimate λ_{B} is simply the solution of the following equation:

$$\sigma^2(\lambda_{\mathrm{B}}) - \frac{I'(\lambda_{\mathrm{B}})}{2I(\lambda_{\mathrm{B}})}\left(2a\lambda_{\mathrm{B}} - \frac{V_{\mathrm{thre}}}{\gamma}\right)^2 = \frac{2\sum_{i=1}^{N} f_2(t_i)}{N}. \tag{15}$$

The implications of equation (7) are clear. The second term, $-\frac{I'(\lambda_{\mathrm{B}})}{2I(\lambda_{\mathrm{B}})}\left(2a\lambda_{\mathrm{B}} - \frac{V_{\mathrm{thre}}}{\gamma}\right)^2$, on the left-hand side of equation (7) is always positive. Hence, the Bayesian estimate tends to be

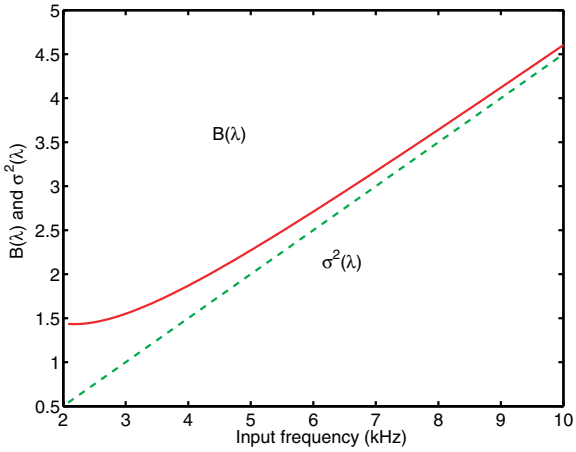

Figure 10. $B(\lambda)$ and $\sigma^2(\lambda)$ versus λ. When the input is higher, $B(\lambda)$ and $\sigma^2(\lambda)$ converge, i.e. the Bayesian estimate and the MLE are identical.

smaller than the maximum likelihood estimate. From figure 10 we can see that the larger the estimate, the larger the error. The Bayesian estimate makes sure that the risk of making mistake is smaller. From equation we see that

$$I'(\lambda) = -\frac{12a^6\langle f_2^2\rangle}{\sigma^8} + \frac{10a^6\langle f_2(t)\rangle}{\sigma^7} - \frac{2a^6}{\sigma^6}. \tag{16}$$

Combining equation (7) with equation (16) we conclude that λ_B is the solution of the following equation:

$$\begin{aligned} B(\lambda) &= \sigma^2(\lambda) - \left[-\frac{6a^6\langle f_2^2\rangle}{\sigma^8} + \frac{5a^6\langle f_2(t)\rangle}{\sigma^7} - \frac{a^6}{\sigma^6}\right]\left(2a\lambda - \frac{V_{\text{thre}}}{\gamma}\right)^2 \\ &= \frac{2\sum_{i=1}^{N} f_2(t_i)}{N}. \end{aligned} \tag{17}$$

As we pointed out before, the Bayesian estimate is lower than the MLE estimate (figure 10). When the input rate is low, the difference between the MLE and the Bayesian estimate is large. They gradually converge when the input rate is high.

7. Discussions

We have presented an approach to decode the input information based upon the output from a spiking neuron network and the maximum likelihood estimate.

The results reported in the current paper could be valuable both for neuroscience and for engineering applications of spiking neuron networks. For neuroscience, as mentioned above, we provide the first template in the literature on how to reliably read out the information in spiking neuron networks; we have clarified a few key issues in neuroscience. For engineering applications, although the expectation of the application of spiking neuron networks to solving practical problems is high, how to decode the actual input information in a stochastic spiking neuron network is not known. To the best of our knowledge, our approach is the first one to really address and solve the issue, although the idea is around for many years [7]. The basic difficulty lies in the fact that we do not have an analytical formula for the interspike interval

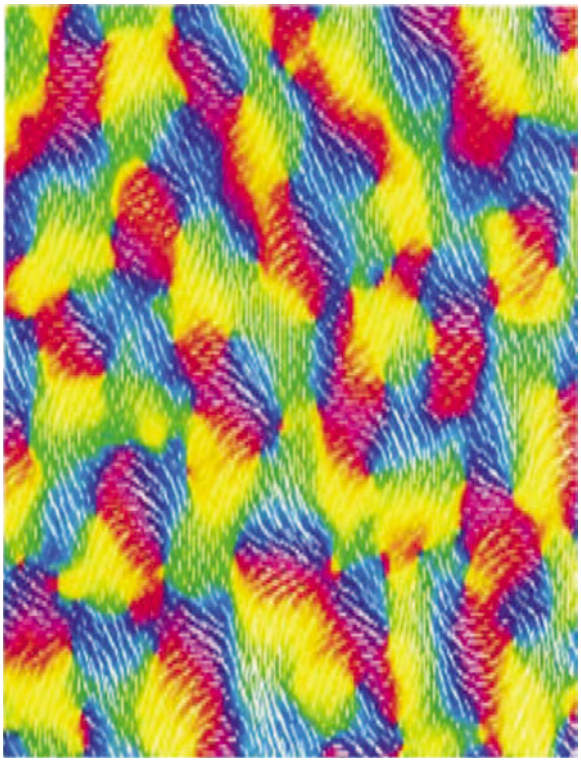

Figure 11. Distribution of orientation patches in V1 revealed by optical imaging. Orientation patches with identical orientation are separated in V1.

distribution and so we are not able to have a rigorous maximum likelihood estimate [2]. As a consequence, it is not possible to test, for example, how large a window is needed to decode the input information. Our approach is in the framework of population coding which has been extensively studied in the literature, see for example [3, 20, 18]. However, our approach is totally different from their. In their work, the starting point is the firing rate of neurons. It would be interesting to combine the two approaches. In [6], the author carried out a study on the dynamics of the first-order statistics: the mean firing rate, of a population of neurons, i.e. a rate coding approach as discussed in figure 4. We note that the decoding of the input information as we rigorously developed here is not simply a rate coding approach. From equation (3.6) we see that the MLE is a nonlinear function of interspike intervals t_i.

There are two essential issues we have not addressed here. The first one is the interaction among neurons. We have assumed that no interactions exist among the neurons. This is obviously a oversimplification. How can we introduce interactions in the network and at the same time ensure the exactly balanced assumption? For each excitatory input arising from the neighbourhood neurons, we can introduce a delayed inhibitory input. Hence, each neuron in the networks receives an exactly balanced input and our approach applies. Upon further reflection this might not be an unrealistic assumption in some circumstances. For example, in figure 11, we intend to read out the orientation of the input, say a light bar. We can treat neurons from one orientation patch as a single unit. Note that orientation patches with the identical orientation are separated and so we could safely assume that the interaction between them is weak. Hence, our approach can be directly applied.

The second issue is the correlated input between neurons [21, 12]. We would easily expect that nearby neuron activity is correlated, although various mechanisms have been introduced to break down the input correlation [1, 8]. Whether a correlation will increase or decrease the decoding accuracy is not known. We will explore the issue in further publications.

Finally, we want to emphasize that although we have concentrated on estimating a single parameter within each time window, we have actually treated the model as a probability model. The input distribution of interspike intervals is uniquely determined by the single parameter. What we actually achieved is to read out input *probability flows*, based upon observed multiple spike trains. In the near future, we will carry out 'experiments' on spiking neuronal networks and test some ideas as described in [1].

Acknowledgments

We are grateful to Henry C Tuckwell and the referees for their valuable comments on the paper. J F was partially supported by grants from UK EPSRC (GR/R54569/01), EPSRC(GR/20574), and a grant of the Royal Society. MD's work was supported by US NIH, NSF and ONR.

References

[1] Barlow H 2001 Redundancy reduction revisited *Network-Comp. Neural* **12** 241–53
[2] Brown E N, Barbieri R, Eden U T and Frank L M 2003 Likelihood methods for neural spike train data analysis *Computational Neuroscience: A Comprehensive Approach* ed J F Feng (Boca Raton, FL: CRC Press)
[3] Deneve S, Latham P E and Pouget A 1999 Reading population codes: a neural implementation of ideal observers *Nat. Neurosci.* **2** 740–5
[4] Feng J F 2001 Is the integrate-and-fire model good enough? A review *Neural Networks* **14** 955–75
[5] Feng J F 2003 *Computational Neuroscience—A Comprehensive Approach* ed J F Feng (London/Boca Raton, FL: Chapman and Hall/CRC Press)
[6] Gerstner W 2000 Population dynamics of spiking neurons: fast transients, asynchronous states, and locking *Neural Comput.* **12** 43–89
[7] Gerstner W and Kistler W 2002 *Spiking Neuron Models Single Neurons, Populations, Plasticity* (Cambridge: Cambridge University Press)
[8] Goldman M S, Maldonado P and Abbott L F 2002 Redundancy reduction and sustained firing with stochastic depressing synapses *J. Neurosci.* **22** 584–91
[9] Koch C 1999 *Biophysics of Computation* (Oxford: Oxford University Press)
[10] Lehmann E and Casella G 1999 *Theory of Point Estimation* (New York, Berlin: Springer)
[11] Rieke F *et al* 1997 *Spikes: Exploring the Neural Code* (Cambridge, MA/London: MIT Press)
[12] Salinas E and Sejnowski T 2001 Correlated neuronal activity and the flow of neural information *Nat. Rev. Neurosci.* **2** 539–50
[13] Shadlen M N and Newsome W T 1994 Noise, neural codes and cortical organization *Curr. Opin. Neurobiol.* **4** 569–79
[14] Thorpe S, Fize D and Marlot C 1996 Speed of processing in the human visual system *Nature* **381** 520–2
[15] Tuckwell H C 1988 *Introduction to Theoretical Neurobiology* vol 2 (Cambridge: Cambridge University Press)
[16] Tuckwell H C 2003 private communications
[17] Grossberg S, Maass W and Markram H 2001 *Neural Networks* **14** 587–975
[18] Pouget A, Zemel R S and Dayan P 2000 Information processing with population codes *Nat. Rev. Neurosci.* **1** 125–32
[19] van Vreeswijk C and Sompolinsky H 1996 Chaos in neuronal networks with balanced excitatory and inhibitory activity *Science* **274** 1724–6
[20] Wu S, Amari S and Nakahara H 2002 Population coding and decoding in a neural field: a computational study *Neural Comput.* **14** 999–1026
[21] Zohary E, Shadlen M N and Newsome W T 1994 Correlated neuronal discharge rate and its implications for psychophysical performance *Nature* **370** 140–3