Decoding spike ensembles: tracking a moving stimulus

Enrico Rossoni*, Jianfeng Feng**
*University of Sussex, Brighton, UK
**University of Warwick, Coventry, UK

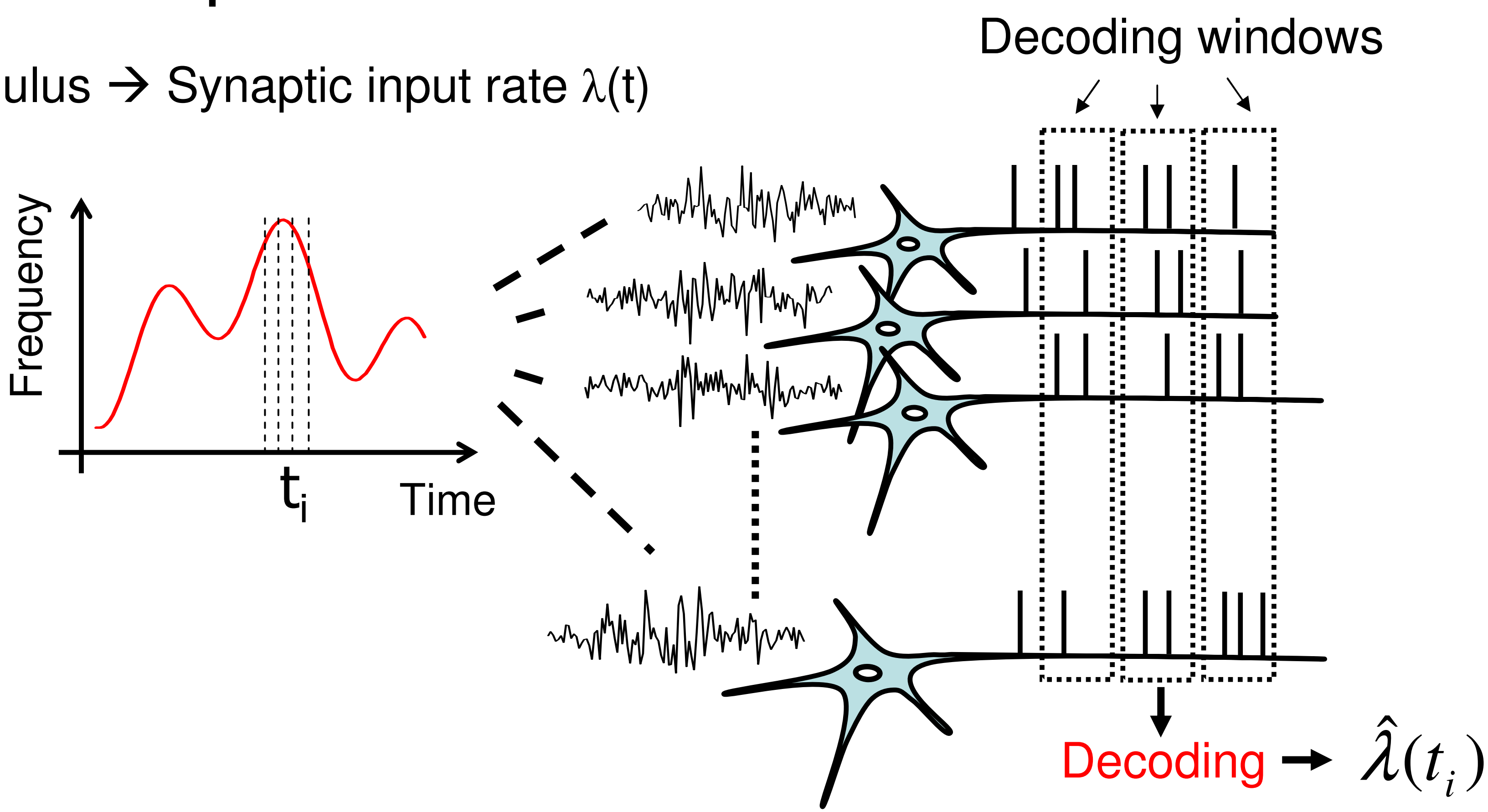
The questions

- Assuming that cortical neurons *in vivo* are ‘noisy’, how can we read out the information encoded in their spike activity?
- What is the most effective strategy for decoding spiking neurons?
- How accurately/rapidly can we extract information from a spike train ensemble?

This issues can be addressed theoretically for simple spiking neuronal models

The setup

Stimulus → Synaptic input rate $\lambda(t)$



The stimulus is encoded in the time-dependent synaptic input rate $\lambda(t)$. In the decoding stage, the output spike trains are 'parsed' into consecutive time windows and the input is inferred from the observed ISI distribution.

Methods

- Neurons are described by the leaky Integrate-and-Fire model
- The synaptic input is given in the diffusion approximation

$$dI_t = \mu dt + \sigma dB_t$$

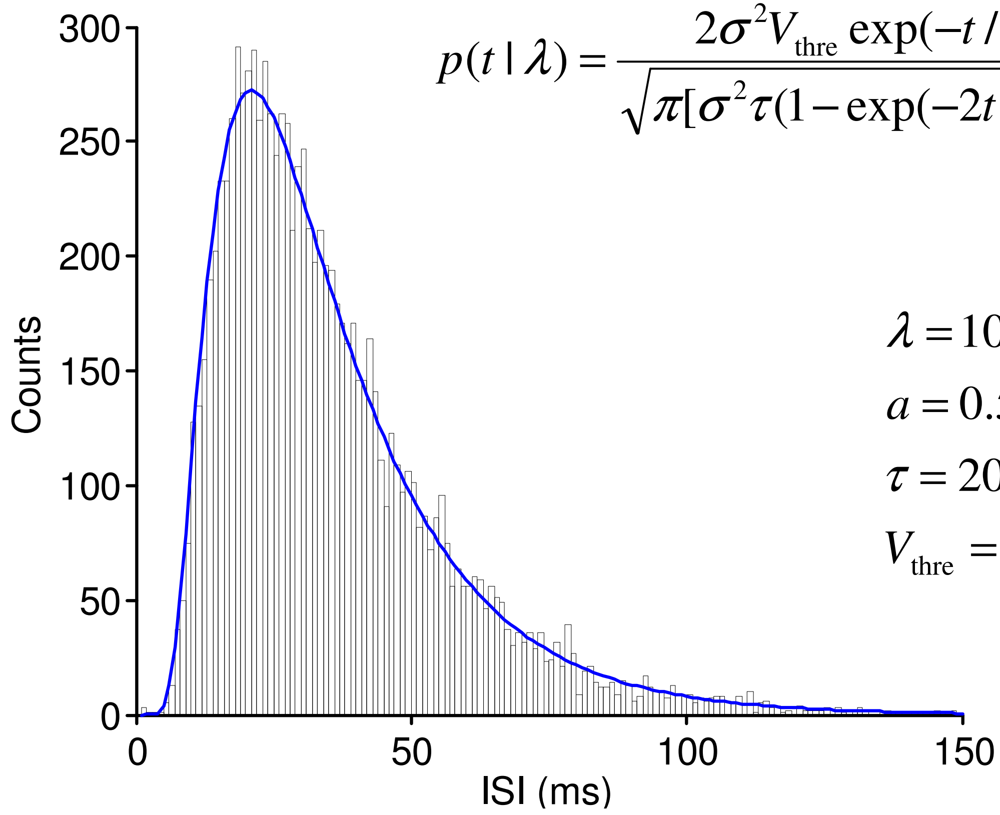
$$\mu = a\lambda(1-r) \qquad \sigma = a\sqrt{\lambda(1+r)}$$

- Inputs to different neurons are assumed to be independent
- The excitatory vs. inhibitory ratio (*r*) is adjusted to ensure that the input is ‘balanced’, i.e. that the mean membrane potential is always at threshold

For this model we derived the exact ISI probability density, thus we can adopt an optimal decoding strategy using a Maximum-Likelihood estimator.

$$p(t \mid \lambda) = \frac{2\sigma^2 V_{\text{thre}} \exp(-t/\tau)}{\sqrt{\pi[\sigma^2\tau(1-\exp(-2t/\tau))]^3}} \exp\left\{-\frac{(V_{\text{thre}})^2 \exp(-2t/\tau)}{\sigma^2\tau[1-\exp(-2t/\tau)]}\right\}$$

$$\lambda = 10\,\text{kHz}$$
$$a = 0.5\,\text{mV}$$
$$\tau = 20\,\text{ms}$$
$$V_{\text{thre}} = 20\,\text{mV}$$



The histogram of the ISI generated by the LIF model compared with the theoretical ISI distribution (solid blue)

The ‘censoring’ problem

- In the time-dependent case, the input must be inferred from short spike train samples.

  However, when the decoding window length (T) is finite, the ‘standard’ ML estimator is biased.

  Indeed, the presence of a finite window length (censoring) modifies the ISI distribution and introduces serial dependencies between consecutive intervals in each window.

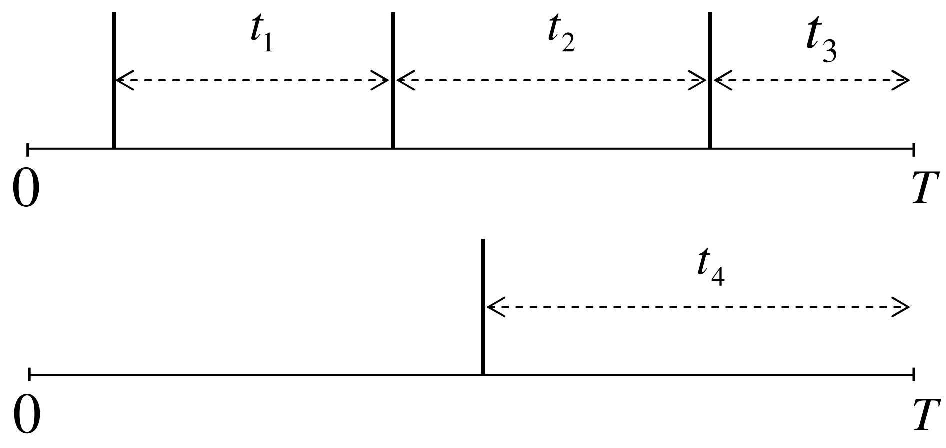
- Also, when T is comparable with the ‘typical’ ISI, only few intervals are expected to fall in each window, leading to poor estimates.

According to the theory of censored data in statistics, an unbiased estimator of λ is given by

$$\lambda_{CMLE} = \arg\max L_C(\lambda)$$

where the censored likelihood function is

$$L_C(\lambda) = \sum_{i \in R} \log p(t_i \mid \lambda) + \sum_{i \in T} \log S(t_i \mid \lambda) \qquad S(t \mid \lambda) = \int_t^{+\infty} p(t' \mid \lambda)\, dt'.$$



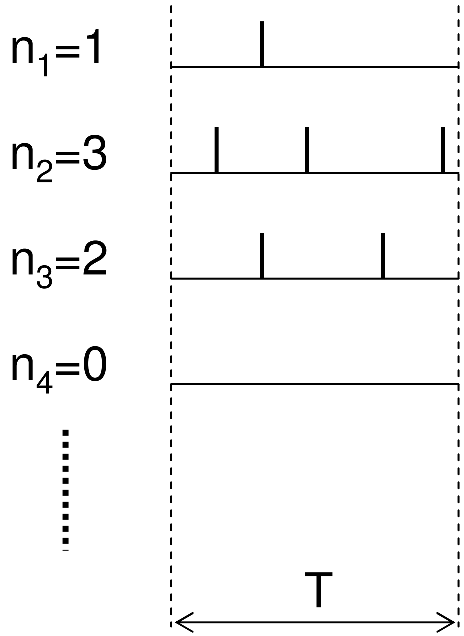
Regular intervals

$$\{t_i\}_{i \in R} = \{t_1, t_2\}$$

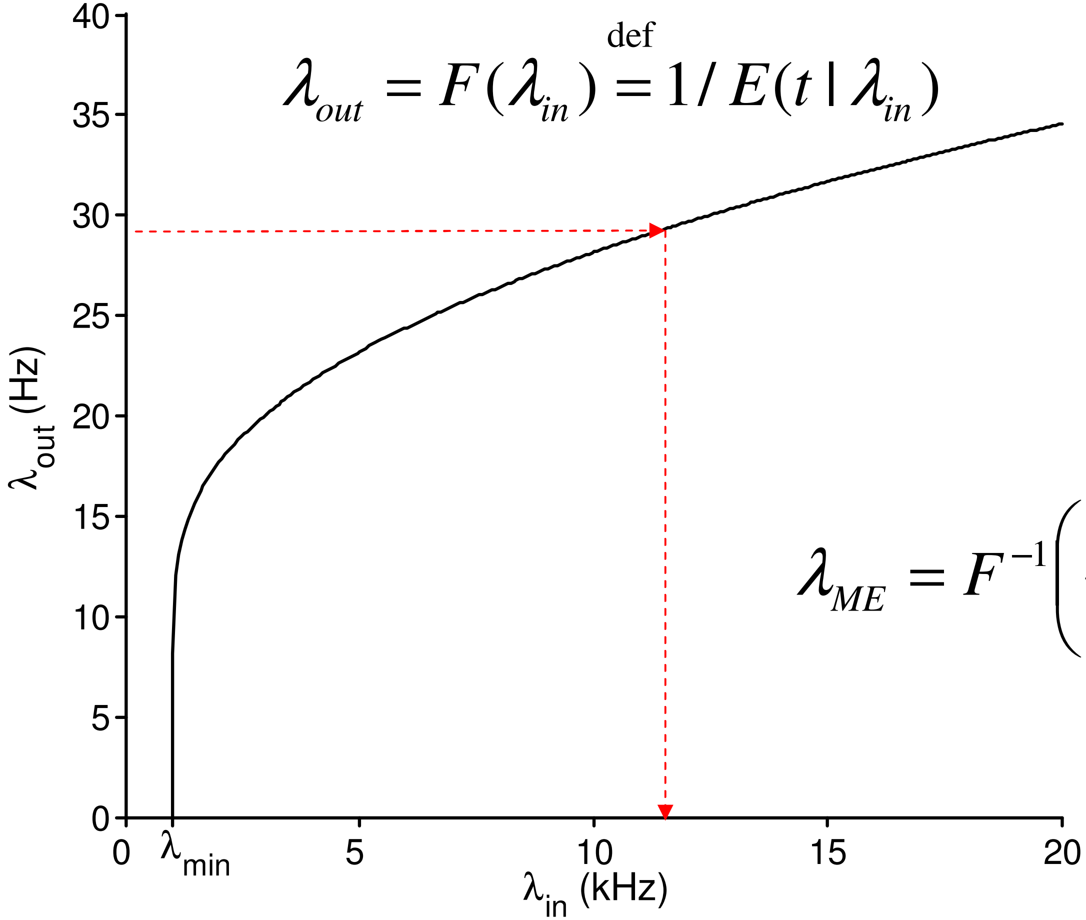
Censored intervals

$$\{t_i\}_{i \in T} = \{t_3, t_4\}$$

In the rate-coding approach, a Moment Estimate (ME) of the stimulus is obtained from the mean spike count,



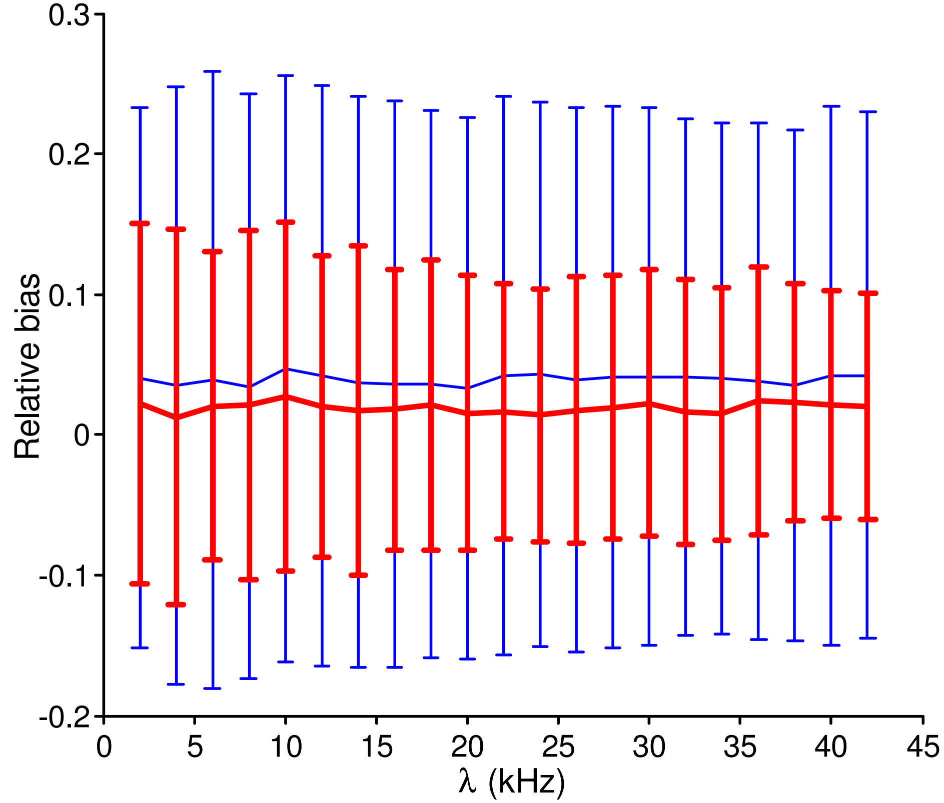
$$\overline{n} = \frac{\sum_{i=1}^{N} n_i}{N}$$



$$\lambda_{out} = F(\lambda_{in}) \overset{\text{def}}{=} 1/E(t \mid \lambda_{in})$$

$$\lambda_{ME} = F^{-1}\left(\frac{\overline{n}}{T}\right)$$

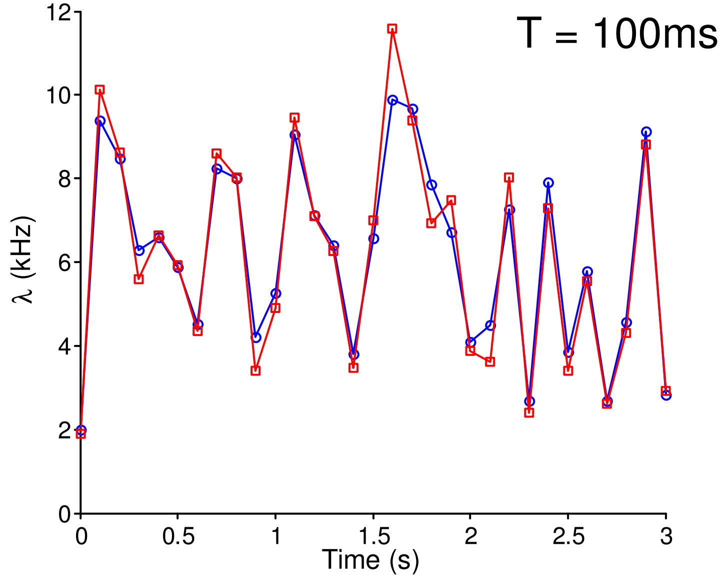
T = 50ms

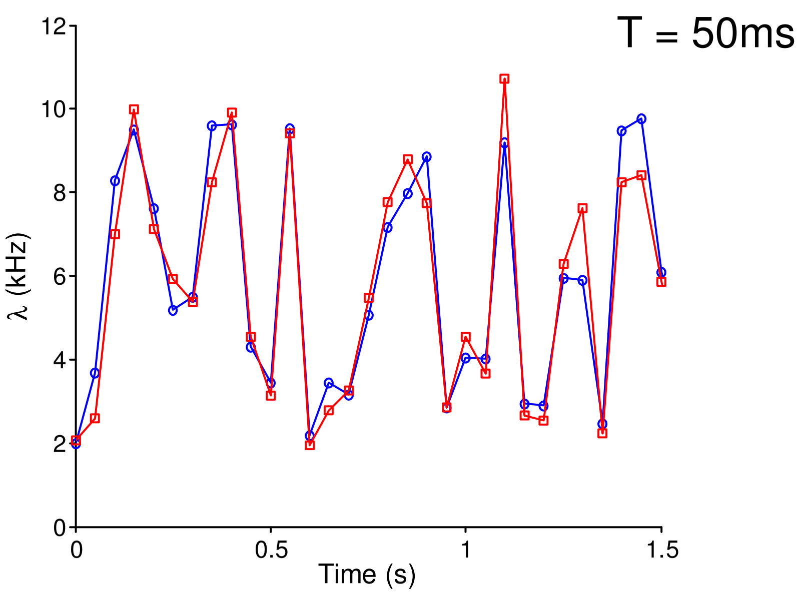


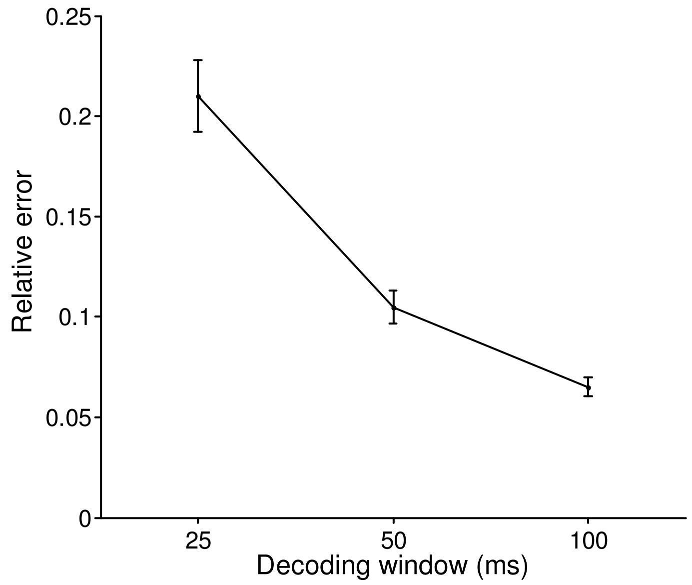
The Censored Maximum Likelihood Estimator (red) is more efficient than the Moment Estimator (blue) as indicated by the smaller SD of the estimates (bars). The bias is almost entirely removed.

Example

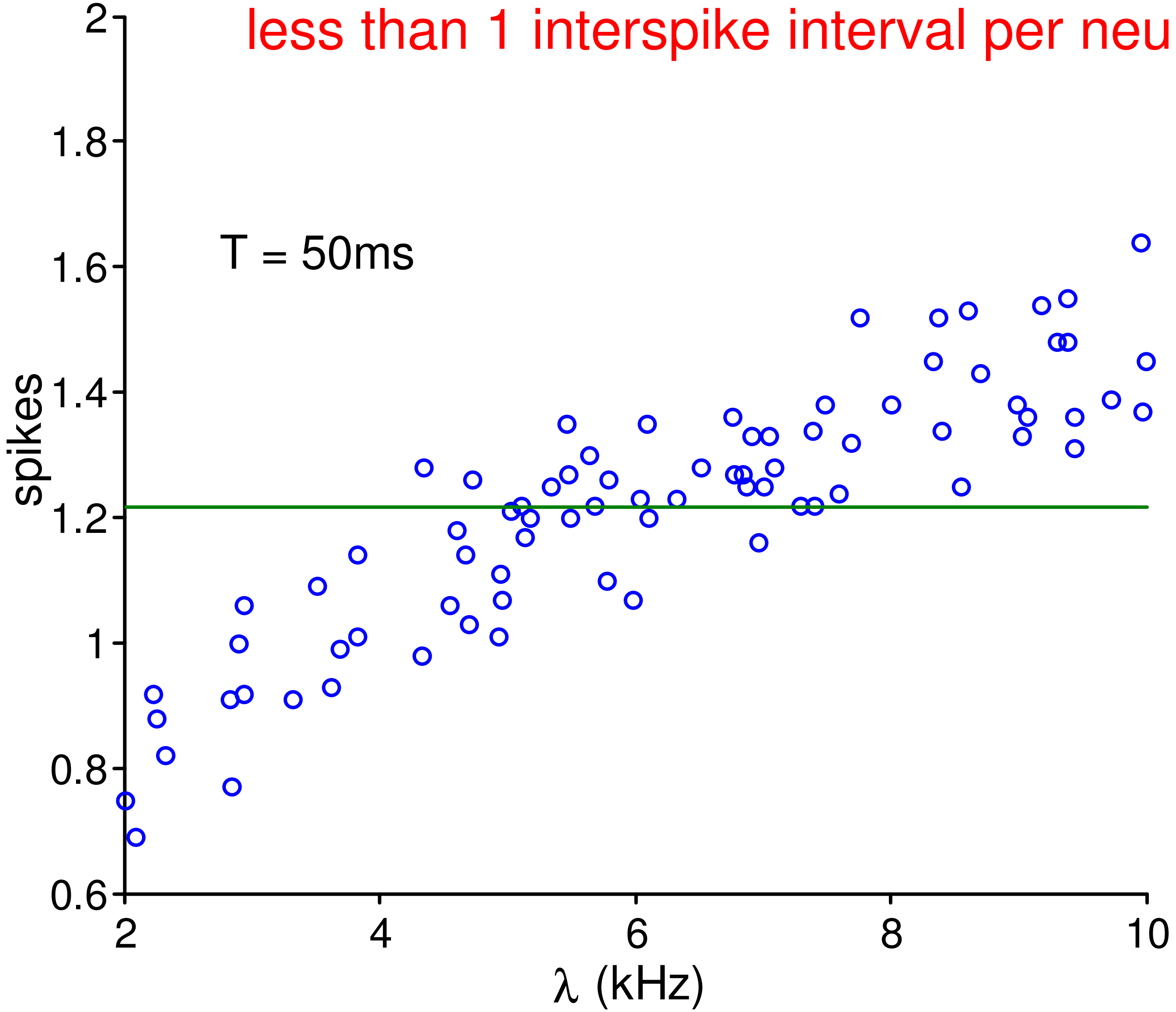
A random steplike signal (red) is estimated via CMLE (blue) using decoding windows of length T.





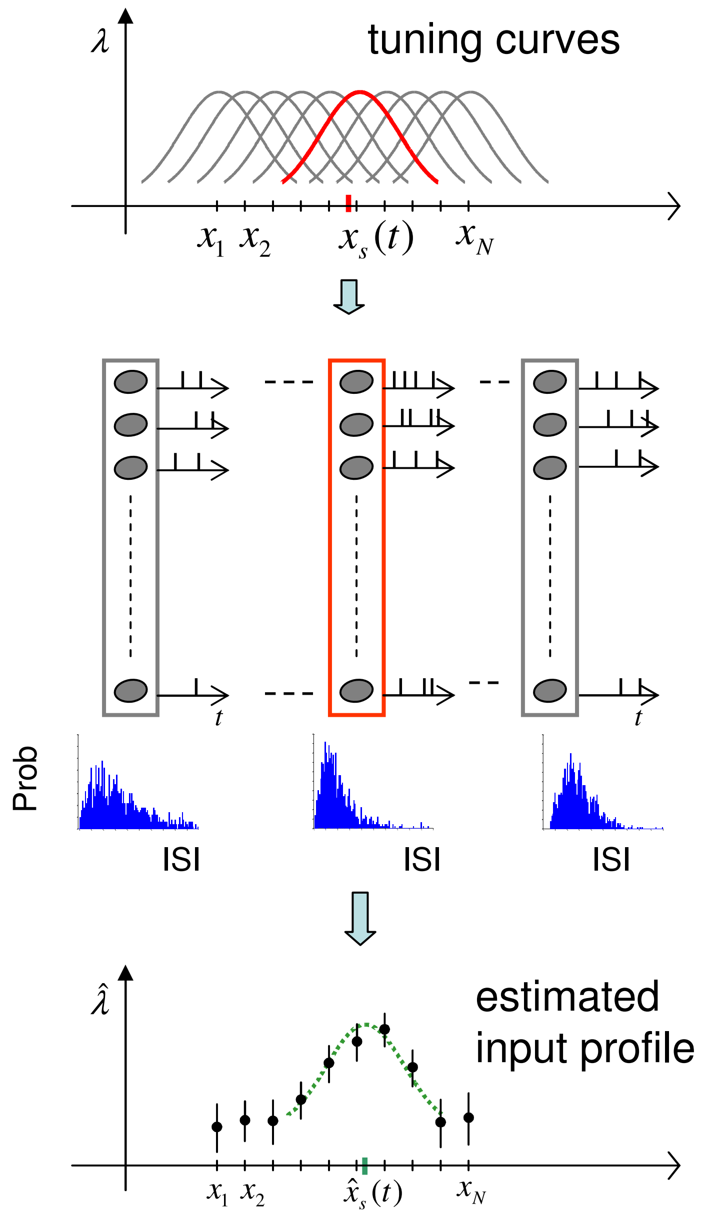


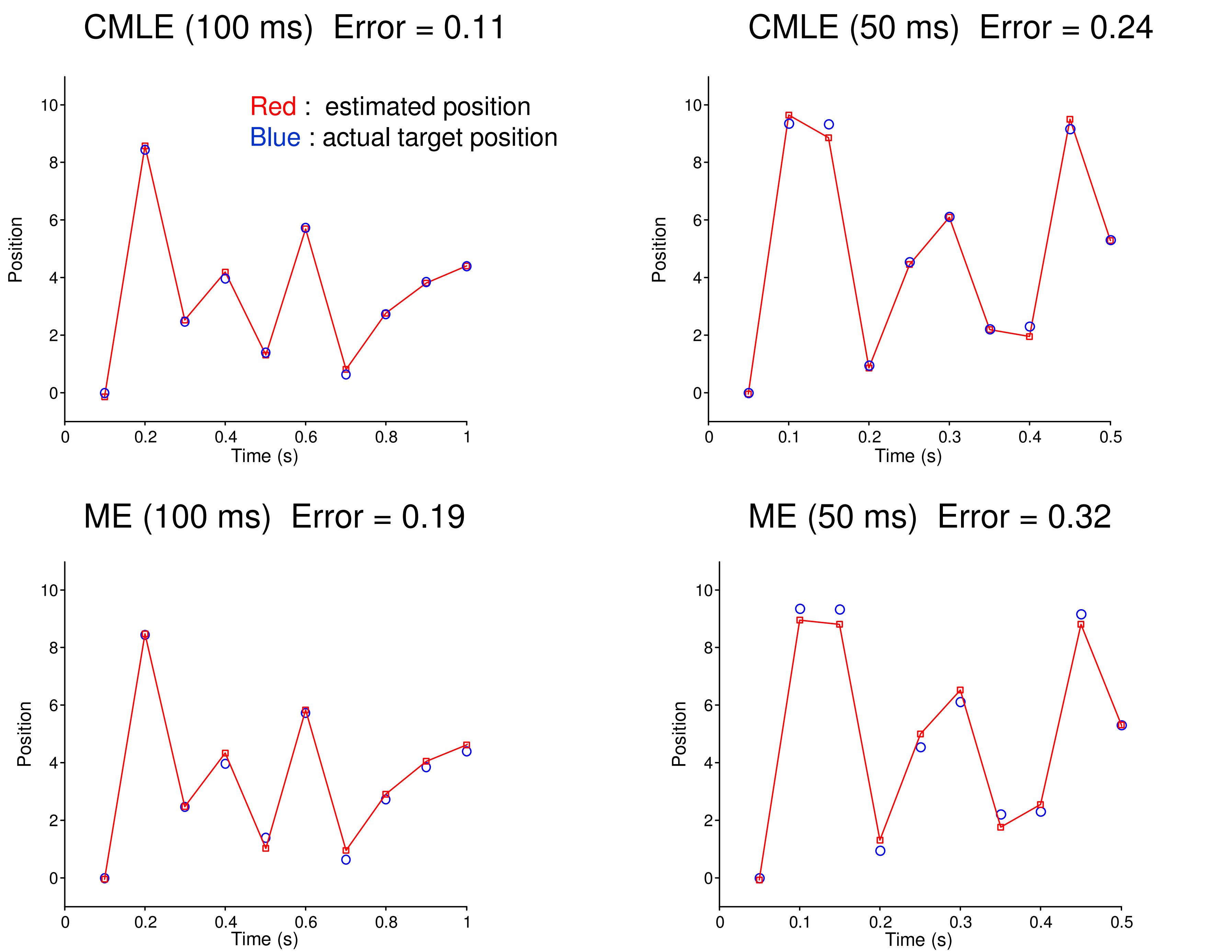
The signal is decoded using an average of
less than 1 interspike interval per neuron

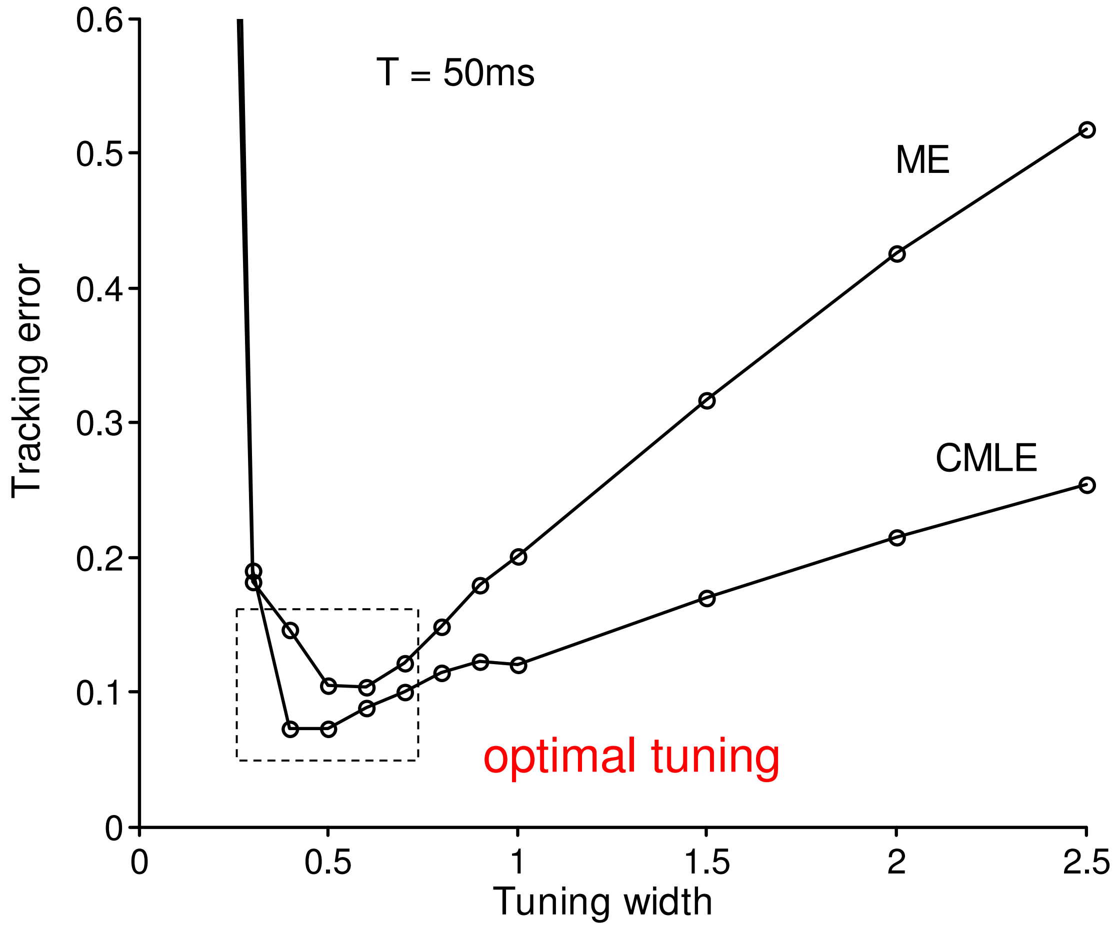


The tracking task

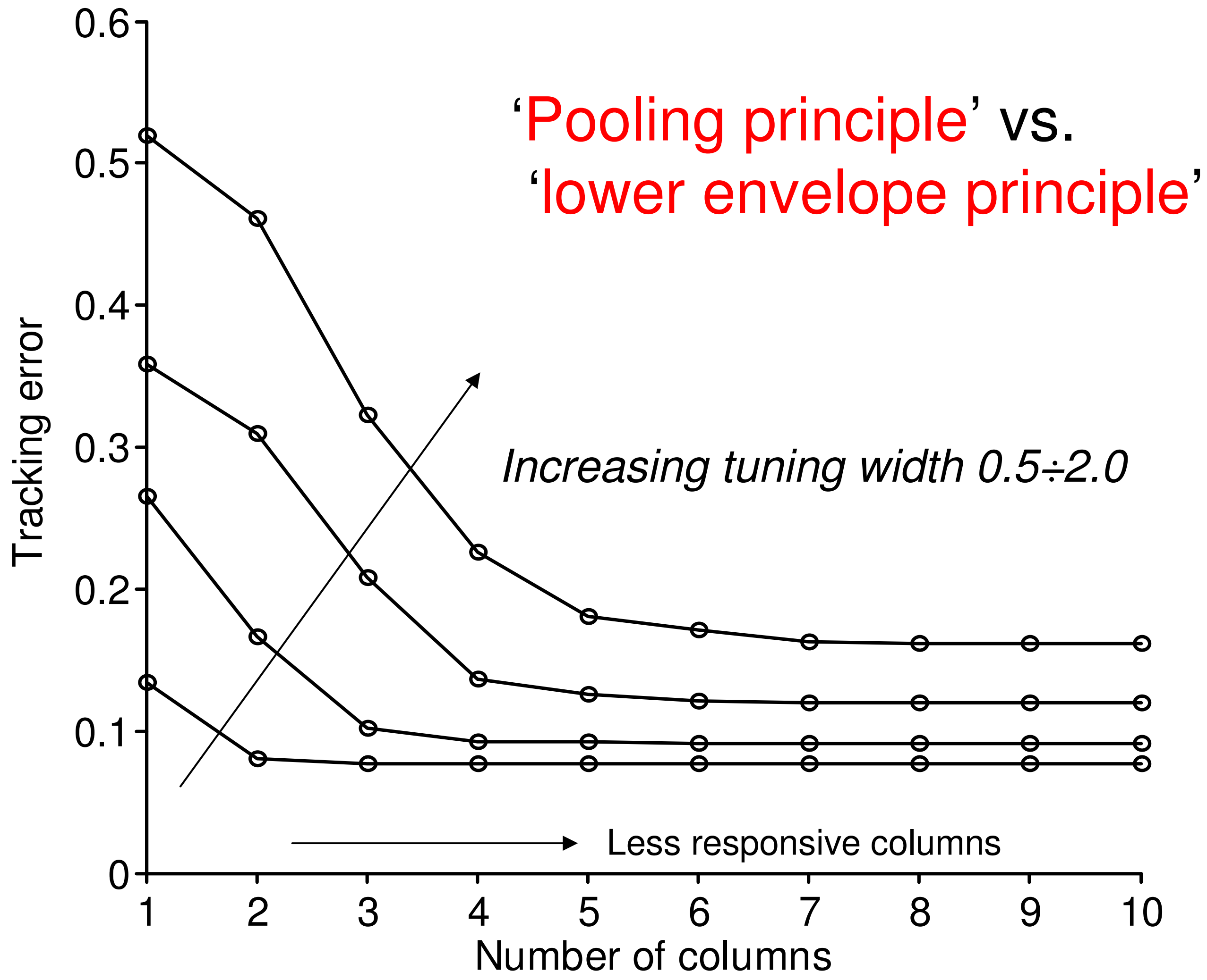
- The position of a target (x_s) is changed randomly at regular intervals of time.
- Neurons are grouped in columns and tuned to specific stimulus locations (x_i) to create a 'topographical map'
- Using CMLE/ME, the input to each column is inferred from the spike activity observed during the latest decoding window.
- The estimated input profile is fitted to determine the most likely position of the target.




CMLE (100 ms) Error = 0.11
Red : estimated position
Blue : actual target position
Position
Time (s)
CMLE (50 ms) Error = 0.24
Position
Time (s)
ME (100 ms) Error = 0.19
Position
Time (s)
ME (50 ms) Error = 0.32
Position
Time (s)



How does the tuning width affect the tracking accuracy?



Is the tracking accuracy increased by considering also less responsive columns for decoding? Would this increase noise instead?

Conclusions

- We present an optimal unbiased strategy to decode the input information from spike train ensembles
- We have applied such approach to a spiking neuronal network in order to perform a tracking task
- We suggest that general issues regarding population coding can be addressed within this framework