Entropy **2014**, *16*, 2244-2277; doi:10.3390/e16042244

OPEN ACCESS
entropy
ISSN 1099-4300
www.mdpi.com/journal/entropy

Article

Parameter Estimation for Spatio-Temporal Maximum Entropy Distributions: Application to Neural Spike Trains

Hassan Nasser * **and Bruno Cessac** *

INRIA, 2004 route de lucioles, 06560, Sophia-Antipolis, France

* Author to whom correspondence should be addressed; E-Mails: Hassan.Nasser@inria.fr (H.N.); Bruno.Cessac@inria.fr (B.C.).

Received: 19 February 2014; in revised form: 28 March 2014 / Accepted: 8 April 2004/ Published: 22 April 2014

Abstract: We propose a numerical method to learn maximum entropy (MaxEnt) distributions with spatio-temporal constraints from experimental spike trains. This is an extension of two papers, [10] and [4], which proposed the estimation of parameters where only spatial constraints were taken into account. The extension we propose allows one to properly handle memory effects in spike statistics, for large-sized neural networks.

Keywords: neural coding; Gibbs distribution; maximum entropy; convex duality; spatio-temporal constraints; large-scale analysis; spike train; MEA recordings

1. Introduction

With the evolution of multi-electrode array (MEA) acquisition techniques, it is currently possible to simultaneously record the activity of a few hundred neurons up to a few thousand [1]. Stevenson *et al.* [2] reported that the number of recorded neurons doubles approximately every eight years. However, beyond the mere recording of an increasing number of neurons, there is a need to extract relevant information from data in order to understand the underlying dynamics of the studied network, how it responds to stimuli and how the spike train response encodes these stimuli. In the realm of spike train analysis, this means having efficient spike sorting techniques [3–6], but also efficient methods to analyze spike statistics. The second aspect requires using canonical statistical models, whose parameters have to be tuned ("learned") from data.

The maximum entropy method (MaxEnt) offers a way of selecting canonical statistical models from first principles. Having its root in statistical physics, MaxEnt consists of fixing a set of constraints,

determined as the empirical average of features measured from the spiking activity. Maximizing the statistical entropy given those constraints provides a unique probability, called a Gibbs distribution, which approaches, at best, data statistics in the following sense: among all probability distributions that match the constraints, this is the one that has the smallest Kullback-Leibler divergence with the data ([7]). Equivalently, it satisfies the constraints without adding additional assumption on the statistics [8].

Most studies have focused on properly describing the statistics of spatially-synchronized patterns of neuronal activity without considering time-dependent patterns and memory effects. In this setting, pairwise models [9,10] or extensions with triplets and quadruplets interactions [11–13] were claimed to correctly fit ≈ 90 to 99% of the information. However, considering now the capacity of these models to correctly reproduce spatio-temporal spike patterns, the performances drop-off dramatically, especially in the cortex [14,15] or in the retina [16].

Taking into account spatio-temporal patterns requires introducing memory in statistics, described as a Markov process. MaxEnt extends easily to this case (see Section 2.2 and the references therein for a short description) producing Gibbs distributions in the spatio-temporal domain. Moreover, rigorous mathematical methods are available to fit the parameters of the Gibbs distribution [16]. However, the main drawback of these methods is the huge computer memory they require, preventing their applications to large-scale neural networks. Considering a model with memory depth D (namely, the probability of a spike pattern at time t depends on the spike activity in the interval $[t-D, t-1]$), there are $2^{N(D+1)}$ possible patterns. The method developed in [16] requires one to handle a matrix of size $2^{N(D+1)} \times 2^{N(D+1)}$. Therefore, it becomes intractable for $N(D+1) > 20$.

In this paper, we propose an alternative method to fit the parameters of a spatio-temporal Gibbs distribution with larger values of the product, $N(D+1)$. We have been able to go up to $N(D+1)$ (~ 120) on a small cluster (64 processors AMD Opteron(tm) $2,300$ MHz). The method is based on [17] and [18], who proposed the estimation of parameters in *spatial* Gibbs distributions. The extension in the spatio-temporal domain is not straightforward, as we show, but it carries over to the price of some modifications. Combined with parallel Monte Carlo computing developed in [19], this provides a numerical method, allowing one to handle Markovian spike statistics with spatio-temporal constraints.

The paper is organized as follow. In Section 2, we recall the theoretical background for spike trains with a Gibbs distribution. We discuss both the spatial and spatio-temporal case. In the next section, 3, we explain the method to fit the parameters of MaxEnt distributions. As we mathematically show, the convex criterion used by [17] still applies for spatio-temporal constraints. However, the method used by [18] to avoid recomputing the Gibbs distribution at each parameters change cannot be directly used and has to be adapted using a linear response scheme. In the last section, 4, we show benchmarks evaluating the performance of this method and discuss the computational obstacles that we encountered. We made tests with both synthetic and real data. Synthetic data were generated from known probability distributions using a Monte Carlo method. Real data correspond to spike trains obtained from retinal ganglion cells activity (courtesy of M.J. Berry and O. Marre). The method shows a satisfying performance in the case of synthetic data. Real data analysis is not systematic, but instead used as an illustration and comparison with the paper of Schneidman *et al.* 2006 ([9]). As we could see in the example, the performance on real data, although satisfying, is affected by the large number of parameters in the distribution, a consequence

of the choice to work with canonical models (Ising, pairwise with memory). This effect is presumably not related to our method, but to a standard problem in statistics.

Some of our notations might not be usual to some readers. Therefore, we added a list of symbols at the end of the paper.

2. Gibbs Distributions in the Spatio-Temporal Domain

2.1. Spike Trains and Observables

2.1.1. Spike Trains

We consider the joint activity of N neurons, characterized by the emission of action potentials ("spikes"). We assume that there is a minimal time scale, δ, set to one without loss of generality, such that a neuron can at most fire a spike within a time window of size δ. This provides a time discretization labeled with an integer time, n. Each neuron activity is then characterized by a binary variable. We use the notation, ω, to differentiate our binary variables $\in \{ 0, 1 \}$ to the notation, σ or S, used for "spin" variables $\in \{ -1, 1 \}$. $\omega_k(n) = 1$ if neuron k fires at time n, and $\omega_k(n) = 0$, otherwise.

The state of the entire network in time bin n is thus described by a vector $\omega(n) \stackrel{\text{def}}{=} [\, \omega_k(n) \,]_{k=1}^{N}$, called a spiking pattern. A spike block is a consecutive sequence of spike patterns, $\omega_{n_1}^{n_2}$, representing the activity of the whole network between two instants, n_1 and n_2.

$$\omega_{n_1}^{n_2} = \{ \, \omega(n) \, \}_{\{ n_1 \leq n \leq n_2 \}} .$$

The time-range (or "range") of a block, $\omega_{n_1}^{n_2}$, is $n_2 - n_1 + 1$, the number of time steps from n_1 to n_2. Here is an example of a spike block with $N = 4$ neurons and range $R = 3$:

$$\begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

A spike train or raster is a spike block, ω_0^T, from some initial time, zero, to some final time, T. To alleviate the notations, we simply write ω for a spike train. We note Ω, the set of spike trains.

2.1.2. Observables

An observable is a function, $\mathcal{O}$, which associates a real number, $\mathcal{O}(\omega)$, to a spike train. In the realm of statistical physics, common examples of observables are the energy or the number of particles (where ω would correspond, e.g., to a spin configuration). In the context of neural networks examples are the number of neuron firing at a given time, n, $\sum_{k=1}^{N} \omega_k(n)$, or the function $\omega_{k_1}(n_1)\omega_{k_2}(n_2)$, which is one if neuron k_1 fires at time n_1 and neuron k_2 fires at time n_2, and is zero, otherwise.

Typically, an observable does not depend on the full raster, but only on a sub-block of it. The time-range (or "range") of an observable is the minimal integer $R > 0$, such that, for any raster, ω, $\mathcal{O}(\omega) = \mathcal{O}\left(\omega_0^{R-1}\right)$. The range of the observable $\sum_{k=1}^{N} \omega_k(n)$ is one; the range of $\omega_{k_1}(n_1)\omega_{k_2}(n_2)$ is $n_2 - n_1 + 1$. From now on, we restrict to observables of range R, fixed and finite. We set $D = R - 1$.

An observable is time-translation invariant if, for any time $n > 0$, we have $\mathcal{O}\left(\omega_n^{n+D}\right) \equiv \mathcal{O}\left(\omega_0^D\right)$ whenever $\omega_n^{n+D} = \omega_0^D$. The two examples above are time-translation invariant. The observable $\lambda(n_1)\omega_{k_1}(n_1)\omega_{k_2}(n_2)$, where λ is a real function of time, is not time-translation invariant. Basically, time-translation invariance means that $\mathcal{O}$ does not depend explicitly on time. We focus on such observables from now on.

2.1.3. Monomials

Prominent examples of time-translation invariant observables with range R are products of the form:

$$m_{p_1,\ldots,p_r}(\omega) \stackrel{\text{def}}{=} \prod_{u=1}^{r} \omega_{k_u}(n_u). \tag{1}$$

where p_u, $u = 1 \ldots r$ are pairs of spike-time events (k_u, n_u), $k_u = 1 \ldots N$ being the neuron index and $n_u = 0 \ldots D$ being the time index. Such an observable, called monomial, takes therefore values in $\{0, 1\}$ and is one, if and only if $\omega_{k_u}(n_u) = 1$, $u = 1 \ldots r$ (neuron k_1 fires at time $n_1, \ldots$; neuron k_r fires at time n_r). A monomial is therefore a binary observable that represents the logic-AND operator applied to a prescribed set of neuron spike events.

We allow the extension of the definition (1) to the case where the set of pairs $p_1, \ldots, p_r$ is empty, and we set $m_\emptyset = 1$. For a number, N, of neurons and a time range, R, there are, thus, 2^{NR} such possible products. Any observable of range R can be represented as a linear combination of products (1). Monomials constitute therefore a canonical basis for observable representation. To alleviate notations, instead of labeling monomials by a list of pairs, as in (1), we shall label them by an integer index, l.

2.1.4. Potential

Another prominent example of an observable is the function called "energy" or potential in the realm of the MaxEnt. Any potential of range R can be written as a linear combination of the 2^{NR} possible monomials (1):

$$\mathcal{H}_{\boldsymbol{\lambda}} = \sum_{l=1}^{2^{NR}} \lambda_l m_l, \tag{2}$$

where some coefficients, λ_l, in the expansion may be zero. Therefore, by analogy with spin systems, monomials somewhat constitute spatio-temporal interactions between neurons: the monomial $\prod_{u=1}^{r} \omega_{k_u}(n_u)$ contributes to the total energy, $\mathcal{H}_{\boldsymbol{\lambda}}(\omega)$, of the raster, ω, if and only if neuron k_1 fires at time n_1, ..., and neuron k_r fires at time n_r in the raster, ω. The number of pairs in a monomial (1) defines the degree of an interaction: Degree 1 corresponds to "self-interactions", Degree 2 to pairwise, and so on. Typical examples of such potentials are the Ising model [9,10,20]:

$$\mathcal{H}_{Ising}\left(\omega(0)\right) = \sum_i \lambda_i \omega_i(0) + \sum_{ij} \lambda_{ij} \omega_i(0)\omega_j(0), \tag{3}$$

where considered events are individual spikes and pairs of simultaneous spikes. Another example is the Ganmor–Schneidman–Segev (GSS) model [11,12]:

$$\mathcal{H}_{GSS}\left(\omega(0)\right) = \sum_i \lambda_i \omega_i(0) + \sum_{ij} \lambda_{ij} \omega_i(0)\omega_j(0) + \sum_{ijk} \lambda_{ijk} \omega_i(0)\omega_j(0)\omega_k(0), \tag{4}$$

where, additionally to 3, simultaneous triplets of spikes are considered (we restrict the form (4) to a triplet, although Ganmor *et al.* were also considering quadruplets). In these two examples, the potential is a function of the spike pattern at a given time. Here, we choose this time equal to zero, without loss of generality, since we are considering time-translation invariant potentials. More generally, the form (2) affords the consideration of spatio-temporal neurons interactions: this allows us to introduce delays, memory and causality in spike statistics estimation. A simple example is a pairwise model with delays, such as:

$$\mathcal{H}_{PR}\left(\omega_0^D\right) = \sum_i \lambda_i \omega_i(D) + \sum_{s=0}^{D} \sum_{ij} \lambda_{ij}^s \omega_i(0)\omega_j(s), \tag{5}$$

where 'PR' stands for 'pairwise with range R', which takes into account the events where neuron i fires s time steps after a neuron, j, with $s = 0 \ldots D$.

2.2. The Maximum Entropy Principle

Assigning equal probabilities (uniform probability distribution) to possible outcomes goes back to Laplace and Bernoulli ([21]) ("principle of insufficient reason"). Maximizing the statistical entropy without constraints is equivalent to this principle. In general, however, one has some knowledge about data, typically characterized by the empirical average of the prescribed observables (e.g., for spike trains, firing rates, the probability that a fixed group of neurons fire at the same time, the probability that K neurons fire at the same time [22]); this constitutes a set of constraints. The maximum entropy principle (MaxEnt) is a method to obtain, from the observation of a statistical sample, a probability distribution that approaches, at best, the statistics of the sample, taking into account these constraints without additional assumptions [8]. Maximizing the statistical entropy given those constraints provides a distribution as far as possible from the uniform one and as close as possible to the empirical distribution. For instance, considering the empirical mean and variance of the sample of a random variable as constraints results in a Gaussian distribution.

Although some attempts have been made to extend MaxEnt to non-stationary data [23–26], it is mostly applied in the context of stationary statistics: the average of an observable does not depend explicitly on time. We shall work with this hypothesis. In its simplest form, the MaxEnt also assumes that the sample has no memory: the probability of an outcome at time t does not depend on the past. We first discuss the MaxEnt in this context in the next section, before considering the case of processes with memory in Section 2.2.2.

2.2.1. Spatial Constraints

In our case, the natural constraints are represented by the empirical probability of occurrence of characteristic spike events in the spike train or, equivalently, by the average of specific monomials. Classical examples of constraints are the probability that a neuron fires at a given time (firing rate) or the probability that two neurons fire at the same time. For a raster, ω, of length T, we note $\pi_\omega^{(T)}$, the empirical distribution, and $\pi_\omega^{(T)}[\ \mathcal{O}\ ]$, the empirical average of the observable, $\mathcal{O}$, in the raster, ω. For example, the empirical firing rate of neuron i is $\pi_\omega^{(T)}[\ \omega_i\ ] = \frac{1}{T}\sum_{n=0}^{T-1}\omega_i(n)$; the empirical probability that two neurons, i, j, fire at the same time is $\pi_\omega^{(T)}[\ \omega_i\omega_j\ ] = \frac{1}{T}\sum_{n=0}^{T-1}\omega_i(n)\omega_j(n)$; and so on. Given a

set of L monomials, m_l, their empirical average, $\pi_\omega^{(T)}[\ m_l\ ]$, measured in the raster, ω, constitute a set of constraints shaping the sought for probability distribution. We consider here monomials corresponding to events occurring at the same time, *i.e.*, $m_l(\omega) \equiv m_l(\omega(0))$, postponing to Section 2.2.2 the general case of events occurring at distinct times.

In this context, the MaxEnt problems is stated as follows. Find a probability distribution, μ, that maximizes the entropy:

$$\mathcal{S}[\mu] = -\sum_{\omega(0)} \mu[\omega(0)] \log \mu[\omega(0)], \tag{6}$$

(where the sum holds on the 2^N possible spike patterns, $\omega(0)$), given the constraints:

$$\mu[m_l] = \pi_\omega^{(T)}[\ m_l\ ],\ l = 1 \dots L. \tag{7}$$

The average of monomials, predicted by the statistical model, μ (noted here as $\mu[m_l]$), must be equal to the average, $\pi_\omega^{(T)}[\ m_l\ ]$, measured in the sample. There is, additionally, the probability normalization constraint:

$$\sum_{\omega(0)} \mu[\omega(0)] = 1 \tag{8}$$

This provides a variational problem:

$$\mu = \arg\max_{\nu \in \mathcal{M}} \left[\mathcal{S}[\nu] + \lambda_0 \left(\sum_{\omega(0)} \nu[\omega(0)] - 1 \right) + \sum_{l=1}^{L} \lambda_l \left(\nu[m_l] - \pi_\omega^{(T)}[\ m_l\ ] \right) \right] \tag{9}$$

where $\mathcal{M}$ is the set of (stationary) probabilities on spike trains. One searches, among all stationary probabilities $\nu \in \mathcal{M}$, for the one which maximizes the right hand side of (9). There is a unique such probability, $\mu = \mu_{\boldsymbol{\lambda}}$, provided N is finite and $\lambda_l > -\infty$. This probability depends on the parameters, $\boldsymbol{\lambda}$.

Stated in this form, the MaxEnt is a Lagrange multipliers problem. The sought probability distribution is the classical Gibbs distribution:

$$\mu_{\boldsymbol{\lambda}}[\omega(0)] = \frac{1}{Z_{\boldsymbol{\lambda}}} e^{\mathcal{H}_{\boldsymbol{\lambda}}[\omega(0)]}, \tag{10}$$

where $Z_{\boldsymbol{\lambda}} = \sum_{\omega(0)} e^{\mathcal{H}_{\boldsymbol{\lambda}}[\omega(0)]}$ is the partition function, whereas $\mathcal{H}_{\boldsymbol{\lambda}}[\omega(0)] = \sum_{l=1}^{L} \lambda_l m_l[\omega(0)]$. Note that the time index (here, zero) does not play a role, since we have assumed $\mu_{\boldsymbol{\lambda}}$ to be stationary (time-translation invariant).

The value of λ_ls is fixed by the relation:

$$\mu_{\boldsymbol{\lambda}}(m_l) = \frac{\partial \log Z_{\boldsymbol{\lambda}}}{\partial \lambda_l} = \pi_\omega^{(T)}[\ m_l\ ],\ l = 1 \dots L. \tag{11}$$

Additionally, note that the matrix $\frac{\partial^2 \log Z_{\boldsymbol{\lambda}}}{\partial \lambda_l \partial \lambda_{l'}}$ is positive. This ensures the convexity of the problem and the uniqueness of the solution of the variational problem.

Note that we do not expect, in general, $\mu_{\boldsymbol{\lambda}}$ to be equal to the (hidden) probability shaping the observed sample. It is only the closest one satisfying the constraints (7) [7]. The notion of closeness is related to the Kullback-Leibler divergence, defined in the next section.

It is easy to check that the Gibbs distribution (10) obeys:

$$\mu_{\boldsymbol{\lambda}}\left[\,\omega_{n_1}^{n_2}\,\right] = \prod_{n=n_1}^{n_2} \mu_{\boldsymbol{\lambda}}\left[\,\omega(n)\,\right], \tag{12}$$

for any spike block, $\omega_{n_1}^{n_2}$. Indeed, the potential of the spike block, $\omega_{n_1}^{n_2}$, is $\mathcal{H}_{\boldsymbol{\lambda}}\left(\,\omega_{n_1}^{n_2}\,\right) = \sum_{n=n_1}^{n_2} \mathcal{H}_{\boldsymbol{\lambda}}\left(\,\omega(n)\,\right)$, whereas the partition function on spike blocks $\omega_{n_1}^{n_2}$ is $Z_{n_2-n_1} = \sum_{\omega_{n_1}^{n_2}} e^{\mathcal{H}_{\boldsymbol{\lambda}}\left[\,\omega_{n_1}^{n_2}\,\right]} = Z_{\boldsymbol{\lambda}}^{n_2-n_1}$. Equation (12) expresses that spiking patterns occurring at different times are independent under the Gibbs distribution (10). This is expected: since the constraints shaping $\mu_{\boldsymbol{\lambda}}$ take only into account spiking events occurring at the same time, we have no information on the causality between spike generation or on memory effects. The Gibbs distributions obtained when constructing constraints only with spatial events leads to statistical models where spike patterns are renewed at each time step, without reference to the past activity.

2.2.2. Spatio-Temporal Constraints

On the opposite side, one expects that spike train generation involves causal interactions between neurons and memory effects. We would therefore like to construct Gibbs distributions taking into account information on the spatio-temporal interactions between neurons and leading to a statistical model, not assuming anymore that successive spikes patterns are independent. Although the notion of the Gibbs distribution extends to processes with infinite memory [27], we shall concentrate here on Gibbs distributions associated with Markov processes with finite memory depth D; that is, the probability of having a spike pattern, $\omega(n)$, at time n, given the past history of spikes reads $P\left[\,\omega(n)\,\middle|\,\omega_{n-D}^{n-1}\,\right]$. Note that those transition probabilities are assumed not to depend explicitly on time (stationarity assumption).

Such a family of transition probabilities, $P\left[\,\omega(n)\,\middle|\,\omega_{n-D}^{n-1}\,\right]$, defines a homogeneous Markov chain. Provided $P\left[\,\omega(n)\,\middle|\,\omega_{n-D}^{n-1}\,\right] > 0$ (this is a sufficient, but not a necessary condition. In the remainder of the paper, we shall work with this assumption) for all ω_{n-D}^{n}, there is a unique probability, μ, called the invariant probability of the Markov chain, such that:

$$\mu\left[\,\omega_1^D\,\right] = \sum_{\omega_0^{D-1}} P\left[\,\omega(D)\,\middle|\,\omega_0^{D-1}\,\right]\,\mu\left[\,\omega_0^{D-1}\,\right]. \tag{13}$$

In a Markov process, the probability of a block, $\omega_{n_1}^{n_2}$, for $n_2 - n_1 + 1 > D$, is:

$$\mu\left[\,\omega_{n_1}^{n_2}\,\right] = \prod_{n=n_1+D}^{n_2} P\left[\,\omega(n)\,\middle|\,\omega_{n-D}^{n-1}\,\right]\,\mu\left[\,\omega_{n_1}^{n_1+D-1}\,\right], \tag{14}$$

the Chapman–Kolmogorov relation [28]. To determine the probability of $\omega_{n_1}^{n_2}$, one has to know the transition probabilities and the probability, $\mu\left[\,\omega_{n_1}^{n_1+D-1}\,\right]$. When attempting to construct a Gibbs distribution obeying (14) from a set of spatio-temporal constraints, one has therefore to determine simultaneously the family of transition probabilities and the invariant probability. Remark that setting:

$$\phi(\omega_0^D) = \log P\left[\,\omega(D)\,\middle|\,\omega_0^{D-1}\,\right], \tag{15}$$

we may write (14) in the form:

$$\mu\left[\,\omega_{n_1}^{n_2}\,\middle|\,\omega_{n_1}^{n_1+D-1}\,\right] = e^{\sum_{n=n_1+D}^{n_2}\phi(\omega_n^{n+D})}. \tag{16}$$

The probability of observing the spike pattern, $\omega_{n_1}^{n_2}$, given the past $\omega_{n_1}^{n_1+D-1}$ of depth D has an exponential form, similar to (10). Actually, the invariant probability of a Markov chain is a Gibbs distribution in the following sense.

In view of (14), probabilities must be defined as whatever, even if $n_2 - n_1$ is arbitrarily large. In this setting, the right objects are probabilities on infinite rasters [28]. Then, the entropy rate (or Kolmogorov–Sinai entropy) of μ is:

$$\mathcal{S}\left[\,\mu\,\right] \,=\, -\,\limsup_{n\to\infty}\frac{1}{n+1}\sum_{\omega_0^n}\mu\left[\,\omega_0^n\,\right]\,\log\mu\left[\,\omega_0^n\,\right], \tag{17}$$

where the sum holds over all possible blocks, ω_0^n. This reduces to (6) when μ obeys (12).

The MaxEnt takes now the following form. We consider a set of L spatio-temporal spike events (monomials), whose empirical average value, $\pi_\omega^{(T)}\left[\,m_l\,\right]$, has been computed. We only restrict to monomials with a range at most equal to $R = D+1$, for some $D > 0$. This provide us a set of constraints of the form (7). To maximize the entropy rate (17) under the constraints (7), we construct a range-R potential $\mathcal{H}_{\boldsymbol{\lambda}} = \sum_{l=1}^{L}\lambda_l m_l$. The generalized form of the MaxEnt states that there is a unique probability measure $\mu_{\boldsymbol{\lambda}} \in \mathcal{M}$, such that [29]:

$$\mathcal{P}\left[\,\boldsymbol{\lambda}\,\right] = \sup_{\nu\in\mathcal{M}}\left(\,\mathcal{S}\left[\,\nu\,\right] \,+\, \nu\left[\,\mathcal{H}_{\boldsymbol{\lambda}}\right]\,\right) = \mathcal{S}\left[\,\mu_{\boldsymbol{\lambda}}\,\right] \,+\, \mu_{\boldsymbol{\lambda}}\left[\,\mathcal{H}_{\boldsymbol{\lambda}}\,\right]. \tag{18}$$

This is the extension of the variational principle (9) to Markov chains. It selects, among all possible probability, ν, a unique probability, $\mu_{\boldsymbol{\lambda}}$, which realizes the supremum. $\mu_{\boldsymbol{\lambda}}$ is called the Gibbs distribution with potential $\mathcal{H}_{\boldsymbol{\lambda}}$.

The quantity, $\mathcal{P}\left[\,\boldsymbol{\lambda}\,\right]$, is called topological pressure or free energy density. For a potential of the form (2) [30,31]:

$$\frac{\partial\mathcal{P}\left[\,\boldsymbol{\lambda}\,\right]}{\partial\lambda_l} = \mu_{\boldsymbol{\lambda}}\left[\,m_l\,\right]. \tag{19}$$

This is the analog of (11), which allows one to tune the parameters, λ_l. Thus, $\mathcal{P}\left[\,\boldsymbol{\lambda}\,\right]$ plays the role of $\log Z_{\boldsymbol{\lambda}}$ in (10). Actually, it is equal to $\log Z_{\boldsymbol{\lambda}}$ when restricting to the memoryless case (In statistical physics, the free energy is $-kT\,\log Z$. The minus sign comes from the minus sign in the Hamiltonian). $\mathcal{P}\left[\,\boldsymbol{\lambda}\,\right]$ is strictly convex thanks to the assumption $P\left[\,\omega(n)\,\middle|\,\omega_{n-1}^{n-D}\,\right] > 0$, which guarantees the uniqueness of $\mu_{\boldsymbol{\lambda}}$.

Note that $\mu_{\boldsymbol{\lambda}}$ has not the form (10) for $D > 0$. Indeed a probability distribution, e.g., of the form $\mu_{\boldsymbol{\lambda}}(\omega_0^{n-1}) = \frac{1}{Z_n}e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}$ with:

$$\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1}) \equiv \sum_{r=0}^{n-D-1}\mathcal{H}_{\boldsymbol{\lambda}}(\omega_r^{r+D}) = \sum_l \lambda_l \sum_{r=0}^{n-D-1} m_l(\omega_r^{r+D}), \tag{20}$$

the potential of the block ω_0^{n-1}, and:

$$Z_n\left[\,\boldsymbol{\lambda}\,\right] = \sum_{\omega_0^{n-1}} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}, \tag{21}$$

the "n-time steps" partition function does not obey the Chapman–Kolmogorov relation (14).

However, the following holds [29,32–34].

1. There exist $A, B > 0$, such that, for any block, ω_0^{n-1}:

$$A \leq \frac{\mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right]}{e^{-(n-D)\mathcal{P}[\boldsymbol{\lambda}]}e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}} \leq B. \tag{22}$$

2. We have:

$$\mathcal{P}\left[\boldsymbol{\lambda}\right] = \lim_{n\to\infty} \frac{1}{n} \log Z_n\left[\boldsymbol{\lambda}\right]. \tag{23}$$

In the spatial case, $Z_n\left[\boldsymbol{\lambda}\right] = Z^n\left[\boldsymbol{\lambda}\right]$ and $\mathcal{P}\left[\boldsymbol{\lambda}\right] = \log Z\left[\boldsymbol{\lambda}\right]$, whereas $A = B = 1$ in (22). Although (23) is defined by a limit, it is possible to compute $\mathcal{P}\left[\boldsymbol{\lambda}\right]$ as the log of the largest eigenvalue of a transition matrix constructed from $\mathcal{H}_{\boldsymbol{\lambda}}$ (Perron–Frobenius matrix) [35]. Unfortunately, this method does not apply numerically as soon as $NR > 20$.

These relations are crucial for the developments made in the next section.

To recap, a Gibbs distribution in the sense of [18] is the invariant probability distribution of a Markov chain. The link between the potential $\mathcal{H}_{\boldsymbol{\lambda}}$ and the transition probabilities $P\left[\omega(D) \,\middle|\, \omega_0^{D-1}\right]$ (respectively, the potential [15]) is given by: $\phi(\omega_0^D) = \mathcal{H}(\omega_0^D) - \mathcal{G}(\omega_0^D)$, where $\mathcal{G}$, called a normalization function, is a function of the right eigenvector of a transition matrix built from $\mathcal{H}$, and a function of $P[\boldsymbol{\lambda}]$. $\mathcal{G}$ reduces to $\log Z_{\boldsymbol{\lambda}} = \mathcal{P}\left[\boldsymbol{\lambda}\right]$ when $D = 0$ [2].

To finish this section, let us introduce the Kullback-Leibler divergence, $d_{KL}(\nu, \mu)$, which provides a notion of similarity between two probabilities, ν, μ. We have $d_{KL}(\nu, \mu) \geq 0$ with equality, if and only if $\mu = \nu$. The Kullback-Leibler divergence between an invariant probability $\nu \in \mathcal{M}$ and the Gibbs distribution, $\mu_{\boldsymbol{\lambda}}$, with potential $\mathcal{H}_{\boldsymbol{\lambda}}$ is given by $d_{KL}\left(\nu, \mu_{\boldsymbol{\lambda}}\right) = \mathcal{P}\left[\boldsymbol{\lambda}\right] - \nu\left[\mathcal{H}_{\boldsymbol{\lambda}}\right] - \mathcal{S}\left[\nu\right]$, [29]. When $\nu = \pi_\omega^{(T)}$, we obtain the divergence between the "model ($\mu_{\boldsymbol{\lambda}}$)" and the "empirical probability ($\pi_\omega^{(T)}$)":

$$d_{KL}\left(\pi_\omega^{(T)}, \mu_{\boldsymbol{\lambda}}\right) = \mathcal{P}\left[\boldsymbol{\lambda}\right] - \pi_\omega^{(T)}\left[\mathcal{H}_{\boldsymbol{\lambda}}\right] - \mathcal{S}\left[\pi_\omega^{(T)}\right]. \tag{24}$$

3. Inferring the Coefficients of a Potential from Data

Equation (11) or (19) provides an analytical way to compute the coefficients of the Gibbs distribution from data. However, they require the computation of the partition function or of the topological pressure, which becomes rapidly intractable as the number of neurons increases. Thus, researchers have attempted to find alternative methods to compute reliably and efficiently the λ_ls. An efficient method has been introduced in [17] and applied to spike trains in [18]. Although these papers are restricted to Gibbs distributions of the form (10) (models without memory), we show in this section how their method can be extended to general Gibbs distributions.

3.1. Bounding the Kullback-Leibler Divergence Variation

3.1.1. The Spatial Case

The method developed in [17] by Dudik *et al.* is based on the so-called convex duality principle, used in mathematical optimization theory. Due to the difficulty in maximizing the entropy (which is a

concave function), one looks for a convex function that easier to investigate. Dudik *et al.* showed that, for spatially constrained MaxEnt distributions, finding the Gibbs distribution amounts to finding the minimum of the negative log likelihood (we have adapted [17] to our notations. Moreover, in our case, $\pi_\omega^{(T)}$ corresponds to the empirical average on a raster, ω, whereas π in [17] corresponds to an average over independent samples):

$$L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda}) = -\pi_\omega^{(T)} [\, \log \mu_{\boldsymbol{\lambda}} \,]. \tag{25}$$

Indeed, in the spatial case, the Kullback-Leibler divergence between the empirical measure, $\pi_\omega^{(T)}$, and the Gibbs distribution at $\mu_{\boldsymbol{\lambda}}$ is:

$$d_{KL}(\pi_\omega^{(T)}, \mu_{\boldsymbol{\lambda}}) = \pi_\omega^{(T)} \left[\frac{\log \pi_\omega^{(T)}}{\log \mu_{\boldsymbol{\lambda}}} \right] = \pi_\omega^{(T)} [\, \log \pi_\omega^{(T)} \,] - \pi_\omega^{(T)} [\, \log \mu_{\boldsymbol{\lambda}} \,], \tag{26}$$

so that, from (24):

$$L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda}) = \mathcal{P}[\boldsymbol{\lambda}] - \pi_\omega^{(T)} [\, \mathcal{H}_{\boldsymbol{\lambda}} \,],$$

where we used $\mathcal{S}\left[\pi_\omega^{(T)}\right] = -\pi_\omega^{(T)}\left[\log(\pi_\omega^{(T)})\right]$.

Since $\mathcal{P}$ is convex and $\pi_\omega^{(T)} [\, \mathcal{H}_{\boldsymbol{\lambda}} \,]$ linear in $\boldsymbol{\lambda}$, $L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda})$ is convex. Its unique minimum is given by (11).

Moreover, we have:

$$L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda}') - L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda}) = \mathcal{P}[\boldsymbol{\lambda}'] - \mathcal{P}[\boldsymbol{\lambda}] - \pi_\omega^{(T)} [\, \Delta\mathcal{H}_{\boldsymbol{\lambda}} \,], \tag{27}$$

with $\Delta\mathcal{H}_{\boldsymbol{\lambda}} = \mathcal{H}_{\boldsymbol{\lambda}'} - \mathcal{H}_{\boldsymbol{\lambda}}$. From (10):

$$\begin{aligned} \frac{Z[\boldsymbol{\lambda}']}{Z[\boldsymbol{\lambda}]} &= \frac{1}{Z[\boldsymbol{\lambda}]} \sum_{\omega(0)} e^{\mathcal{H}_{\boldsymbol{\lambda}'}(\omega(0))} \\ &= \sum_{\omega(0)} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega(0))} \mu_{\boldsymbol{\lambda}}[\omega(0))] \\ &= \mu_{\boldsymbol{\lambda}}\left[e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}} \right], \end{aligned} \tag{28}$$

and since $P[\boldsymbol{\lambda}] = \log Z[\boldsymbol{\lambda}]$ in the spatial case:

$$\mathcal{P}[\boldsymbol{\lambda}'] - \mathcal{P}[\boldsymbol{\lambda}] = \log \mu_{\boldsymbol{\lambda}}\left[e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}} \right]. \tag{29}$$

Therefore:

$$L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda}') - L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda}) = \log \mu_{\boldsymbol{\lambda}}\left[e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}} \right] - \pi_\omega^{(T)} [\, \Delta\mathcal{H}_{\boldsymbol{\lambda}} \,]. \tag{30}$$

The idea proposed by Dudik *et al.* is then to bound this difference by an easier-to-compute convex quantity, with the same minimum as $L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda})$, and to reach this minimum by iterations on $\boldsymbol{\lambda}$. They proposed a sequential and a parallel method. Let us summarize first the sequential method. The goal here is not to rewrite their paper [17], but to explain some crucial elements that are not directly appliable to the spatio-temporal case.

In the sequential case, one updates $\boldsymbol{\lambda}$ as $\boldsymbol{\lambda}' = \boldsymbol{\lambda} + \delta\boldsymbol{e}_l$, for some l, where $\boldsymbol{e}_l$ is the canonical basis vector in direction l, so that $\Delta\mathcal{H}_{\boldsymbol{\lambda}} = \delta m_l$, and:

$$L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda}') - L_{\pi_\omega^{(T)}}(\boldsymbol{\lambda}) = \log \mu_{\boldsymbol{\lambda}}\left[e^{\delta m_l} \right] - \delta\pi_\omega^{(T)} [\, m_l \,].$$

Using the following property:

$$e^{\delta x} \leq 1 + (e^{\delta} - 1)x, \tag{31}$$

for $x \in [0, 1]$, and since $m_l \in \{0, 1\}$, we have:

$$\log \mu_{\boldsymbol{\lambda}} \left[e^{\delta m_l} \right] \leq \log \left(1 + (e^{\delta} - 1)\mu_{\boldsymbol{\lambda}}[m_l] \right). \tag{32}$$

This bound, proposed by Dudik *et al.*, is remarkably clever. Indeed, it replaces the computation of the average $\mu_{\boldsymbol{\lambda}} \left[e^{\delta m_l} \right]$, which is computationally hard, by the computation of $\mu_{\boldsymbol{\lambda}} \left[m_l \right]$, which is computationally easy. Finally,

$$L_{\pi_{\omega}^{(T)}}(\boldsymbol{\lambda}') - L_{\pi_{\omega}^{(T)}}(\boldsymbol{\lambda}) \leq -\delta \pi_{\omega}^{(T)} \left[\, m_l \, \right] + \log \left(1 + (e^{\delta} - 1)\mu_{\boldsymbol{\lambda}} \left[\mathbf{m}_l \right] \right). \tag{33}$$

In the parallel case, the computation and results differ. One now updates $\boldsymbol{\lambda}$ as $\boldsymbol{\lambda}' = \boldsymbol{\lambda} + \sum_{l=1}^{L} \delta_l \boldsymbol{e}_l$. Moreover, one has to renormalize the m_ls in $m'_l = \frac{m_l}{L}$ in order that Equation (34) below holds. We have, therefore, $\Delta \mathcal{H}_{\boldsymbol{\lambda}} = \sum_{l=1}^{L} \delta_l m'_l$.

Thus,

$$L_{\pi_{\omega}^{(T)}}(\boldsymbol{\lambda}') - L_{\pi_{\omega}^{(T)}}(\boldsymbol{\lambda}) = \log \mu_{\boldsymbol{\lambda}} \left[e^{\sum_{l=1}^{L} \delta_l m'_l} \right] - \sum_{l=1}^{L} \delta_l \pi_{\omega}^{(T)} \left[\, m'_l \, \right].$$

Using the following property [36]:

$$e^{\sum_{l=1}^{L} \delta_l m'_l} \leq 1 + \sum_{l=1}^{L} m'_l \left(e^{\delta_l} - 1 \right), \tag{34}$$

for $\delta_l \in \mathbb{R}$ and $m'_l \geq 0$, $\sum_{l=1}^{L} m'_l \leq 1$, we have:

$$\log \mu_{\boldsymbol{\lambda}} \left[e^{\sum_{l=1}^{L} \delta_l m'_l} \right] \leq \log \left(1 + \sum_{l=1}^{L} \left(e^{\delta_l} - 1 \right) \mu_{\boldsymbol{\lambda}}[m'_l] \right).$$

Since $\log(1 + x) \leq x$ for $x > -1$, Dudick *et al.* obtain:

$$\log \mu_{\boldsymbol{\lambda}} \left[e^{\sum_{l=1}^{L} \delta_l m'_l} \right] \leq \sum_{l=1}^{L} \left(e^{\delta_l} - 1 \right) \mu_{\boldsymbol{\lambda}}[m'_l],$$

provided $\sum_{l=1}^{L} \left(e^{\delta_l} - 1 \right) \mu_{\boldsymbol{\lambda}}[m'_l] > -1$. (this constraint has to be checked during iterations). Finally, using the definition of m'_l:

$$L_{\pi_{\omega}^{(T)}}(\boldsymbol{\lambda}') - L_{\pi_{\omega}^{(T)}}(\boldsymbol{\lambda}) \leq \frac{1}{L} \left[-\sum_{l=1}^{L} \delta_l \pi_{\omega}^{(T)} \left[\, m_l \, \right] + \sum_{l=1}^{L} \left(e^{\delta_l} - 1 \right) \mu_{\boldsymbol{\lambda}}[m_l] \right]. \tag{35}$$

To be complete, let us mention that Dudik *et al.* consider the case where some error, ϵ_l, is allowed in the estimation of the coefficient, λ_l. This relaxation on the parameters alleviates the overfitting.

In this case, the bound on the right-hand side in (33) (sequential case) becomes:

$$F_l(\boldsymbol{\lambda}, \delta) = -\delta \pi_{\omega}^{(T)} \left[\, m_l \, \right] + \log \left(1 + (e^{\delta} - 1)\mu_{\boldsymbol{\lambda}} \left[m_l \right] \right) + \epsilon_l \left(\, | \, \lambda_l + \delta \, | - | \, \lambda_l \, | \, \right). \tag{36}$$

whereas the right-hand side in (35) becomes $\sum_{l=1}^{L} G_l(\boldsymbol{\lambda}, \boldsymbol{\delta})$ with:

$$G_l(\boldsymbol{\lambda}, \boldsymbol{\delta}) = \frac{1}{L}\left[-\delta_l \pi_\omega^{(T)}[\, m_l \,] + \left(e^{\delta_l} - 1 \right) \mu_{\boldsymbol{\lambda}}[m_l] \right] + \epsilon_l \left(|\, \lambda_l + \delta \,| - |\, \lambda_l \,| \right), \tag{37}$$

The minimum of these functions is easy to find, and one obtains, for a given $\boldsymbol{\lambda}$, the variation, $\boldsymbol{\delta}$, required to lower bound the log-likelihood variation. The authors have shown that both the sequential and parallel method produce a sequence, $\boldsymbol{\lambda}^{(k)}$, which converges to the minimum of $L_{\pi_\omega^{(T)}}$ as $k \to +\infty$. Note, however, that one strong condition in their convergence theorem is $\epsilon_l > 0$. This requires a sharp estimate of the error, ϵ_l, which cannot be solely based on the central limit theorem or on Hoeffding inequality in our case, because when the empirical average, $\pi_\omega^{(T)}(m_l)$, is too small, the minima of F, computed in [18], may not be defined.

3.1.2. Extension to the Spatio-Temporal Case

We now show how to extend these computations to the spatio-temporal case, provided one replaces the log-likelihood, $L_{\pi_\omega^{(T)}}$, by the Kullback-Leibler divergence (24). The main obstacle is that the Gibbs distribution does not have the form, $\frac{e^{\mathcal{H}}}{Z}$. We obtain, thus, a convex criterion to minimize Kullback-Leibler divergence variation, hence reaching the minimum, $\pi_\omega^{(T)}$.

Replacing ν in Equation (24) by $\pi_\omega^{(T)}$, the empirical measure, one has:

$$d_{KL}(\pi_\omega^{(T)}, \mu_{\boldsymbol{\lambda}'}) - d_{KL}(\pi_\omega^{(T)}, \mu_{\boldsymbol{\lambda}}) = \mathcal{P}\left[\boldsymbol{\lambda}'\right] - \mathcal{P}\left[\boldsymbol{\lambda}\right] - \pi_\omega^{(T)}\left[\, \Delta\mathcal{H}_{\boldsymbol{\lambda}} \,\right], \tag{38}$$

because the entropy, $\mathcal{S}\left[\pi_\omega^{(T)}\right]$, cancels. This is the analog of (27). The main problem now is to compute $\mathcal{P}\left[\boldsymbol{\lambda}'\right] - \mathcal{P}\left[\boldsymbol{\lambda}\right]$.

From (22), we have:

$$\begin{aligned}
&Ae^{-(n-D)\mathcal{P}[\boldsymbol{\lambda}]} \sum_{\omega_0^{n-1}} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \\
&\leq \sum_{\omega_0^{n-1}} \mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right] e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \\
&\leq Be^{-(n-D)\mathcal{P}[\boldsymbol{\lambda}]} \sum_{\omega_0^{n-1}} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}
\end{aligned}$$

so that:

$$\begin{aligned}
&\lim_{n\to\infty} \frac{1}{n}\left[\log A - (n-D)\mathcal{P}\left[\boldsymbol{\lambda}\right] + \log\left(\sum_{\omega_0^{n-1}} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \right) \right] \\
&\leq \lim_{n\to\infty} \frac{1}{n} \log\left(\sum_{\omega_0^{n-1}} \mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right] e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \right) \\
&\leq \lim_{n\to\infty} \frac{1}{n}\left[\log B - (n-D)\mathcal{P}\left[\boldsymbol{\lambda}\right] + \log\left(\sum_{\omega_0^{n-1}} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \right) \right].
\end{aligned} \tag{39}$$

Since $\mathcal{H}_{\boldsymbol{\lambda}'}(\omega_0^{n-1}) = \mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1}) + \Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})$, from (23):

$$\lim_{n\to\infty} \frac{1}{n} \log \sum_{\omega_0^{n-1}} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} = \mathcal{P}\left[\boldsymbol{\lambda}'\right].$$

Therefore:

$$\mathcal{P}\left[\boldsymbol{\lambda}'\right]-\mathcal{P}\left[\boldsymbol{\lambda}\right]=\lim_{n\to\infty}\frac{1}{n}\log\sum_{\omega_0^{n-1}}\mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right]e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}. \tag{40}$$

This is the extension of (29) to the spatio-temporal case. In the spatial case, it reduces to (29) from (12). This equation is obviously numerically intractable, but it has two advantages: on the one hand, it allows one to extend the bounds, (33) (sequential case) and (35) (parallel case), and on the other hand, it can be used to get a $\boldsymbol{\delta}$-power expansion of $\mathcal{P}\left[\boldsymbol{\lambda}'\right]-\mathcal{P}\left[\boldsymbol{\lambda}\right]$. This last point is used in Section 3.2.3.

To get the analog of (33) in the sequential case where $\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})=\delta\sum_{r=0}^{n-D-1}m_l(\omega_r^{r+D})$, one may still apply (31) which holds, provided:

$$m_l(\omega_0^{n-1})\equiv\sum_{r=0}^{n-1-D}m_l(\omega_r^{r+D})<1 \tag{41}$$

Therefore, compared to the spatial, we have to replace m_l by $\frac{m_l}{n-D}$ in $\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})$. We have, therefore:

$$\begin{aligned}\sum_{\omega_0^{n-1}}\mu_{\boldsymbol{\lambda}}[\omega_0^{n-1}]e^{\Delta\mathcal{H}(\omega_0^{n-1})} &= \sum_{\omega_0^{n-1}}\mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right]e^{\delta\frac{1}{n-D}m_l(\omega_0^{n-1})}\\ &\leq 1+(e^{\delta}-1)\frac{1}{n-D}\sum_{\omega_0^{n-1}}\mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right]m_l(\omega_0^{n-1}).\end{aligned}$$

From the time translation invariance of $\mu_{\boldsymbol{\lambda}}$, we have:

$$\begin{aligned}\frac{1}{n-D}\sum_{\omega_0^{n-1}}\mu_{\boldsymbol{\lambda}}[\omega_0^{n-1}]m_l(\omega_0^{n-1}) &= \frac{1}{n-D}\sum_{r=0}^{n-D-1}\sum_{\omega_0^{n-1}}\mu_{\boldsymbol{\lambda}}[\omega_0^{n-1}]m_l(\omega_r^{r+D})\\ &= \frac{1}{n-D}\sum_{r=0}^{n-D-1}\mu_{\boldsymbol{\lambda}}[m_l]\\ &= \mu_{\boldsymbol{\lambda}}[m_l]\end{aligned}$$

so that:

$$\sum_{\omega_0^{n-1}}\mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right]e^{\delta\frac{1}{n-D}m_l(\omega_0^{n-1})}\leq 1+(e^{\delta}-1)\mu_{\boldsymbol{\lambda}}\left[m_l\right].$$

At first glance this bound is not really useful. Indeed, from (40), we obtain:

$$\mathcal{P}\left[\boldsymbol{\lambda}'\right]-\mathcal{P}\left[\boldsymbol{\lambda}\right]\leq\lim_{n\to\infty}\frac{1}{n}\log\left(1+(e^{\delta}-1)\mu_{\boldsymbol{\lambda}}\left[m_l\right]\right)=0.$$

Since this holds for any δ, this implies $\mathcal{P}\left[\boldsymbol{\lambda}'\right]=\mathcal{P}\left[\boldsymbol{\lambda}\right]$. The reason for this is evident. Renormalizing m_l, as we did to match the condition imposed by the bound (31), is equivalent to renormalizing δ by $\frac{\delta}{n-D}$. As $n\to+\infty$, this perturbation tends to zero and $\boldsymbol{\lambda}'=\boldsymbol{\lambda}$. Therefore, the clever bound (31) would here be of no interest if we were seeking exact results. However, the goal here is to propose a numerical scheme, where, obviously, n is finite. We replace, therefore, the limit $n\to+\infty$ by a fixed n in the computation of $\mathcal{P}\left[\boldsymbol{\lambda}'\right]-\mathcal{P}\left[\boldsymbol{\lambda}\right]$. Keeping in mind that m_l must also be renormalized in $\pi_{\omega}^{(T)}\left[\,\Delta\mathcal{H}_{\boldsymbol{\lambda}}\,\right]$ and using $\frac{1}{n}<\frac{1}{n-D}$, the Kullback-Leibler divergence (38) obeys:

$$d_{KL}(\pi_{\omega}^{(T)},\mu_{\boldsymbol{\lambda}'})-d_{KL}(\pi_{\omega}^{(T)},\mu_{\boldsymbol{\lambda}})\leq\frac{1}{n-D}\left[-\delta\pi_{\omega}^{(T)}\left[\,m_l\,\right]+\log\left(1+(e^{\delta}-1)\mu_{\boldsymbol{\lambda}}\left[m_l\right]\right)\right], \tag{42}$$

the analog of (33).

In the parallel case, similar remarks hold. In order to apply the bound (34), we have to renormalize the m_ls in $m'_l = \frac{1}{L(n-D)}$. As for the spatial case, we also need to check that $\sum_{l=1}^{L} \left(e^{\delta_l} - 1 \right) \mu_{\boldsymbol{\lambda}}[m'_l] > -1$. (this constraint is not a guarantee and has to be checked during iterations). One obtains finally:

$$d_{KL}(\pi_\omega^{(T)}, \mu_{\boldsymbol{\lambda}'}) - d_{KL}(\pi_\omega^{(T)}, \mu_{\boldsymbol{\lambda}}) \leq \frac{1}{L(n-D)} \left[-\sum_{l=1}^{L} \delta_l \pi_\omega^{(T)} [\, m_l \,] + \sum_{l=1}^{L} \left(e^{\delta_l} - 1 \right) \mu_{\boldsymbol{\lambda}}[m_l] \right], \quad (43)$$

the analog of (35).

Compared with the spatial case, we see, therefore, that n must not be too large to have a reasonable Kullback-Leibler divergence variation. It must not be too small, however, to get a good approximation of the empirical averages.

3.2. Updating the Target Distribution when the Parameters Change

When updating the parameters, $\boldsymbol{\lambda}$, one has to compute again the average values, $\mu_{\boldsymbol{\lambda}}[\, m_l \,]$, since the probability, $\mu_{\boldsymbol{\lambda}}$, has changed. This has a huge computational cost. The exact computation (e.g., from (11, 19)) is not tractable for large N, so approximate methods have to be used, like Monte Carlo [19]. Again, this is also CPU time consuming, especially if one recomputes it again at each iteration, but at least it is tractable.

In this spirit, Broderick *et al.* [18] propose generating a Monte Carlo raster distributed according to $\mu_{\boldsymbol{\lambda}}$ and to use it to compute $\mu_{\boldsymbol{\lambda}'}$ when $\|\boldsymbol{\lambda}' - \boldsymbol{\lambda}\|$ is sufficiently small. We explain their method, limited to the spatial case, in the next section, and we explain why it is not applicable in the spatio-temporal case. We then propose an alternative method.

3.2.1. The Spatial Case

The average of m_l is obtained by the derivative of the topological pressure, $\mathcal{P}[\, \boldsymbol{\lambda} \,]$. In the spatial case, where $\mathcal{P}(\boldsymbol{\lambda}) = \log Z_{\boldsymbol{\lambda}}$, we have:

$$\begin{aligned} \mu_{\boldsymbol{\lambda}'}[\, m_l \,] &= \frac{\partial \mathcal{P}(\boldsymbol{\lambda}')}{\partial \lambda'_j} \\ &= \frac{1}{Z[\, \boldsymbol{\lambda}' \,]} \sum_{\omega(0)} m_l(\omega(0)) e^{\mathcal{H}_{\boldsymbol{\lambda}'}(\omega(0))} \\ &= \frac{Z[\, \boldsymbol{\lambda} \,]}{Z[\, \boldsymbol{\lambda}' \,]} \sum_{\omega(0)} m_l(\omega(0)) e^{\Delta \mathcal{H}_{\boldsymbol{\lambda}}(\omega(0))} \mu_{\boldsymbol{\lambda}}[\, \omega(0) \,] \end{aligned} \quad (44)$$

Using (28), one finally obtains:

$$\mu_{\boldsymbol{\lambda}'}[\, m_l \,] = \frac{\mu_{\boldsymbol{\lambda}}\left[m_l(\omega(0))\, e^{\Delta \mathcal{H}_{\boldsymbol{\lambda}}(\omega(0))} \right]}{\mu_{\boldsymbol{\lambda}}\left[e^{\Delta \mathcal{H}_{\boldsymbol{\lambda}}(\omega(0))} \right]}, \quad (45)$$

which is Equation (18) in [18]. Using this formula, one is able to compute the average of m_l with respect to the new probability, $\mu_{\boldsymbol{\lambda}'}$, only using the old one, $\mu_{\boldsymbol{\lambda}}$.

3.2.2. Extension to the Spatio-Temporal Case

We now explain why the Broderick *et al.* method does not extend to the spatio-temporal case. The main problem is that if one tries to obtain the analog of the equality (45), one obtains, in fact, an inequality:

$$\frac{A}{B}\mu_{\boldsymbol{\lambda}'}\left[\, m_l \,\right] \leq \lim_{n\to\infty} \frac{1}{n} \frac{\mu_{\boldsymbol{\lambda}}\left[m_l\left(\, \omega_0^{n-1} \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \right]}{\mu_{\boldsymbol{\lambda}}\left[e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \right]} \leq \frac{B}{A}\mu_{\boldsymbol{\lambda}'}\left[\, m_l \,\right], \tag{46}$$

where A, B are the constants in (22). They are not known in general (they depend on the potential), and they are different. However, in the spatial case $A = B = 1$, whereas $\mu_{\boldsymbol{\lambda}}\left[m_l\left(\, \omega_0^{n-1} \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \right] = \mu_{\boldsymbol{\lambda}}\left[m_l\left(\, \omega(0) \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega(0))} \right]$, because the potential has range one. Then, one recovers (45). Let us now explain how we obtain (46).

The averages of quantities are obtained by the derivative of the topological pressure (Equation (19)). We have:

$$\mu_{\boldsymbol{\lambda}'}\left[\, m_l \,\right] = \frac{\partial \mathcal{P}}{\partial \lambda_l'} = \frac{\partial \lim_{n\to\infty} \frac{1}{n} \log Z_n\left[\, \boldsymbol{\lambda}' \,\right]}{\partial \lambda_l'}. \tag{47}$$

Assuming that the limit and the derivative commute (see, e.g., [37]), gives:

$$\begin{aligned}
\mu_{\boldsymbol{\lambda}'}\left[\, m_l \,\right] &= \lim_{n\to\infty} \frac{1}{n} \frac{1}{Z_n\left[\, \boldsymbol{\lambda}' \,\right]} \sum_{\omega_0^{n-1}} m_l\left(\, \omega_0^{n-1} \,\right) e^{\mathcal{H}_{\boldsymbol{\lambda}'}(\omega_0^{n-1})} \\
&= \lim_{n\to\infty} \frac{1}{n} \frac{1}{Z_n\left[\, \boldsymbol{\lambda}' \,\right]} \sum_{\omega_0^{n-1}} m_l\left(\, \omega_0^{n-1} \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \\
&= \lim_{n\to\infty} \frac{1}{n} \frac{\sum_{\omega_0^{n-1}} m_l\left(\, \omega_0^{n-1} \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}}{\sum_{\omega_0^{n-1}} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}}.
\end{aligned} \tag{48}$$

From (22):

$$\begin{aligned}
& A\, e^{-(n-D)\mathcal{P}[\boldsymbol{\lambda}]} \sum_{\omega_0^{n-1}} m_l\left(\, \omega_0^{n-1} \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \\
& \leq \sum_{\omega_0^{n-1}} m_l\left(\, \omega_0^{n-1} \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \mu_{\boldsymbol{\lambda}}\left[\, \omega_0^{n-1} \,\right] \\
& \leq B\, e^{-(n-D)\mathcal{P}[\boldsymbol{\lambda}]} \sum_{\omega_0^{n-1}} m_l\left(\, \omega_0^{n-1} \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}
\end{aligned} \tag{49}$$

and:

$$\begin{aligned}
& A\, e^{-(n-D)\mathcal{P}[\boldsymbol{\lambda}]} \sum_{\omega_0^{n-1}} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \\
& \leq \sum_{\omega_0^{n-1}} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} \mu_{\boldsymbol{\lambda}}\left[\, \omega_0^{n-1} \,\right] \\
& \leq B\, e^{-(n-D)\mathcal{P}[\boldsymbol{\lambda}]} \sum_{\omega_0^{n-1}} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}.
\end{aligned}$$

Therefore:

$$\frac{A}{B} \frac{\sum_{\omega_0^{n-1}} m_l\left(\, \omega_0^{n-1} \,\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}}{\sum_{\omega_0^{n-1}} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})} e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}}$$

$$\leq \frac{\sum_{\omega_0^{n-1}} m_l\left(\omega_0^{n-1}\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}\mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right]}{\sum_{\omega_0^{n-1}} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}\mu_{\boldsymbol{\lambda}}\left[\omega_0^{n-1}\right]}$$

$$\leq \frac{B}{A}\frac{\sum_{\omega_0^{n-1}} m_l\left(\omega_0^{n-1}\right) e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}}{\sum_{\omega_0^{n-1}} e^{\Delta\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}e^{\mathcal{H}_{\boldsymbol{\lambda}}(\omega_0^{n-1})}}.$$

Now, from [29,31], (49) gives (46).

3.2.3. Taylor Expansion of the Pressure

The idea is here to use a Taylor expansion of the topological pressure. This approach is very much in the spirit of [38], but extended here to the spatio-temporal case. Since $\boldsymbol{\lambda}' = \boldsymbol{\lambda} + \boldsymbol{\delta}$, we have:

$$\begin{aligned}\mu_{\boldsymbol{\lambda}'}[m_l] &= \mu_{\boldsymbol{\lambda}}[m_l] + \sum_{j=1}^{L}\frac{\partial\mu_{\boldsymbol{\lambda}}[m_l]}{\partial\lambda_j}\delta_j + \frac{1}{2}\sum_{j,k=1}^{L}\frac{\partial^2\mu_{\boldsymbol{\lambda}}[m_l]}{\partial\lambda_j\partial\lambda_k}\delta_j\delta_k + \ldots \\ &= \mu_{\boldsymbol{\lambda}}[m_l] + \sum_{j=1}^{L}\frac{\partial^2\mathcal{P}[\boldsymbol{\lambda}]}{\partial\lambda_j\partial\lambda_l}\delta_j + \frac{1}{2}\sum_{j,k=1}^{L}\frac{\partial^3\mathcal{P}[\boldsymbol{\lambda}]}{\partial\lambda_j\partial\lambda_k\partial\lambda_l}\delta_j\delta_k + \ldots \end{aligned} \tag{50}$$

The second derivative of the pressure is given by [29,32–34]:

$$\frac{\partial^2\mathcal{P}[\boldsymbol{\lambda}]}{\partial\lambda_j\partial\lambda_l} = \sum_{n=-\infty}^{+\infty} C_{jl}(n) \equiv \chi_{jl}[\boldsymbol{\lambda}], \tag{51}$$

where:

$$C_{jl}(n) = \mu_{\boldsymbol{\lambda}}[m_j\, m_l \circ \sigma^n] - \mu_{\boldsymbol{\lambda}}[m_j]\,\mu_{\boldsymbol{\lambda}}[m_l], \tag{52}$$

is the correlation function between m_l, m_k at time n, computed with respect to $\mu_{\boldsymbol{\lambda}}$. (51) is a version of the fluctuation-dissipation theorem in the spatio-temporal case. σ^n is the time shift applied n times. The third derivatives can be computed, as well, by taking the derivative (51) and using (47). This generates terms with third order correlations, and so on [37]. Up to second order, we have:

$$\mu_{\boldsymbol{\lambda}'}[m_l] = \mu_{\boldsymbol{\lambda}}[m_l] + \sum_{j=1}^{L}\chi_{jl}[\boldsymbol{\lambda}]\,\delta_j + \ldots \tag{53}$$

Since the observable are monomials, they only take the values zero or one, and the computation of χ_{jl} is straightforward, reducing to counting the occurrence of time pairs, $t, t+n$, such that $m_j(t) = 1$ and $m_l(t+n) = 1$.

On practical grounds, we introduce a parameter $\Delta = \|\boldsymbol{\lambda}' - \boldsymbol{\lambda}\|$, which measures the variation in the parameters after updating. If Δ is small enough (smaller than some Δ_c), the terms of order three in the Taylor expansion are negligible; then, we can use (53). Otherwise, if Δ is big, we compute a new Monte Carlo estimation of $\mu'_{\boldsymbol{\lambda}}$ (as described in [19]). We explain in Section 4.2 how Δ_c was chosen in our data. Then, we use the following trick. If $\|\boldsymbol{\delta}\| > \Delta_c$, we compute the new value $\mu_{\boldsymbol{\lambda}'}[m_j]$. If $\Delta_c > \|\boldsymbol{\delta}\| > \frac{\Delta_c}{10}$, we use the linear response approximation (53) of $\mu_{\boldsymbol{\lambda}'}$. Finally, if $\|\boldsymbol{\delta}\| < \frac{\Delta_c}{10}$, we use $\mu_{\boldsymbol{\lambda}}[m_l]$ instead of $\mu_{\boldsymbol{\lambda}'}[m_l]$ in the next iteration of the method. Thus, in the case, $\|\boldsymbol{\delta}\| < \Delta_c$, we use the Gibbs distribution computed at some time step, say n, to infer the values at the next iteration. If we

do that several successive time steps, the distance to the original value, $\boldsymbol{\lambda}_n$, of the parameters increases. Therefore, we compute the norm $\|\boldsymbol{\lambda}_n - \boldsymbol{\lambda}_{n+k}\|$ at each time step, k, and we do not compute a new raster until this norm is larger than Δ_c.

3.3. The Algorithms

We have two algorithms, sequential and parallel, which are very similar to Dudik *et al.* Especially, the convergence of their algorithms, proven in their paper, extends to our case, since it only depends on the shape of the cost functions (36, 37). We describe here the algorithms coming out from the presented mathematical framework, in a sequential and parallel version. We iterate the algorithms until the distance $\eta = d\left(\mu_{\boldsymbol{\lambda}}, \pi_\omega^{(T)}\right)$ is smaller than some η_c. We use the Hellinger distance:

$$d\left(\mu_{\boldsymbol{\lambda}}, \pi_\omega^{(T)}\right) = \frac{1}{\sqrt{2}}\sqrt{\sum_{l=1}^{L}\left(\sqrt{\pi_\omega^{(T)}(m_l)} - \sqrt{\mu_{\boldsymbol{\lambda}}(m_l)}\right)^2} \tag{54}$$

3.3.1. Sequential Algorithm

Algorithm 1: The sequential algorithm.

Input: The features' empirical probabilities, $\pi_\omega^{(T)}[\ m_l\ ]$
Output: The vector of parameters, $\boldsymbol{\lambda}$,
initialization: $\lambda_l = 0$ for every l, $\Delta = 0$
while $\eta > \eta_c$ **do**
  $(\delta, l) = \arg\min_{l,\delta} F_l(\boldsymbol{\lambda}, \delta)$
  $\lambda_l \leftarrow \lambda_l + \delta$
  $\Delta \leftarrow \sqrt{\Delta^2 + \delta^2}$
  if $\Delta > \Delta_c$ **then**
    Compute a new Gibbs sample using the Monte Carlo method [19]
  else
    Compute the new features probabilities using the Taylor expansion (Equation 53)
  end
end

.

δ is the learning rate by which we change the value of a parameter, λ_l. η is the convergence criterion (54)). Δ is the parameter allowing us to decide whether we update the parameter change by computing a new Gibbs sample or by the Taylor expansion. F_l is given by Equation (36)

3.4. Parallel Algorithm

Algorithm 2: The parallel algorithm. G_l is given by (37).

Input: The features' empirical probabilities, $\pi_\omega^{(T)}\ [\ m_l\ ]$
Output: parameters λ_l
initialization: $\lambda_l = 0$ for every l, $\Delta = 0$
while $\eta > \eta_c$ **do**
  for $l \leftarrow 1$ **to** L **do**
    $\delta_l = \arg\min_\delta G_l(\boldsymbol{\lambda}, \boldsymbol{\delta})$
  end
  $\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \boldsymbol{\delta}$
  $\Delta \leftarrow \sqrt{\Delta^2 + \sum_{l=1}^{L} \delta_l^2}$
  if $\Delta > \Delta_c$ **then**
    Compute a new Gibbs sample using the Monte Carlo method [19]
  else
    Compute the new features probabilities using the Taylor expansion (Equation 53)
  end
end

The implementation of those algorithms consists of an important part in software developed at INRIA (Institut National de Recherche en Informatique et en Automatique) and called EnaS (Event Neural Assembly Simulation). The executable is freely available at [39].

4. Results

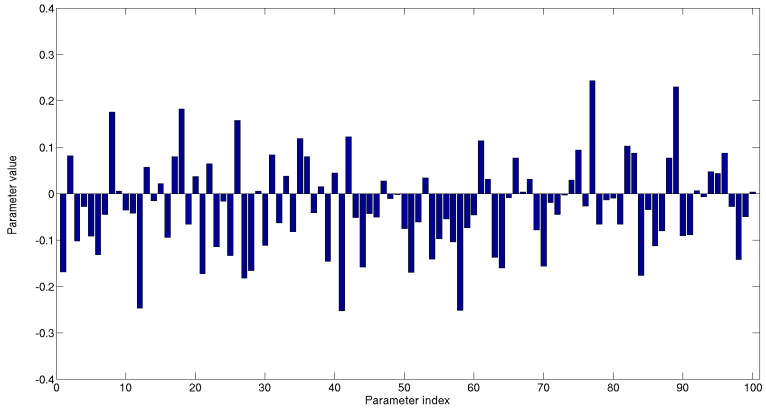
In this section, we perform several tests on our method. We first consider synthetic data generated with a known Gibbs potential and recover its parameters. This step also allows us to tune the parameter, Δ_c, in the algorithms. Then, we consider real data analysis, where the Gibbs potential form is unknown. This last step is not a systematic study that would be out of the scope of this paper, but simply provided as an illustration and comparison with the paper of Schneidman *et al.* 2006 [9].

4.1. Synthetic Data

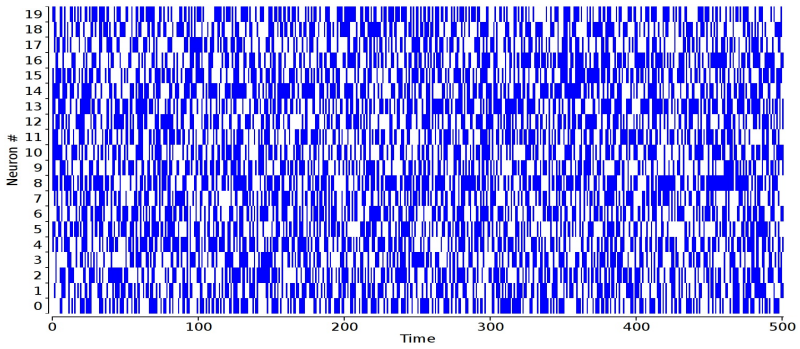
Synthetic data are obtained by generating a raster distributed according to a Gibbs distribution, whose potential (2) is known. We consider two families of Gibbs potentials. For each family, there are $L > N$ monomials, whose range belongs to $\{1, \ldots, R\}$. Among them, there are N "rate monomials" $\omega_i(D)$, $i = 1 \ldots N$, whose average gives the firing rate of neuron i, denoted r_i; the $L - N$ other monomials, with degree $k > 1$, are chosen at random with a probability law $\sim e^{-k}$, which favors, therefore, pairwise interactions. The difference between the two families comes from the distribution of coefficients, λ_l.

1. **"Dense" raster family**. The coefficients are drawn with a Gaussian distribution with mean zero and variance $\frac{1}{L}$ to ensure a correct scaling of the coefficients dispersion as L increases (Figure 1(a)). This produces typically a dense raster (Figure 1(b)) with strong multiple correlations.

Figure 1. The dense family.



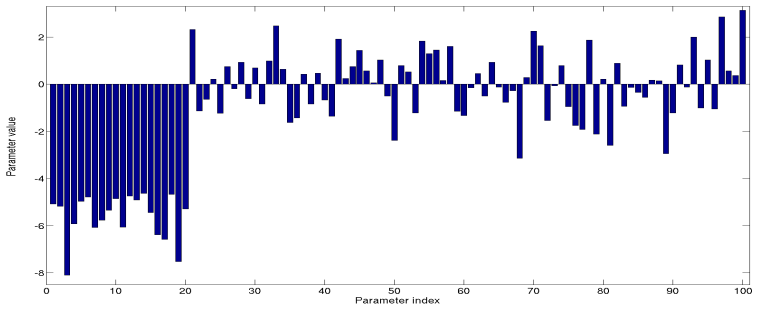
(a) Example of the coefficient distribution in the dense raster family.



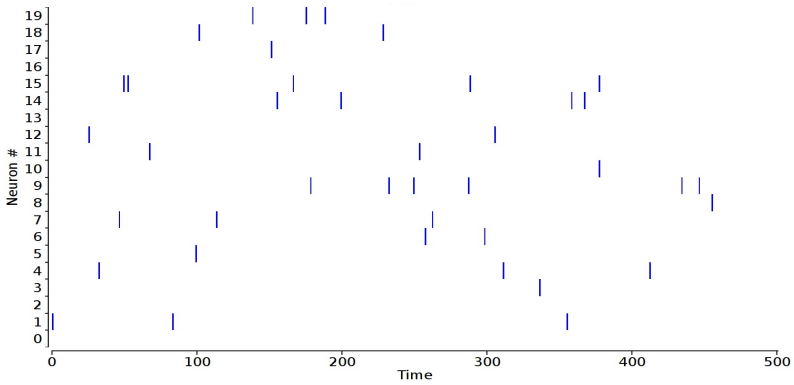
(b) Dense spike train.

2. **"Sparse" raster family**. The rate coefficients in the potential are very negative: the coefficient, h_i, of the rate monomial, $\omega_i(D)$, is $h_i = \log\left(\frac{r_i}{1-r_i}\right)$, where $r_i \in [0 : 0.01]$ with a uniform probability distribution. Other coefficients are drawn with a Gaussian distribution with mean 0.8 and variance one (Figure 2(a)). This produces a sparse raster (Figure 2(b)) with strong multiple correlations.

Figure 2. The sparse family.



(a) Example of the coefficient distribution in the sparse raster family.



(b) Sparse spike train.

4.2. Tuning Δ_c

For small N, R ($NR \leq 20$), it is possible to exactly compute the topological pressure using the transfer matrix technique [16]. We have therefore a way to compare the Taylor expansion (51) and the exact value.

If we perturb $\boldsymbol{\lambda}$ by an amount, δ, in the direction, l, this induces a variation on $\mu_{\boldsymbol{\lambda}}[m_l]$, $l = 1 \ldots L$, given by the Taylor expansion (53). To the lowest order $\mu_{\boldsymbol{\lambda}'}[m_l] = \mu_{\boldsymbol{\lambda}}[m_l] + O^{(1)}$, so that:

$$\epsilon^{(1)} = \frac{1}{L} \sum_{l=1}^{L} \frac{|\mu_{\boldsymbol{\lambda}'}[m_l] - \mu_{\boldsymbol{\lambda}}[m_l]|}{|\mu_{\boldsymbol{\lambda}'}[m_l]|}$$

is a measure of the relative error when considering the lowest order expansion.

In the same way, to the second order:

$$\mu_{\boldsymbol{\lambda}'}[m_l] = \mu_{\boldsymbol{\lambda}}[m_l] + \sum_{j=1}^{L} \chi_{jl}[\boldsymbol{\lambda}]\,\delta_j + O^{(2)},$$

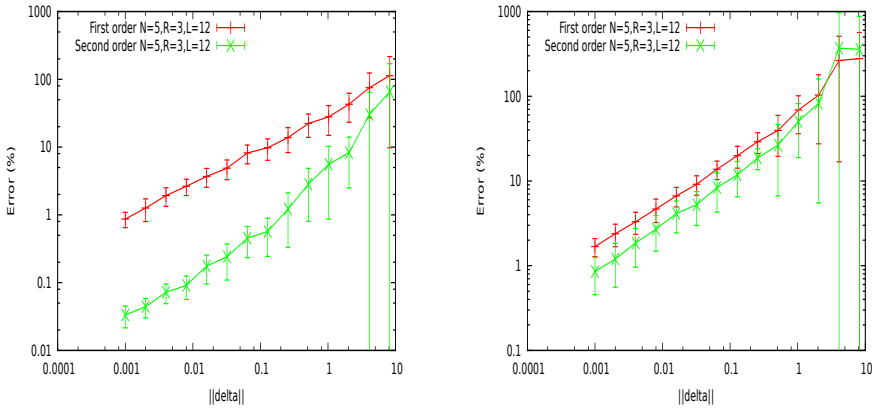
so that:

$$\epsilon^{(2)} = \frac{1}{L} \sum_{l=1}^{L} \frac{\left|\mu_{\boldsymbol{\lambda}'}[m_l] - \mu_{\boldsymbol{\lambda}}[m_l] - \sum_{j=1}^{L} \chi_{jl}[\boldsymbol{\lambda}]\,\delta_j\right|}{|\mu_{\boldsymbol{\lambda}'}[m_l]|},$$

is a measure of the relative error when considering the next order expansion.

In Figure 3, we show the relative errors, $\epsilon^{(1)}, \epsilon^{(2)}$ (in percent), as a function of δ. For each point, we generate 25 potentials, with $N = 5, R = 3, L = 12$. For each of these potentials, we randomly perturb the λ_js, with a random sign, so that the norm of the perturbation $\|\boldsymbol{\delta}\|$ is fixed. The linear response, χ, is computed from a raster of length $T = 100,000$.

Figure 3. Error on the average $\mu_{\boldsymbol{\lambda}'}[\, m_l \,]$ as a function of the perturbation amplitude, δ. First order corresponds to $\epsilon^{(1)}$ and second order to $\epsilon^{(2)}$ (see the text). The curves correspond to $N = 5, R = 3, L = 12$. (**Left**) The dense case; (**right**) the sparse case.



These curves show a big difference between the dense and sparse case. In the dense case, the second order error is about 5% for $\Delta_c = 1$, whereas we need a $\Delta_c \sim 0.03$ to get the same 5% in the sparse case. We choose to align on the sparse case, and in typical experiments, we take $\Delta_c = 0.1$, corresponding to about 10% of the error on the second order.

4.3. Computation of the Kullback-Leibler Divergence

To compute the Kullback-Leibler divergence between the empirical distribution, $\pi_\omega^{(T)}$, and the fitted predicted distribution, $\mu_{\boldsymbol{\lambda}}$, we need to know the value of the pressure, $\mathcal{P}[\,\boldsymbol{\lambda}\,]$, the empirical probability of the potential, $\pi_\omega^{(T)}[\,\mathcal{H}_{\boldsymbol{\lambda}}\,]$, and the entropy, $\mathcal{S}\left[\,\pi_\omega^{(T)}\,\right]$. For small networks, we can compute the pressure using the Perron–Frobenius theorem ([16]). However, for large scales, since we cannot compute the pressure, computing the Kullback-Leibler divergence is not direct and exact. We compute an approximation using the following technique. From Equation (18) and (24), we can write:

$$\begin{aligned} d_{kl}(\pi_\omega^{(T)}, \mu_{\boldsymbol{\lambda}}) &= \mu_{\boldsymbol{\lambda}}[\,\mathcal{H}_{\boldsymbol{\lambda}}\,] + \mathcal{S}[\,\mu_{\boldsymbol{\lambda}}\,] - \pi_\omega^{(T)}[\,\mathcal{H}_{\boldsymbol{\lambda}}\,] - \mathcal{S}\left[\,\pi_\omega^{(T)}\,\right] \\ &= \sum_l \lambda_l \big(\mu_{\boldsymbol{\lambda}}[m_l] - \pi_\omega^{(T)}[\,m_l\,]\big) + \mathcal{S}[\,\mu_{\boldsymbol{\lambda}}\,] - \mathcal{S}\left[\,\pi_\omega^{(T)}\,\right] \end{aligned} \tag{55}$$

From the parameters, $\boldsymbol{\lambda}$, we compute a spike train distributed as $\mu_{\boldsymbol{\lambda}}$ using the Monte Carlo method ([19]). From this spike train, we compute the monomials averages, $\mu_{\boldsymbol{\lambda}}[m_l]$, and the entropy, $\mathcal{S}[\,\mu_{\boldsymbol{\lambda}}\,]$, using the method of Strong *et al.* ([40]). $\pi_\omega^{(T)}[\,m_l\,]$ and $\mathcal{S}[\pi_\omega^{(T)}]$ are computed directly on the empirical data set.

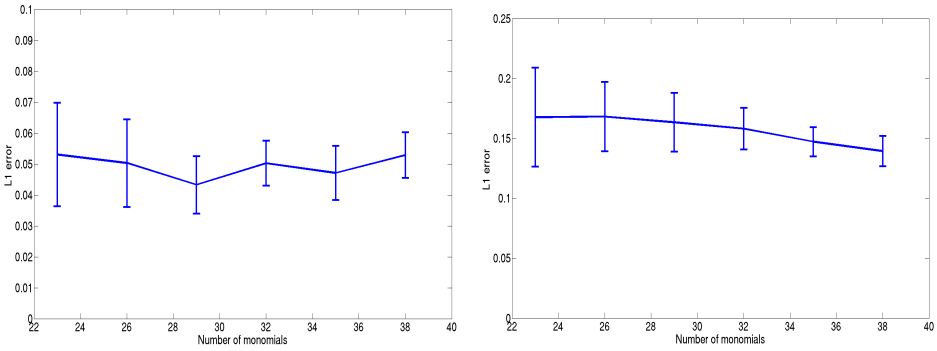
4.4. Performances on Synthetic Data

Here, we test the method on synthetic data, where the shape of the sought potential is known: only the λ_ls have to be estimated. Experiments were designed according to the following steps:

- We start from a potential $\mathcal{H}_{\boldsymbol{\lambda}^*} = \sum_{l\in\mathcal{L}} \lambda_l^* m_l$. The goal is to estimate the coefficient values, λ_l^*, knowing the set, $\mathcal{L}$, of monomials spanning the potential.
- We generate a synthetic spike train (ω_s) distributed according to the Gibbs distribution of $\mathcal{H}_{\boldsymbol{\lambda}^*}$.
- We take a potential $\mathcal{H}_{\boldsymbol{\lambda}} = \sum_{l\in\mathcal{L}} \lambda_l m_l$ with random initial coefficients λ_l. Then, we fit the parameters, λ_l, to the synthetic spike train, $\omega_s^{(T)}$.
- We evaluate the goodness of fit.

For the last step (goodness of fit), we have used three criteria. The first one simply consists of computing the L_1 error $d_1 = \frac{1}{L}\sum_{l=1}^{L} \left| \lambda_l^* - \lambda_l^{(est)} \right|$, where $\lambda_k^{(est)}$ is the final estimated value. d_1 is then averaged on 10 random potentials. Figure 4 shows the committed error in the case of sparse and dense potentials. The method showed a good performance, both in the dense and sparse case, for large $N \times R \sim 60$.

Figure 4. The distance between the exact value of coefficients and the estimated value, averaged on the set of 10 random potentials for $NR = 60$. (**a**) Dense spike trains; (**b**) sparse spike trains.



(a) (b)

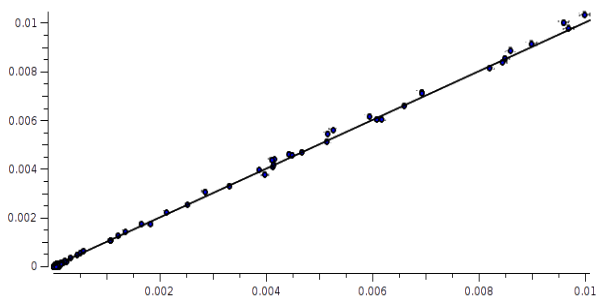
The main advantage of this criterion is that it provides an exact estimation of the error made on coefficient estimation. Its drawback is that we have to know the shape of the potential that generated the raster: this is not the case anymore for real neural network data. We therefore used a second criterion: confidence plots. For each spike block, ω_0^D, appearing in the raster, ω_s, we draw a point in a two-dimensional diagram with, on abscissa, the observed empirical probability, $\pi_{\omega_s}^{(T)}\left[\ \omega_0^D\ \right]$, and, on ordinate, the predicted probability, $\mu_{\boldsymbol{\lambda}}\left[\ \omega_0^D\ \right]$. Ideally, all points should align on the diagonal $y = x$ (equality line). However, since the raster is finite, there are finite-sized fluctuations ruled by the central limit theorem. For a block, ω_0^D, generated by a Gibbs distribution, $\mu_{\boldsymbol{\lambda}}$, and having an exact probability, $\mu_{\boldsymbol{\lambda}}\left[\ \omega_0^D\ \right]$, the empirical probability, $\pi_{\omega_s}^{(T)}\left[\ \omega_0^D\ \right]$, is a Gaussian random variable with mean $\mu_{\boldsymbol{\lambda}}\left[\ \omega_0^D\ \right]$ and mean-square deviation $\sigma = \frac{\sqrt{\mu_{\boldsymbol{\lambda}}\left[\ \omega_0^D\ \right]\left(1-\mu_{\boldsymbol{\lambda}}\left[\ \omega_0^D\ \right]\right)}}{\sqrt{T}}$. The probability that $\pi_{\omega_s}^{(T)}\left[\ \omega_0^D\ \right] \in$

$\left[\mu_{\boldsymbol{\lambda}}\left[\omega_0^D\right]-3\sigma, \mu_{\boldsymbol{\lambda}}\left[\omega_0^D\right]+3\sigma\right]$ is therefore of about $99,6\%$. This interval is represented by confidence lines spreading around the diagonal. As a third criterion, we have used the Kullback-Leibler divergence (55).

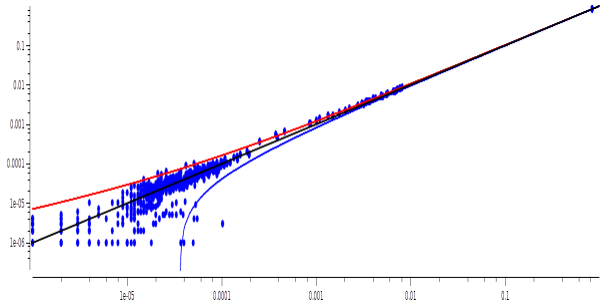
We have plotted two examples in Figures 5 and 6 for sparse data types:

1. Spatial case, 40 neurons, ($NR = 40$): Ising model (3). Figure 5.
2. Spatio-temporal, 40 neurons, $R = 2$ ($NR = 80$): Pairwise model with delays (5). Figure 6

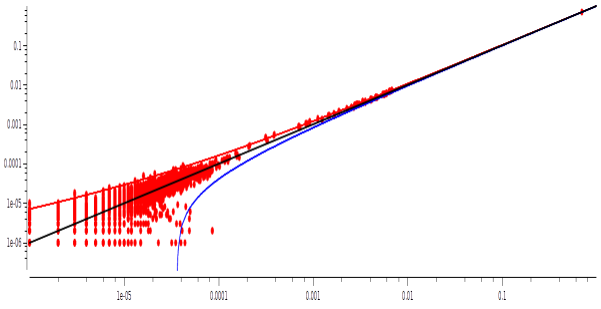
Figure 5. Data were generated with an Ising distribution. After fitting with an Ising model, we show the comparison between observed and predicted probabilities of monomials in (**a**). (**b**,**c**,**d**) The comparison of predicted and observed probabilities of patterns of Depths 1, 2 and 3, respectively. In the four plots, the x-axis represents the observed probabilities and the y-axis the predicted probabilities. The estimated Kullback-Leibler divergence is 0.0107.



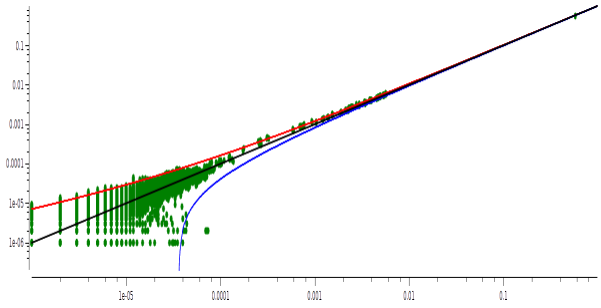
(a) Monomial averages.



(b) Patterns of Depth 1.



(c) Patterns of Depth 2.

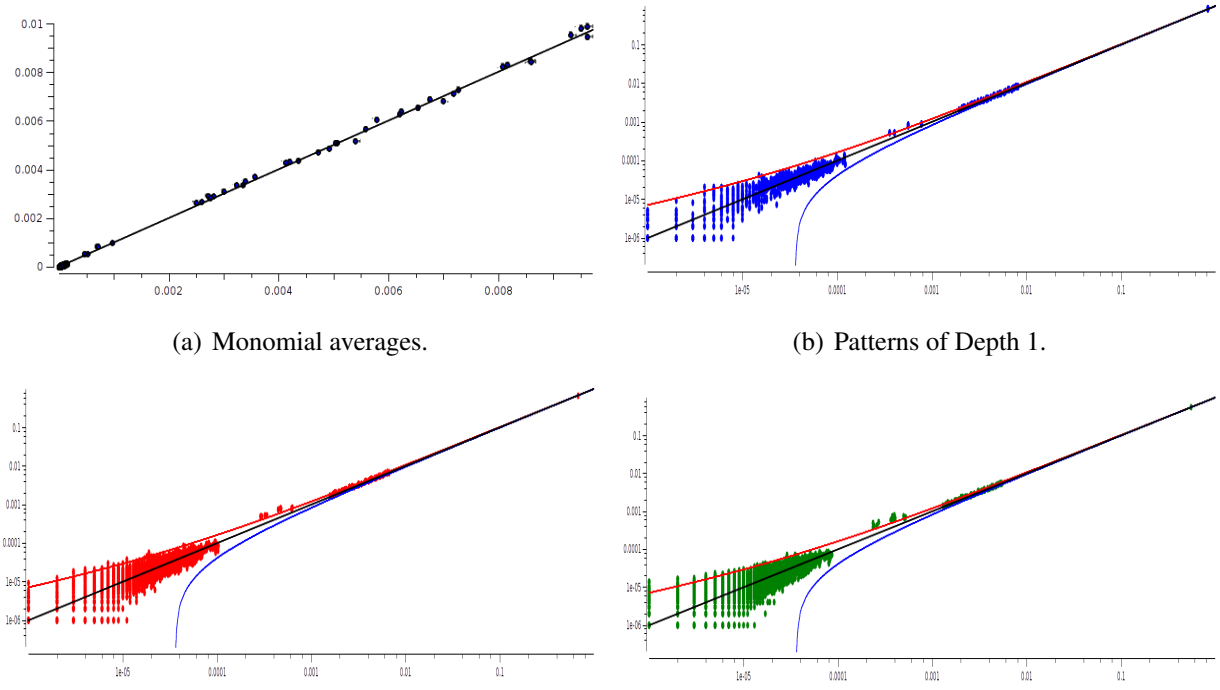


(d) Patterns of Depth 3.

4.5. The Performance on Real Data

Here, we show the inferring of the MaxEnt distribution on real spike trains. We analyzed a data set of 20 and 40 neurons with spatial and spatio-temporal constraints (data courtesy of M. J. Berry and O. Marre, 40 is the maximal number of neurons in this data set). Data are binned at 20 ms. We show the confidence plots and an example of convergence curves using the Hellinger distance. The goal here is to check the goodness of fit, not only for spatial patterns (as done in [9–12]), but also for spatio-temporal patterns.

Figure 6. Data were generated with a pairwise distribution of range $R = 2$. After fitting with a pairwise model of range $R = 2$, we show the comparison between observed and predicted probabilities of monomials in (**a**). (**b**,**c**,**d**) The comparison of predicted and observed probabilities of patterns of Depths 1, 2 and 3, respectively. In the four plots, the x-axis represents the observed probabilities and the y-axis the predicted probabilities. The estimated Kullback-Leibler divergence is 0.0174.



(a) Monomial averages.

(b) Patterns of Depth 1.

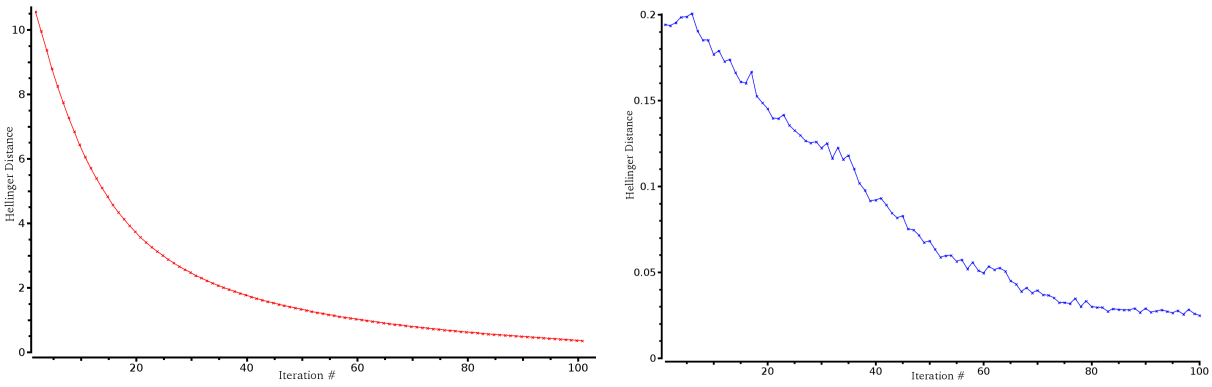
(c) Patterns of Depth 2.

(d) Patterns of Depth 3.

Figure 7 shows the evolution of the Hellinger distance during the parameter update both in the parallel and sequential update process.

After estimating the parameters of an Ising and pairwise model of range $R = 2$ on a set of 20 neurons, we evaluate the confidence plots. Figures 8 and 9 show, respectively, the confidence plots for patterns of Ranges 1, 2 and 3 after fitting with an Ising model and the pairwise model of range $R = 2$. Our results on 20 neurons confirm the observations made in [16] for $N = 5, R = 2$: a pairwise model with memory performs quite better than an Ising model to explain spatio-temporal patterns.

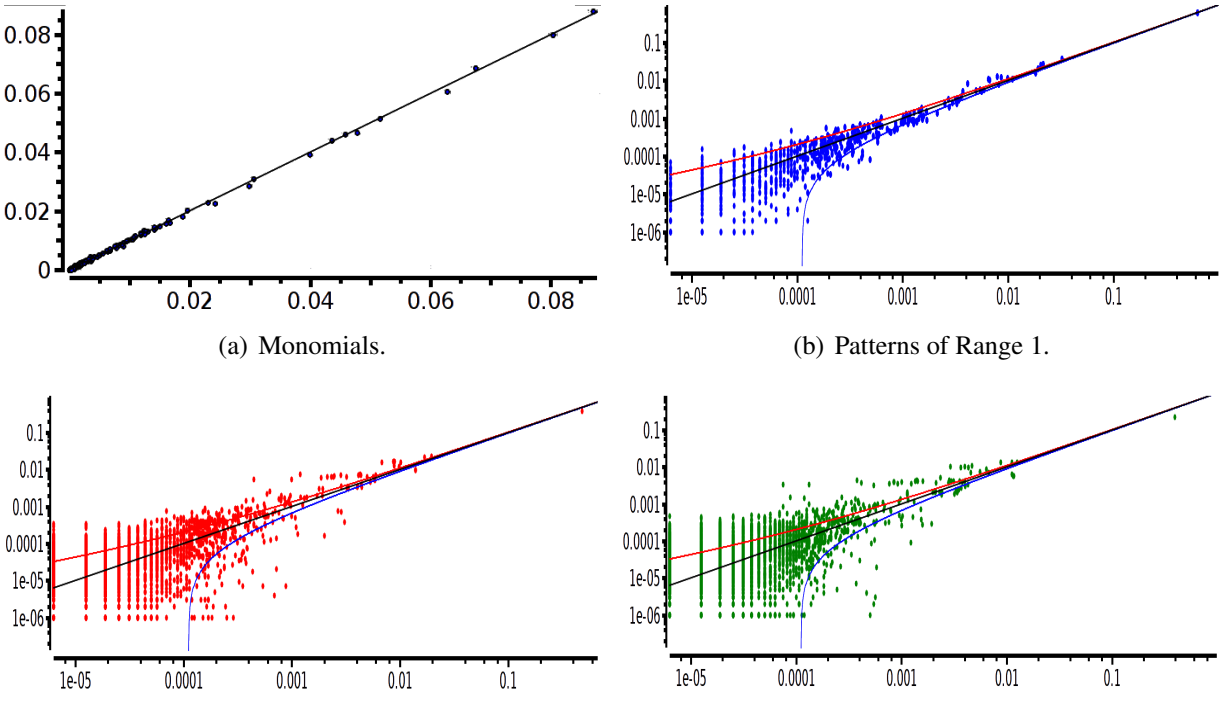
Figure 7. Evolution of the Hellinger distance during the parallel (**a**) and the sequential (**b**) update in the case of modeling a real data set with a pairwise model of range $R = 2$. The parallel update provides a fast convergence; however, it is steady after a hundred iterations. The,n we iterate the sequential algorithm.



(a) Convergence during the parallel update.

(b) Convergence during the parallel update.

Figure 8. A 20-neuron data set binned at 20 ms with an Ising model. After fitting, we show the comparison between observed (in the real spike train) and predicted average values of monomials in (**a**). (**b**,**c**,**d**) The comparison of predicted and observed probabilities for patterns of Ranges 1, 2 and 3, respectively. In (a), (b), (c) and (d), the x-axis represents the observed probabilities and the y-axis the predicted probabilities. The computation time is equal to 18 hours on a small cluster of 64 processors (around 5 min per iteration). The estimated Kullback-Leibler divergence is 0.307.



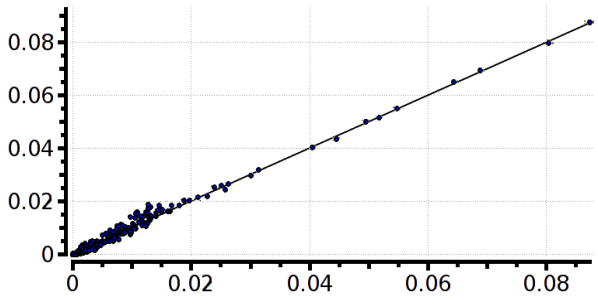
(a) Monomials.

(b) Patterns of Range 1.

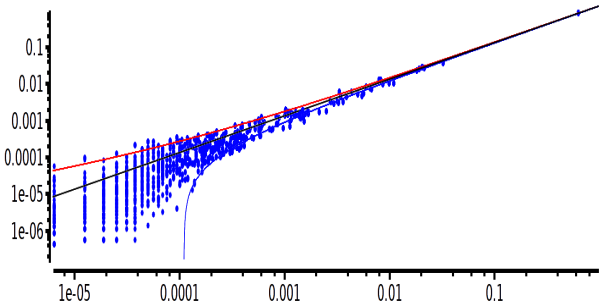
(c) Patterns of Range 2.

(d) Patterns of Range 3.

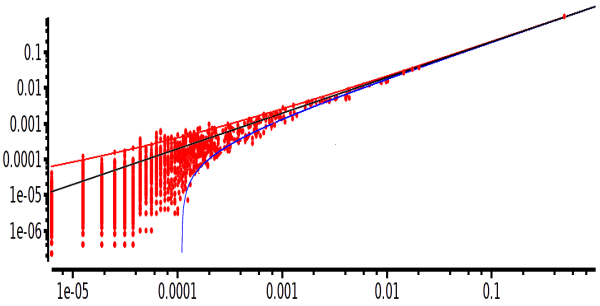
Figure 9. A 20-neuron data set binned at 20 ms with a pairwise model of Range 2. After fitting, we show the comparison between observed (in the real spike train) and predicted average values of monomials in (**a**). (**b**,**c**,**d**) The comparison of predicted and observed probabilities for patterns of Ranges 1, 2 and 3, respectively. In (a), (b), (c) and (d), the x-axis represents the observed probabilities and the y-axis the predicted probabilities. The computation time is equal to 40 hours on a small cluster of 64 processors (around 12 min per iteration). The estimated Kullback-Leibler divergence is 0.281.



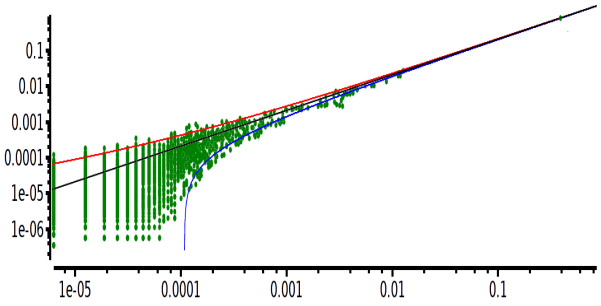
(a) Monomial averages.



(b) Patterns of Range 1.



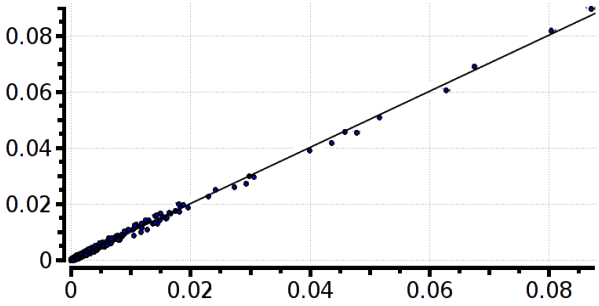
(c) Patterns of Range 2.



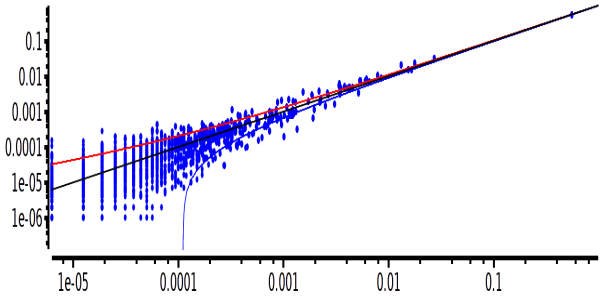
(d) Patterns of Range 3.

We then made the same analysis for 40 neuron. Figures 10 and 11 show, respectively, the confidence plots for patterns of Ranges 1, 2 and 3 after fitting with an Ising model and the pairwise model of range $R = 2$. In this case, we were not able to obtain a good convergence for $N = 40, R = 2$. This is presumably due to the insufficient length of the data set, which does not allow us to estimate accurately the probability of some monomials. This aspect is discussed in the next section.

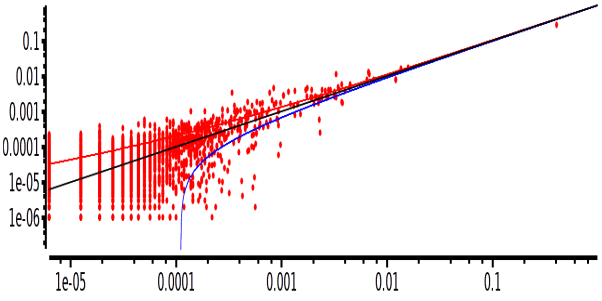
Figure 10. A 40-neuron data set binned at 20 ms with an Ising model. After fitting, we show the comparison between observed (in the real spike train) and predicted average values of monomials in (**a**). (**b**,**c**,**d**) The comparison of predicted and observed probabilities for patterns of Ranges 1, 2 and 3, respectively. In (a), (b), (c) and (d), the x-axis represents the observed probabilities and the y-axis the predicted probabilities. The computation time is equal to three days on a small cluster of 64 processors (around 21 min per iteration). The estimated Kullback-Leibler divergence is 0.930.



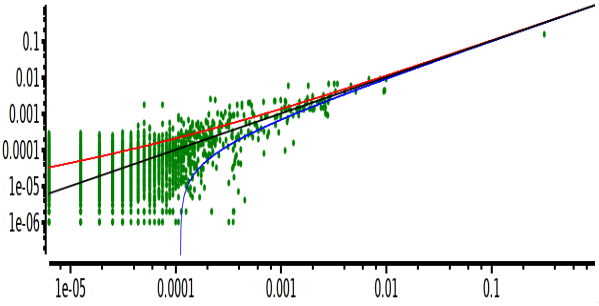
(a) Monomial averages.



(b) Patterns of Range 1.



(c) Patterns of Range 2.



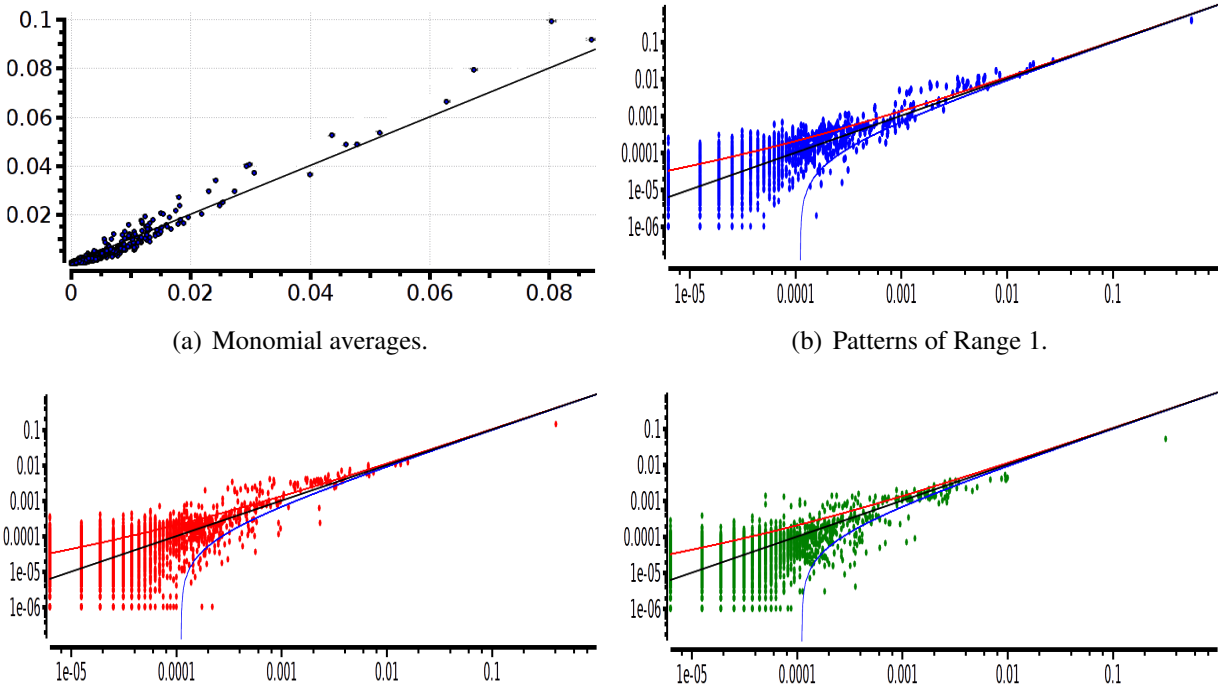
(d) Patterns of Range 3.

5. Discussion and Conclusion

The method shows better performances for synthetic data than for real data, although we did not make extensive studies of real data. The main reason, we believe, is that in the second case, we do not know the form of the potential. As a consequence, we stick to existing canonical forms of potentials, e.g., Ising and pairwise. The main problem with this approach is that the number of parameters to estimate dramatically grows with NR. The increase is moderate for the Ising model (N rates + $\frac{N(N-1)}{2}$ symmetric pairwise couplings), but it becomes prohibitively large even for pairwise range R models. On the opposite, our analysis of synthetic data used a relatively small number of parameters to fit.

The large number of parameters has two drawbacks: the increasing of the computation time and errors in the estimation. Let us comment on the second problem. It is not intrinsic to our method, nor is it intrinsic to MaxEnt; this is a well-known problem, which arises already when doing linear regression analysis. Increasing the number of parameters may eventually lead to catastrophic estimations, where the addition of a degree of freedom can seriously hinder the resolution.

Figure 11. A 40-neuron data set binned at 20 ms with a pairwise model of Range 2. After fitting, we show the comparison between observed (in the real spike train) and predicted average values of monomials in (**a**). (**b**,**c**,**d**) The comparison of predicted and observed probabilities for patterns of Ranges 1, 2 and 3, respectively. In (a), (b), (c) and (d), the x-axis represents the observed probabilities and the y-axis the predicted probabilities. The computation time is equal to seven days on a small cluster of 64 processors (around 47 min per iteration). The estimated Kullback-Leibler divergence is 0.983.



(a) Monomial averages.

(b) Patterns of Range 1.

(c) Patterns of Range 2.

(d) Patterns of Range 3.

In the case of MaxEnt, the situation can be described as follows. We generate a finite raster, ω_0^T, from a known distribution, $\mu_{\boldsymbol{\lambda}^*}$, with a potential of the form (2). Denote $\mu_{\boldsymbol{\lambda}^*}[\,\mathbf{m}\,]$ as the vector with entries $\mu_{\boldsymbol{\lambda}^*}[\,m_l\,]$ and $\pi_\omega^{(T)}[\,\mathbf{m}\,]$ as the vector with entries $\pi_\omega^{(T)}[\,m_l\,]$. From (19), we have $\mu_{\boldsymbol{\lambda}^*}[\,\mathbf{m}\,] = \nabla_{\boldsymbol{\lambda}^*}\mathcal{P}$. This exact solution is obtained when the Gibbs distribution, $\mu_{\boldsymbol{\lambda}^*}$, can be exactly sampled, namely, for an infinite raster. For a finite raster, if T is large enough to apply the central limit theorem, the empirical distribution, $\pi_\omega^{(T)}[\,\mathbf{m}\,]$, is Gaussian with mean $\mu_{\boldsymbol{\lambda}}[\,\mathbf{m}\,]$ and covariance $\frac{1}{T}\chi$ given by (51). We have, therefore, $\pi_\omega^{(T)}[\,\mathbf{m}\,] = \mu_{\boldsymbol{\lambda}^*}[\,\mathbf{m}\,] + \boldsymbol{\beta}$, where $\boldsymbol{\beta}$ is a centered Gaussian with covariance $\frac{1}{T}\chi$. Solving (19), where the exact probability, $\mu_{\boldsymbol{\lambda}^*}$, is replaced by the empirical one, $\pi_\omega^{(T)}$, one obtains an approximate solution of $\boldsymbol{\lambda}$, $\boldsymbol{\lambda}^*$ with: $\boldsymbol{\lambda} = \boldsymbol{\lambda}^* + \boldsymbol{\epsilon}$, where: $\nabla_{\boldsymbol{\lambda}}\mathcal{P} = \pi_\omega^{(T)}[\,\mathbf{m}\,]$. Therefore, $\nabla_{\boldsymbol{\lambda}}\mathcal{P} = \mu_{\boldsymbol{\lambda}^*}[\,\mathbf{m}\,] + \boldsymbol{\beta} = \nabla_{\boldsymbol{\lambda}^*+\boldsymbol{\epsilon}}\mathcal{P} = \nabla_{\boldsymbol{\lambda}^*}\mathcal{P} + \boldsymbol{\epsilon}\chi + O(\|\boldsymbol{\epsilon}\|^2)$. Hence, $\boldsymbol{\epsilon} = \chi^{-1}\boldsymbol{\beta}$. χ is invertible, since $\mathcal{P}$ is convex.

The fluctuations of the estimated solution, $\boldsymbol{\lambda}$, around the exact solution, $\boldsymbol{\lambda}^*$, are therefore Gaussian, centered, with covariance $\mathbb{E}[\,\epsilon.\tilde{\epsilon}\,] = \mathbb{E}\left[\,\chi^{-1}.\boldsymbol{\beta}.\tilde{\boldsymbol{\beta}}.\tilde{\chi}^{-1}\right]$. Since χ is symmetric, we have $\mathbb{E}[\,\epsilon.\tilde{\epsilon}\,] = \chi^{-1}.\mathbb{E}\left[\,\boldsymbol{\beta}.\tilde{\boldsymbol{\beta}}\right].\chi^{-1} = \frac{1}{T}\,\chi^{-1}$. We arrive, therefore, at the conclusion that the fluctuations on the estimated coefficients, $\boldsymbol{\lambda}$, are highly constrained by the convexity of the pressure, as expected. Mathematically, everything goes nicely, since $\mathcal{P}$ is convex. However, it may happen that $\mathcal{P}$ is quite flat in some directions/monomials. Then, small errors will be largely amplified. Therefore, when considering

potentials of the form (2), it is expected that some terms (monomials) not only are irrelevant, but also dramatically deteriorate the estimation problem, introducing almost zero eigenvalues in χ. This is presumably what happened in Figure 11, where we were not able to obtain a good convergence for monomial averages.

At this stage, the main question is therefore: can we have an idea of the potential shape from data before fitting the parameters? This question is not only related to the goodness of fit, but it is also a question of concept. Is is useful to represent a pairwise distribution for 40 neurons with nearly 2,000 parameters? The idea would then be to filter irrelevant monomials. For that, a feature selection method is useful and should complement this work. There are many directions we can take in favor of the feature selection; for instance, selecting the features on the threshold ([41,42]), using a χ^2 method ([43]), as well as the incremental feature selection algorithm ([44], [45]). Other methods based on periodic orbit sampling ([46]) and information geometry ([47,48]) are under current investigation.

We have presented a method to fit the parameters of the MaxEnt distribution with spatio-temporal constraints. In the process of exploring the dynamics of neural data, we hypothesize the model, fit it and, finally, judge the quality of the suggested model. Hence, this work is positioned as an important intermediate step in neural coding using the MaxEnt framework, opening the door for analyzing the dynamics of large networks, not being limited to spatial and/or traditional MaxEnt models.

Finally, we would like to highlight two points that should be investigated in further studies:

- The effect of binning. In many experimental studies, data is binned. Basically, binning was used in order to account for time spiking sensitivity, which is not the same for all the biological neural networks. For instance, [9] used 20 ms of binning for retinal spike trains. In the present paper, we have used the same as these authors, but we have not considered the effect of binning on our statistical estimations. This is certainly a matter of further investigations, especially because, to our best knowledge, no systematic study on the binning effects on statistics has been done. In particular, three distinct dimensions should be considered:
  - The statistical dimension: how does binning biases statistics? Could binning introduce spurious effects, such as, e.g., creating fallacious long-range correlations?
  - The computational dimension: how does the performance of the algorithm change with the bin size?
  - The biological dimension: cross-correlograms are not the same in all brain areas. Therefore, the optimal bin size is expected to depend on the investigated area.
- Maximum entropy: There are several methods now in use to model the spatio-temporal correlations in ensembles of neurons. The generalized linear model (GLM) approach uses the maximum likelihood and point-process to assess connectivity (e.g., [10]). Reverse correlation methods can also work well (e.g., [49]). Finally, there are causality metrics, like Granger causality or transfer entropy ([50]). Some of these methods have been compared in [51], but further investigations should be helpful, starting from synthetic data, where statistics is under good control. Especially, how does maximum entropy perform compared to these others methods?

Our method allows one to investigate these two questions on numerical grounds although such an investigation should be completed by mathematical insights, using the properties of spatio-temporal Gibbs distributions.

6. List of symbols

$\omega_i(n)$	Spike event
$\omega(n)$	Spike pattern
$\omega_{n_1}^{n_2}$	Spike block
ω	Spike train
T	Length (in time) of the spike train
N	Number of neurons
R	Model range
D	Model memory ($R = D - 1$)
$m_l(\omega)$	Monomial number l
$\mathbf{m}$	Vector of monomials
L	Total number of parameters (monomials) in the model
λ_l	Parameter number l
$\boldsymbol{\lambda}$	Parameters vector
$\mathcal{H}$	Gibbs potential
$Z_{\boldsymbol{\lambda}}$	Partition function
$\mathcal{S}$	Entropy
$\mathcal{P}$	Topological pressure
$\pi_\omega^{(T)}$	Empirical probability measured on the spike train, ω, of length T
$\mu_{\boldsymbol{\lambda}}$	Gibbs density with parameters $\boldsymbol{\lambda}$
$\mathcal{M}$	Set of invariant probabilities
$\delta_l = \lambda_l' - \lambda_l$	Learning rate or the value by which we update the parameters, λ_l
$\boldsymbol{\delta}$	Vector of learning rates
d_{KL}	Kullback-Leibler divergence
C_{jk}	Correlation between two monomials, j and k
χ	Hessian matrix (second derivative of the pressure)
Δ	Root sum square of the learning rates
$\boldsymbol{\beta}$	Fluctuations on the monomials averages
$\boldsymbol{\epsilon}$	Fluctuations on the parameters (relaxation)

Acknowledgments

We thank the reviewers for helpful remarks and constructive criticism. We also warmly acknowledge M.J. Berry and O. Marre for providing us MEA recordings from the retina and G. Tkacik, who provided us the references, [17,18], and helped us in the algorithm design. This work was partially supported by the ERC-NERVI number 227747, KEOPS ANR-CONICYT, and European FP7 projects RENVISION (FP7-600847), BRAINSCALES (FP7-269921).

Author's contribution

Real data was provided by M.J. Berry and O. Marre from Princeton university. The authors contributed equally to the presented mathematical and computational framework and the writing of the paper.

Conflicts of Interest

The authors declare no conflicts of interest.

References

1. Ferrea, E.; Maccione, A.; Medrihan, L.; Nieus, T.; Ghezzi, D.; Baldelli, P.; Benfenati, F.; Berdondini, L. Large-scale, high-resolution electrophysiological imaging of field potentials in brain slices with microelectronic multielectrode arrays. *Front. Neural. Circ.* **2012**, *6*.
2. Stevenson, I.H.; Kording, K.P. How advances in neural recording affect data analysis. *Nat. Neurosci.* **2011**, *14*, 139–142.
3. Marre, O.; Amodei, D.; Deshmukh, N.; Sadeghi, K.; Soo, F.; Holy, T.; Berry II, M. Mapping a Complete Neural Population in the Retina. *J. Neurosci.* **2012**, *43*, 14859–14873.
4. Hill, D.N.; Mehta, S.B.; Kleinfeld, D. Quality Metrics to Accompany Spike Sorting of Extracellular Signals. *J. Neurosci.* **2011**, *31*, 8699–8705.
5. Litke, A.M.; Bezayiff, N.; Chichilnisky, E.J.; Cunningham, W.; Dabrowski, W.; Grillo, A.A.; Grivich, M.; Grybos, P.; Hottowy, P.; Kachiguine, S.; Kalmar, R.S.; Mathieson, K.; Petrusca, D.; Rahman, M.; Sher, A. What does the eye tell the brain?: Development of a system for the large scale recording of retinal output activity. *IEEE Trans. Nucl. Sci.* **2004**, *51*, 1434–1440.
6. Quiroga, R.Q.; Nadasdy, Z.; Ben-Shaul, Y. Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. *Neural Comput.* **2004**, *16*, 1661–1687.
7. Csiszár, I. On the computation of rate-distortion functions (Corresp.). *Inform. Theory, IEEE T on* **1974**, *20*, 122–124.
8. Jaynes, E. Information theory and statistical mechanics. *Phys. Rev.* **1957**, *106*, 620.
9. Schneidman, E.; Berry, M.; Segev, R.; Bialek, W. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature* **2006**, *440*, 1007–1012.
10. Pillow, J.W.; Shlens, J.; Paninski, L.; Sher, A.; Litke, A.M.; Chichilnisky, E.J.; Simoncelli, E.P. Spatio-temporal correlations and visual signaling in a complete neuronal population. *Nature* **2008**, *454*, 995–999.
11. Ganmor, E.; Segev, R.; Schneidman, E. Sparse low-order interaction network underlies a highly correlated and learnable neural population code. *Proc. Natl. Acad. Sci. USA* **2011**, *108*, 9679–9684.
12. Ganmor, E.; Segev, R.; Schneidman, E. The architecture of functional interaction networks in the retina. *J. Neurosci.* **2011**, *31*, 3044–3054.
13. Tkačik, G.; Schneidman, E.; Berry II, M.J.; Bialek, W. Spin glass models for a network of real neurons. *arXiv preprint arXiv:0912.5409* **2009**.

14. Tang, A.; Jackson, D.; Hobbs, J.; Chen, W.; Smith, J.L.; Patel, H.; Prieto, A.; Petrusca, D.; Grivich, M.I.; Sher, A.; Hottowy, P.; Dabrowski, W.; Litke, A.M.; Beggs, J.M. A Maximum Entropy Model Applied to Spatial and Temporal Correlations from Cortical Networks *In Vitro*. *J. Neurosci.* **2008**, *28*, 505–518.
15. Marre, O.; El Boustani, S.; Frégnac, Y.; Destexhe, A. Prediction of spatiotemporal patterns of neural activity from pairwise correlations. *Phys. Rev. Lett.* **2009**, *102*.
16. Vasquez, J.C.; Marre, O.; Palacios, A.G.; Berry, M.J.; Cessac, B. Gibbs distribution analysis of temporal correlation structure on multicell spike trains from retina ganglion cells. *J. Physiol. Paris* **2012**, *106*, 120–127.
17. Dudík, M.; Phillips, S.; Schapire, R. Performance Guarantees for Regularized Maximum Entropy Density Estimation. Proceedings of the 17th Annual Conf. on Comp. Learn. Theory, 2004.
18. Broderick, T.; Dudik, M.; Tkacik, G.; Schapire, R.E.; Bialek, W. Faster solutions of the inverse pairwise Ising problem. *arXiv:0712.2437* **2007**.
19. Nasser, H.; Marre, O.; Cessac, B. Spatio-temporal spike train analysis for large scale networks using the maximum entropy principle and Montecarlo method. *J. Stat. Mech.* **2013**, *2013*, P03006.
20. Schaub, M.T.; Schultz, S.R. The Ising decoder: reading out the activity of large neural ensembles. *arXiv:1009.1828* **2010**.
21. Garibaldi, U.; Penco, M.A. Probability Theory and Physics Between Bernoulli and Laplace: The Contribution of J. H. Lambert (1728-1777). In Proceeding of the Fifth National Congress on the History of Physics, Rome, 1985; Volume 9, pp. 341–346.
22. Tkacik, G.; Marre, O.; Mora, T.; Amodei, D.; 2nd, M.B.; Bialek, W. The simplest maximum entropy model for collective behavior in a neural network. *J. Stat. Mech.* **2013**, P03011.
23. Jaynes, E.T. Where do we stand on maximum entropy. In *The Maximum Entropy Formalism*; Levine, D.; Tribus, M., Eds.; MIT Press: Cambridge, MA, USA, 1978; pp. 15–118.
24. Jaynes, E.T. The minimum entropy production principle. *Ann. Rev. Phys. Chem.* **1980**, *31*, 579–601.
25. Jaynes, E. Macroscopic prediction. In *Complex Systems - Operational Approaches in Neurobiology, Physics, and Computers*; Springer: Berlin, Germany, 1985; pp. 254–269.
26. Otten, M.; Stock, G. Maximum caliber inference of nonequilibrium processes. *J. Chem. Phys.* **2010**, *133*, 034119.
27. Fernandez, R.; Maillard, G. Chains with complete connections : General theory, uniqueness, loss of memory and mixing properties. *J. Stat. Phys.* **2005**, *118*, 555–588.
28. Gikhman, I.; Skorokhod, A. *The Theory of Stochastic Processes*; Springer: Berlin, Germany, 1979.
29. Chazottes, J.; Keller, G. Pressure and Equilibrium States in Ergodic Theory. *Isr. J. Math.* **2008**, *131*.
30. Ruelle, D. *Statistical Mechanics: Rigorous Results*; Benjamin: New York, NY, USA, 1969.
31. Keller, G. *Equilibrium States in Ergodic Theory*; Cambridge University Press: Cambridge, UK, 1998.
32. Ruelle, D. *Thermodynamic Formalism*; Addison-Wesley: Reading, MA, USA, 1978.

33. Bowen, R. Equilibrium States and the Ergodic Theory of Anosov Diffeomorphisms. In *Lecture Notes in Mathatics*; Springer-Verlag: New York, NY, USA, 1975. Volume 470.
34. Georgii, H.O. *Gibbs Measures and Phase Transitions (De Gruyter Studies in Mathematics)*; Springer: Berlin, Germany, 1988.
35. Vasquez, J.C.; Palacios, A.; Marre, O.; II, M.J.B.; Cessac, B. Gibbs distribution analysis of temporal correlation structure on multicell spike trains from retina ganglion cells. *J. Physiol. Paris* **2012**, *106*, 120–127.
36. Collins, M.; Schapire, R.E.; Singer, Y. Logistic Regression, AdaBoost and Bregman Distances. *Mach. Lear.* **2002**, *48*, 253–285.
37. Mayer, V.; Urbański, M. Thermodynamical formalism and multifractal analysis for meromorphic functions of finite order. *Memoir. Am. Math. Soc.* **2010**, *203*.
38. Kappen, H.; Rodriguez, F. Boltzmann Machine learning using mean field theory and linear response correction. In *NIPS*; Kearns, M, Ed.; MIT Press: Cambridge, MA, USA, 1998; Volume 12, pp. 280–286.
39. Event neural assembly Simulation: v3 version. Available online: http://enas.gforge.inria.fr/v3/download.html, accessed on 21 April 2014.
40. Strong, S.; Koberle, R.; de Ruyter van Steveninck, R.; Bialek, W. Entropy and information in neural spike trains. *Phys. Rev. Let* **1998**, *80*, 197–200.
41. Rosenfeld, R.; Carbonell, J.; Rudnicky, A. Adaptive Statistical Language Modeling: A Maximum Entropy Approach. Technical report, School of Computer Science, Carnegie Mellon University, 1994.
42. Koeling, R. Chunking with maximum entropy models. In *Proceeding ConLL '00 Proceedings of the 2nd Workshop on Learning Language in Logic and the 4th conference on Computational Natural Language Learning*; Volume 7, pp. 139-141; Association for Computational Linguistics: Stroudsburg, PA, USA, 2000.
43. Chen, S.F.; Rosenfeld, R. Efficient Sampling and Feature Selection in Whole Sentence Maximum *Ent. Lang. Mod.* 1999.
44. Berger, A.L.; Pietra, S.A.D.; Pietra, V.J.D. A Maximum Entropy approach to Natural Language Processing. *Comp. Lang.* **1996**, *22*, 39–71.
45. Zhou, Y.; Wu, L. A fast algorithm for feature selection in conditional maximum entropy modeling. in Proceedings of the Empirical Methods in Natural Language Processing (EMNLP 2003), Sapporo, Japan, July 2003; pp. 153–159.
46. Cessac, B.; Cofre, R. Estimating maximum entropy distributions from periodic orbits in spike trains. research report RR-8329, INRIA, 2013.
47. Nakahara, H.; Amari, S. Information-Geometric Decomposition in Spike Analysis. *Adv. Neural Inform. Process. Syst.* 2001, pp. 253–260.
48. Amari, S. Information geometry on hierarchy of probability distributions. *IEEE T. Inf. Theory* **2001**, *47*, 1701–1711.
49. Chichilnisky, E.J. A simple white noise analysis of neuronal light responses. *Network Comput. Neural Syst.* **2001**, *12*, 199–213.

50. Li, Z.; Li, X. Estimating Temporal Causal Interaction between Spike Trains with Permutation and Transfer Entropy. *PloS One* **2013**, *8*, e70894.
51. Truccolo, W.; Hochberg, L.R.; Donoghue, J.P. Collective dynamics in human and monkey sensorimotor cortex: predicting single neuron spikes. *Nature Neurosci.* **2009**, *13*, 105–111.