Mesoscopic modeling of hidden spiking neurons

Anonymous Author(s)
Affiliation
Address
email

Abstract

Can we use spiking neural networks (SNN) as generative models of multi-neuronal recordings, while taking into account that most neurons are unobserved? Modeling the unobserved neurons with large pools of hidden spiking neurons leads to severely underconstrained problems that are hard to tackle with maximum likelihood estimation. In this work, we use coarse-graining and mean-field approximations to derive a bottom-up, neuronally-grounded latent variable model (neuLVM), where the activity of the unobserved neurons is reduced to a low-dimensional mesoscopic description. In contrast to previous latent variable models, neuLVM can be explicitly mapped to a recurrent, multi-population SNN, giving it a transparent biological interpretation. We show, on synthetic spike trains, that a few observed neurons are sufficient for neuLVM to perform efficient model inversion of large SNNs, in the sense that it can recover connectivity parameters, infer single-trial latent population activity, reproduce ongoing metastable dynamics, and generalize when subjected to perturbations mimicking photo-stimulation.

1 Introduction

The progress of large-scale electrophysiological recording techniques [1] begs the following question: can we reverse engineer the probed neural microcircuit from the recorded data? If so, should we try to design large spiking neural networks (SNN), representing the whole microcircuit, capable of generating the recorded spike trains? Such networks would constitute fine-grained mechanistic models and would make *in silico* experiments possible. However appealing this endeavor may appear, it faces a major obstacle – that of unobserved neurons. Indeed, despite the large number of neurons that can be simultaneously recorded, they add up to a tiny fraction of the total number of neurons involved in any given task [2], making the problem largely underdetermined. Training SNNs with large numbers of hidden neurons is challenging because a huge number of possible latent spike patterns result in the same recurrent input to the recorded neurons, making training algorithms nontrivial [3–6].

From the perspective of a single recorded neuron, the spike activity of all the other neurons can be reduced to a single causal variable – the total recurrent input (Figure 1 A). Hence, we argue that fine-grained SNNs are not necessary to model the inputs from hidden neurons, but can be replaced by a coarse-grained model of the sea of unobserved neurons. One possible coarse-graining scheme consists in clustering neurons into homogeneous populations with uniform intra- and inter-population connectivity. With the help of mean-field neuronal population equations [7–10], this approach enables the reduction of large SNNs to low-dimensional mesoscopic models composed of neuronal populations interacting with each other [11–13]. Clusters can reflect the presence of different cell-types [11, 14, 15] or groups of highly interconnected excitatory neurons [16–21]. From a computational point of view, coarse-grained SNNs offer biologically plausible implementations of rate coding by ensembles of neurons [22, 23] and 'computation through neural population dynamics' [24].

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

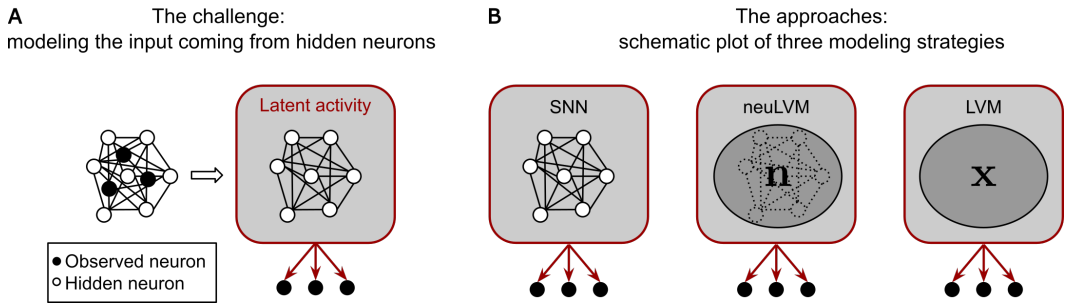

Figure 1: **Training SNNs with large number of hidden neurons: challenge and approaches.** (A) The challenge of modeling the input to the observed neurons (black) coming from the hidden neurons (white) while only a small fraction of neurons is observed. (B) Modeling strategies for the input coming from the hidden neurons: SNNs (left) model the fine-grained spike trains of all hidden neurons; neuLVM (middle) uses a mesoscopic description of the population activity, clustering neurons into homogeneous populations; classic LVMs (right) model the latent activity with low-dimensional phenomenological variables (the link to SNNs is lost).

In this paper, we show that, after clustering the unobserved neurons into several homogeneous populations, the finite-size neuronal population equation of Schwalger et al. [11] can be used to derive a neuronally-grounded latent variable model (neuLVM) where the activity of the unobserved neurons is summarized in a coarse-grained mesoscopic description. The hallmark of neuLVM is its direct correspondence to a multi-population SNN. As a result, both model parameters and latent variables have a transparent biological interpretation: the model is parametrized by single-neuron properties and synaptic connectivity; the latent variables are the summed spiking activities of the neuronal populations. Coarse-graining by clustering therefore turns an underdetermined problem – fitting a SNN with a large number of hidden neurons – into a tractable problem with interpretable solutions.

Switching metastable activity patterns that are not stimulus-locked have attracted a large amount of attention in systems neuroscience [25–27] for their putative role in decision-making [28], attention [29], and sensory processing [30]. Since generative SNN models of these metastable dynamics are available [11, 21, 31, 32], metastable networks constitute ready-to-use testbeds for bottom-up mechanistic latent variable models. Therefore, we propose metastable networks as a new benchmark and compare our neuLVM approach with competing approaches.

2 Relation to prior work

While many latent variable models (LVM), including Poisson Linear Dynamical Systems (PLDS) [33] and Switching Linear Dynamical Systems (SLDS) [34–37], have been designed for inferring low-dimensional population dynamics [38–48], their account for the population activity is a phenomenological one. In contrast, the LVM derived in this work is a true multiscale model, as latent population dynamics directly stem from neuronal dynamics.

Our method builds on René et al. [49], who showed that the mesoscopic model of Schwalger et al. [11] enables the inference of neuronal and connectivity parameters of multi-population SNNs via likelihood-based methods in the case where the mesoscopic population activity is *fully observable*. Here, for the first time, we show that mesoscopic modeling can also be applied to unobserved neurons, relating LVMs to mean-field theories for populations of spiking neurons [7–12]. Our neuLVM approach towards unobserved neurons differs from the Generalized Linear Model (GLM) approach [50–52] (and recent extensions [53–55]), which either neglects unobserved neurons or replaces unobserved neurons by stimulus-locked inputs. Our approach also avoids microscopic simulations of hidden spiking neurons [3, 4, 6], which scale poorly with the number of hidden neurons.

3 Background: mesoscopic modeling of the population activity

Detailed biophysical neuron models can be accurately approximated by simple spiking neuron models [56–58], which can be efficiently fitted to neural data [59–62]. Stochastic spiking neuron models where the neuron's memory of its own spike history is restricted to the time elapsed since its last spike (its *age*) are said to be of renewal-type.[1] Examples of renewal-type neurons include noisy leaky integrate-and-fire and 'Spike-Response Model 0' neurons [9, 10]. The dynamics of a homogeneous population of interacting renewal-type neurons can be described, in the mean-field limit, by an exact integral equation [7, 9, 10, 12, 64] (see [65–67] for rigorous proofs). In the case of homogeneous but finite populations, Schwalger et al. [11] derived a stochastic integral equation which provides a mesoscopic description (i.e. a description including finite-size fluctuations) of the population dynamics.

For clarity of exposition, in this section and the next, we focus on the case of a single homogeneous population with no external input. All the arguments presented below can be readily generalized to the case of multiple interacting populations with external input (Appendices B and C).

Let us consider a homogeneous SNN of N recurrently connected renewal-type spiking neurons. For T discrete time steps of length Δt, let $\mathbf{y} \in \{0,1\}^{N\times T}$ (a $N \times T$ binary matrix) denote the spike trains generated by the N neurons. The fact that the neurons are of renewal-type implies, by definition, that the probability for neuron i to emit a spike at time t can be written $p(y_t^i = 1|\mathbf{y}_{1:t-1}, \Theta) = \rho_{\theta_i}^{\Delta t}(a_i, \sum_j J^{ij}\mathbf{y}_{1:t-1}^j)$ where a_i is the age of neuron i (i.e. the number of time steps elapsed since the last spike of neuron i), the J^{ij} are the recurrent synaptic weights of the network, and θ_i are the parameters of neuron i. The sum $\sum_j J^{ij}\mathbf{y}_{1:t-1}^j$ represents the past input received by neuron i in all time steps up to $t-1$. The superscript Δt of the function $\rho_{\theta_i}^{\Delta t}$ indicates that we consider here the discrete-time 'escape rate' of the neuron but the transition to continuous time is possible [10]. The explicit expression for $\rho_{\theta_i}^{\Delta t}$ in the case of leaky integrate-and-fire neurons with 'escape noise' (LIF) is given in Appendix A.

A crucial notion in this work is that of 'homogeneous population'. The SNN described above forms a homogeneous population if all the recurrent synaptic weights are identical, that is, $J^{ij} = J$ (mean-field approximation) and if all the neurons share the same parameters, that is, $\theta_i = \theta$. In a homogeneous population, all the neurons share the same past input $J\mathbf{n}_{1:t-1}$, where $\mathbf{n}_{1:t-1} = (n_1, n_2, \ldots, n_{t-1})$ denotes the total number of spikes in the population in time steps $1, 2, \ldots, t-1$ with $n_{t'} = \sum_{i=1}^N y_{t'}^i$ being the total number of spikes in the population at time t'. Then, for *any* neuron in the population, the probability to emit a spike at time t, given its age a, is

$$p_{t,a}^{\text{fire}} = \rho_\theta^{\Delta t}(a, J\mathbf{n}_{1:t-1}). \tag{1}$$

Importantly, Eq. (1) is independent of the identity of the neuron.

In a microscopic description of the spiking activity, the vector $\mathbf{y}_t$ depends nonlinearly on the past $\mathbf{y}_{1:t-1}$. A mesoscopic description aims to reduce the high-dimensional microscopic dynamics to a lower-dimensional dynamical system involving the population activity n_t only (in the case of multiple interacting populations, $\mathbf{n}_t$ is vector of dimension k equal to the number of populations, Appendix A). While an exact reduction is not possible in general (neuron models being nonlinear), an excellent approximation, in the form of a stochastic integral equation, was proposed by Schwalger et al. [11]. In discrete time, the stochastic integral equation reads

$$n_t \sim \text{Binomial}\left(N,\ \bar{n}_t/N\right), \tag{2a}$$

$$\bar{n}_t = \left[\sum_{a\geq 1} p_{t,a}^{\text{fire}} S_{t,a}\, n_{t-a} + \underbrace{\Lambda_t\left(N - \sum_{a\geq 1} S_{t,a}\, n_{t-a}\right)}_{\text{'finite-size correction'}}\right]_+ . \tag{2b}$$

The variable $\bar{n}_t$ can be interpreted as the expected number of neurons firing at time t. The survival $S_{t,a} = \prod_{s=0}^{a-1}(1 - p_{t-a+s,s}^{\text{fire}})$ is the probability for a neuron to stay silent between time $t-a$ and

[1] Traditional renewal theory in the mathematical literature [63] is restricted to stationary input whereas we use 'renewal-type' in the broader sense that also includes time-dependent input.

$t-1$. The finite-size correction term stabilizes the model by enforcing the approximate neuronal mass conservation $\sum_{a\geq 1} S_{t,a}\, n_{t-a} \approx N$ (see [13] for an in-depth mathematical discussion). The 'modulating factor' Λ_t has an explicit expression [11, 13] in terms of p^{fire}, S and $\mathbf{n}$ (indices are dropped here for simplicity, complete formulas are presented in Appendix A, as well as explanations on how to initialize Eq. (2)). Importantly, for a populations of *interacting* neurons, p^{fire}, S and Λ depend on $\mathbf{n}$, which makes the stochastic equation (2) highly nonlinear. While the mesoscopic model (2) is not mathematically exact, it provides an excellent approximation of the first and second order statistics of the population activity [11], and is much more tractable than the exact 'field' equation [68, 69]. Also, the mesoscopic model (2) can be generalized to the case of non-renewal neurons with spike-frequency adaptation [11] and short-term synaptic plasticity [70].

Formally, Eq. (2) is reminiscent of the Point Process Generalized Linear Model (GLM) [50-52] for single neurons, with the notable difference that Eq. (2) contains additional nonlinearities beyond those of the GLM because p^{fire}, S and Λ all depend on $\mathbf{n}$ (Appendix A). Importantly, Equation (2) readily defines an expression for the probability $p(\mathbf{n}|\Theta)$ [49], where $\Theta = \{J, \theta\}$ denotes the model parameters. Thus, the mesoscopic model (2) allows us to avoid the intractable sum encountered if we naively try to derive $p(\mathbf{n}|\Theta)$ directly from the microscopic description (the intractable sum stems from the fact that the identity of neurons is lost in the observation $n_{t'}$ at each time step t', Figure 1 B).

4 Theoretical result: Neuronally-grounded latent variable model

In this section, we first recall why training SNN with large numbers of hidden neurons via the maximum likelihood estimator is computationally expensive. Then, we show that the mesoscopic description, Eq. (2), allows us to derive a tractable, neuronally-grounded latent variable model, which can be mapped to a multi-population SNNs.

For the sake of simplicity, all the arguments are presented for a single homogeneous population, but the generalization to multiple interacting populations is straightforward (Appendix C). Let us assume that we observe during T time steps the spike trains of q simultaneously recorded neurons that are part of a homogeneous population of N neurons, with $N > q$. We split the spike trains of the entire population $\mathbf{y} \in \{0,1\}^{N\times T}$ into the observed spike trains $\mathbf{y^o}$ (q neurons) and hidden spike trains $\mathbf{y^h}$ ($N-q$ neurons). Even for a single population, it is difficult to infer the parameters $\Theta = \{J, \theta\}$ of the SNN from observation $\mathbf{y^o}$ using the maximum likelihood estimator because, in the presence of hidden neurons, the likelihood $\mathcal{L}$ involves a marginalization over the latent spike trains $\mathbf{y^h}$:

$$\mathcal{L} = p(\mathbf{y^o}|\Theta) = \sum_{\mathbf{y^h}} p(\mathbf{y^o}, \mathbf{y^h}|\Theta). \tag{3}$$

While different variants of the Expectation-Maximization (EM) algorithm [71] relying on sampling $\mathbf{y^h}$ have been used to maximize the likelihood [3, 4, 6], these algorithms scale poorly with the number of hidden neurons.

Instead, we exploit that, for a homogeneous population, the fine-grained knowledge of the latent activity $\mathbf{y^h}$ is not necessary since all the observed neurons receive at time t the same input Jn_t, where $n_t = \sum_{i=1}^{N} y_t^i$ is the population activity of Section 3. Hence, we rewrite the likelihood (3) as

$$\mathcal{L} = p(\mathbf{y^o}|\Theta) = \sum_{\mathbf{n}} p(\mathbf{y^o}, \mathbf{n}|\Theta), \tag{4a}$$

where the probability $p(\mathbf{y^o}, \mathbf{n}|\Theta)$ factorizes in T terms of the form

$$p(\mathbf{y}_t^{\mathbf{o}}, n_t|\mathbf{y}_{1:t-1}^{\mathbf{o}}, \mathbf{n}_{1:t-1}, \Theta) = \underbrace{p(\mathbf{y}_t^{\mathbf{o}}|\mathbf{y}_{1:t-1}^{\mathbf{o}}, \mathbf{n}_{1:t-1}, \Theta)}_{\text{given by neuron model, Eq. (1)}} \quad \underbrace{p(n_t|\mathbf{n}_{1:t-1}, \Theta)}_{\text{approx. by meso. model, Eq. (2)}} . \tag{4b}$$

A comparison of Eqs. (3) and (4) shows that the high-dimensional latent activity $\mathbf{y^h}$ has been reduced to a low-dimensional mesoscopic description. Importantly, the q observed spike trains are conditionally independent given the population activity $\mathbf{n}$. While the conditional dependence structure implied by Eq. (4b) is typical of standard latent variable models of multi-neuronal recordings [33, 38, 72, 73], in our approach, the latent variable explicitly represents the population activity of the generative SNN and the parameters of the model are identical to those of the SNN. As the

latent population dynamics directly stem from neuronal dynamics, we call our LVM the neuronally-grounded latent variable model (neuLVM).

Since the nonlinearity and the non-Markovianity of Eq. (2) prevents us from using previous EM algorithms [33, 38, 72, 73], we fit the neuLVM via the Baum-Viterbi algorithm [74] (also known as Viterbi training or hard EM [75]), which alternates estimation (E) and maximization (M) step

E-step. $\widehat{\mathbf{n}}^n = \mathrm{argmax}_{\mathbf{n}} \log p(\mathbf{y^o}, \mathbf{n} | \widehat{\Theta}^{n-1})$,

M-step. $\widehat{\Theta}^n = \mathrm{argmax}_{\Theta} \log p(\mathbf{y^o}, \widehat{\mathbf{n}}^n | \Theta)$.

The estimated parameters $\widehat{\Theta}$ and the estimated latent population activity $\widehat{\mathbf{n}}$ are the result of many iterations of **E-step** and **M-step** (Appendix C). Note that the computational cost of this algorithm does not depend on the number of hidden neurons (it only depends on the number of populations).

5 Experimental results

5.1 Single homogenous population: SNN with metastable cluster states

Although seemingly simple, homogeneous populations of leaky integrate-and-fire (LIF) neurons without external stimulation are SNNs with a rich repertoire of population dynamics, including asynchronous states, synchronous states, and cluster states [7, 9]. In a m-cluster state (with $m \geq 2$), the population activity oscillates at a frequency m times higher than the average neuronal firing rate: a neuron spikes every m cycles on average; conversely, approximately N/m neurons fire in each cycle (N being to total number of neurons). Cluster states have therefore been described as 'higher harmonics' of the synchronous state (or 1-cluster state) [9, 76–78] where all neurons fire in synchrony.

In this set of experiments, we always consider the same network of 600 LIF neurons (Figure 2 A), where only the connectivity parameter J can vary. When initialized, at time 0, in the same unstable asynchronous state, the network can spontaneously settle in a m-cluster state, where m depends on the recurrent connectivity parameter J (Figure 2 B): finite-size fluctuations break the symmetry of the asynchronous state and the population splits into m groups of synchronized neurons. The cluster state to which the network converges can be read from the power spectrum of the neuronal spike trains (Figure 2 B) (the fundamental frequency of the m-cluster state is approximately m times lower than that of the 1-cluster state). Generating spike trains for 6 observed neurons (1% of the population), we tested whether neuLVM could recover the connectivity parameter J (neuronal parameters θ were given), for different J's in the 1-, 2-, and 3-cluster states range (Figure 2 C, Table S3). The Pearson correlation between the learned $\widehat{J}$ and the true J was 0.81 with p-value $2.8\mathrm{e}{-17}$, showing that, statistically, neuLVM could recover the connectivity parameter of the SNN.

To assess how well neuLVM can infer the latent population activity and how neuLVM compares with the methods assuming full observability (like René et al. [49]), we studied in detail a single trial showing a transition from a metastable 4-cluster state to a 3-cluster state (Figure 2 D,E). To generate this trial, we chose $J = 60.32$ and initialized the network in the 4-cluster state. From the spike trains of only two neurons (red stars in Figure 2 D), neuLVM could infer the ground truth population activity $\mathbf{n}^*$ during the 4-cluster state, and during the 3-cluster state, and could approximately detect the transition between the two states (Figure 2 E). While the summed, smoothed spike trains missed two out of four population activity peaks in the 4-cluster state, and one out of three peaks in the 3-cluster state (purple curve in Figure 2 E), the strong inductive biases contained in neuLVM still enabled the inference of the 'missing' peaks (blue curve in Figure 2 E). Finally, neuLVM and a method assuming full observability (equivalent to a naive application of René et al. [49]) were compared through their ability to recover the connectivity parameter J, for varying number of observed neurons (Figure 2 F, Table S4). Since their algorithm does not take into account hidden neurons, the method of René et al. [49] led to wildly inaccurate estimate $\widehat{J}$ when the summed spike train was far from the ground truth population activity (which happened when the number of observed neurons was small, see Figure 2 E for an example). In contrast, the neuLVM managed to recover the connectivity parameter, thanks to the fact that the Baum-Viterbi algorithm of Section 4 also infers the population activity (see Figure 2 E for an example).

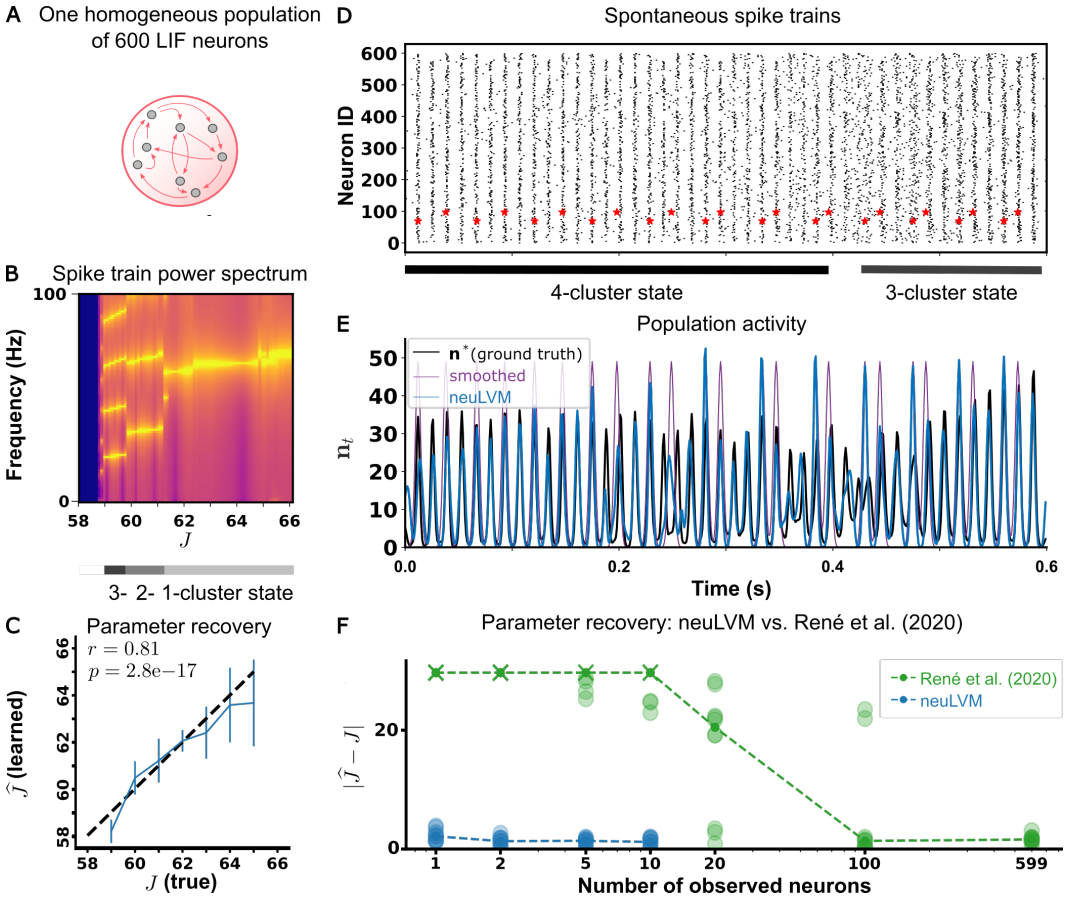

Figure 2: **Single-population SNN with metastable cluster states.** (A) Network architecture (for visualization purpose, only a few connections are drawn). (B) Spike train power spectrum for different choices of connectivity parameter J. All simulations start from the same unstable asynchronous state. The corresponding cluster states are indicated below. Blue around $J = 58$ indicates absence of activity. (C) Connectivity recovered by the neuLVM $\widehat{J}$ vs ground truth J. The neuLVM was fitted on one-second single-trials recordings of six neurons (1% of the population). For each ground truth J value (seven in total), ten different trials were generated: bars indicate the standard deviations of the recovered $\widehat{J}$. The Pearson correlation coefficient between the recovered $\widehat{J}$ and J is $r = 0.81$ and the associated p-value is $2.8\text{e}{-17}$ (see Table S3). (D) Spike trains generated by the ground truth SNN for a trial showing a transition from a metastable 4-cluster state to a 3-cluster state. The spike trains of two randomly sampled neurons (red stars) formed the training data (for visualization purpose, only the first 0.6 second of the one-second trial is shown) on which neuLVM was fitted: (E) the inferred population activity $\widehat{\mathbf{n}}|\mathbf{y^o}$ is compared to the ground truth $\mathbf{n}^*$ and the summed, smoothed spike trains (Gaussian smoothing window with $\sigma = 1.4$ ms) of the two observed neurons. (F) Absolute difference between the recovered $\widehat{J}$ and the ground truth J for the neuLVM algorithm and the method of René et al. (2020) for varying numbers of observed neurons. Using the same trial as in D, for each number of observed neurons, the two methods were tested on 10 different samples of observed neurons (see Table S4). '$\times$' indicates that the difference $|\widehat{J} - J|$ is larger than 30. The median samples are linked with dashed lines to show the trends.

5.2 Multiple populations: SNN with metastable point attractors

5.2.1 Latent population activity inference and reproduction of metastable dynamics

As a second benchmark, we tested neuLVM on synthetic data generated by three interacting populations with two populations of 400 excitatory LIF neurons and one population of 200 inhibitory neurons (Figure 3 A). Recurrent connections between these population drive winner-take-all dynamics with finite-size fluctuations-induced switches of activity between the two excitatory populations [11, 79] – an example of 'itinerancy between attractor states' [25]. The population activities of this metastable, three-population SNN constitute the ground truth against which different models will be tested.

To build a spiking benchmark dataset, we proceed in two steps. In a preprocessing step, we cluster the data to assign the observed neurons to homogeneous populations. To verify the feasibility of clustering, we take $M = 20$ samples of $q = 25$ neurons each, selected randomly (with replacement) from the 600 neurons. We low-pass filter the spike trains ($\tau_{\text{cluster}} = 400\,\text{ms}$) and use the van-Rossum-Distance [80] to evaluate pairwise distance. Searching for three clusters with k-means clustering [81], we find that in all $M = 20$ experiments, the three different clusters correspond to the three different populations so that all members of a given cluster represented neurons of the same population. Moreover, we found that no cluster contained less than three neurons. In a second step, we randomly selected only 9 neurons (3 neurons from each of the three populations) and consider the spike trains of these neurons as the observed data. The complete dataset consists of 20 trials of 10 seconds. An example trial is shown in Figure 3 B.

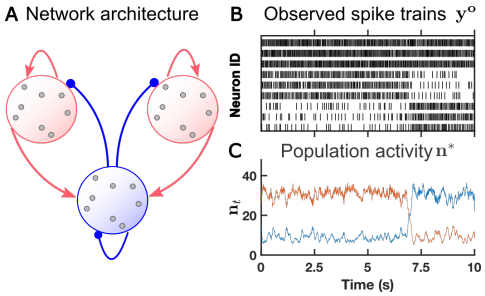

Figure 3: **Network architecture and example trial.** (A) Architecture of the three-population, metastable, winner-take-all SNN. (B) Example trial from the spiking benchmark dataset: 10 seconds recordings of 9 observed neurons (3 neurons from each of the three populations) and (C) corresponding ground truth latent population activity (for the two excitatory populations).

In contrast to the experiments of Section 5.1 where the neuronal parameters were given, here, neuronal and connectivity parameters are not given to neuLVM (see Appendix E). We compared the performance of neuLVM with other generative models of spiking data – PLDS [33], SLDS [34], and GLM [50–52] – on single trials of the spiking benchmark dataset in two ways: (i) we measured the Pearson correlation r between the inferred latent population activity $\widehat{\mathbf{n}}|\mathbf{y^o}$ and the ground truth population activity $\mathbf{n}^*$ (Table 1 first column); (ii) we assessed if the fitted models could reproduce metastable dynamics by counting the occurrences of stochastic switches in free simulations of the fitted models (Table 1 second column). Tests (i) and (ii) on an example trial are shown in Figure 4.

The Poisson Linear Dynamical Systems approach (PLDS, [33]) assumes that the recorded spikes can be explained by point processes driven by a latent linear dynamical system of low-dimension. The Poisson Switching Linear Dynamical System (SLDS, [34]) extends PLDS by allowing the latent variables to switch randomly between two dynamical systems with distinct parameters. We should stress that, in PLDS and SLDS, the latent variables are *phenomenological* representations of neural population activity which have no direct link with the ground truth population activity $\mathbf{n}^*$. In order to still make test (i) possible for PLDS and SLDS, we will consider the best linear transformation of the inferred latent variables which minimizes the mean squared error with the ground truth population activity $\mathbf{n}^*$.

On test (i), neuLVM gave better estimates $\widehat{\mathbf{n}}|\mathbf{y^o}$ of the latent population population activity $\mathbf{n}^*$ (Pearson correlation $r = 0.81$) than the best linear transformation of the latent activity inferred by PLDS and SLDS ($r = 0.69$ and $r = 0.73$ respectively) (Table 1 first column). The GLM approach cannot be included in test (i) since it ignores unobserved neurons. Interestingly, the example trial in Figure 4 A shows the latent population activity $\widehat{\mathbf{n}}|\mathbf{y^o}$ inferred by neuLVM is smoother than the ground truth $\mathbf{n}^*$ before and after the switch (finite-size fluctuations are reduced) but $\widehat{\mathbf{n}}|\mathbf{y^o}$ and $\mathbf{n}^*$

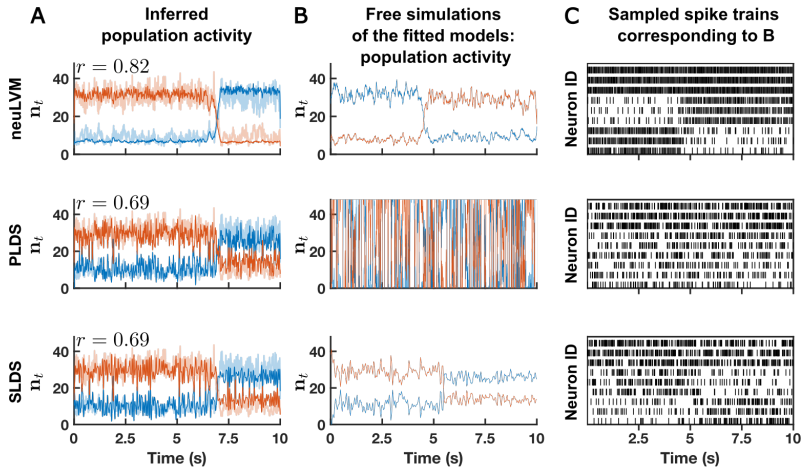

Figure 4: **Three-population SNN with metastable point attractors.** (A) Latent population activity of the two excitatory populations inferred by neuLVM / PLDS / SLDS for one example trial (the same as in Figure 3). The value r is the Pearson correlation coefficient between the inferred $\widehat{\mathbf{n}}|\mathbf{y^o}$ and the ground truth $\mathbf{n}^*$ population activities. (B-C) Examples of free simulations of the fitted neuLVM / PLDS / SLDS.

Table 1: Model performance summary (corresponding to Figure 4).

Models	Pearson correlation r between $\widehat{\mathbf{n}}\|\mathbf{y^o}$ and $\mathbf{n}^*$	Number of switches during 100 seconds free simulations of the fitted models (10.3 ± 2.7 for the ground truth SNN)
neuLVM	0.81 ± 0.02	7.8 ± 4.1
PLDS	(0.68 ± 0.11)	not visible
SLDS	(0.73 ± 0.02)	11.9 ± 8.9
GLM	-	not visible

Mean and ($\pm$) standard deviation were computed over 20 different trials. Parentheses for PLDS and SLDS indicate that these results are for the best linear transformation of the inferred latent variables.

closely match around the time of the switch. In contrast, fluctuations are exaggerated for PLDS and SLDS. The population activity estimated by simply summing and smoothing the observed spike trains is shown in Figure S6.

On test (ii), neuLVM, fitted on a single trial of 10 seconds, was able to reproduce stochastic switches similar to that of the ground truth SNN (Table 1 second column): free simulations of the fitted neuLVM showed 7.8 switches in 100 seconds on average (10.3 switches on average for the ground truth SNN). To make sure that stochastic switches were the result of parameter learning via the Baum-Viterbi algorithm, we verified that, before learning, neuLVM did not show any metastable dynamics (Figure S7). Examples of simulated trial are shown in Figure 4 B. PLDS failed to reproduce stochastic switches, which is not surprising since winner-take-all dynamics are typically nonlinear. SLDS could reproduce stochastic switches at the correct mean frequency (11.9 instead of the ground truth 10.3), but the standard deviation of the simulated switch count, 8.9 (2.7 for the ground truth SNN), indicates that a single 10 seconds trial was probable not sufficient for SLDS to learn switching probabilities reliably. Finally, neuronal stochasticity and small network size (9 neurons) did not allow GLM to produce stochastic switches, even when the training trial was prolonged to 500 seconds.

Taken together, only neuLVM could infer the latent population activity and reliably learn the metastable dynamics on single trials of 10 seconds, demonstrating the effectiveness of its neuronally-grounded inductive biases.

5.2.2 Generalization: towards experimental predictions with neuronally-grounded modeling

Bottom-up, mechanistic models allow us to perform *in silico* experiments and generate predictions about neural microcircuits, which can then be tested experimentally. So we wondered: can neuLVM,

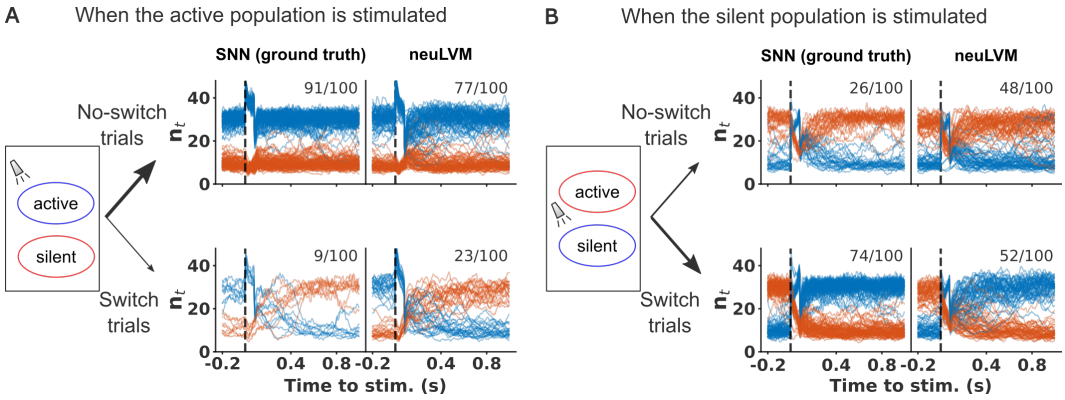

Figure 5: **Network responses to perturbations mimicking photo-stimulation.** (A) Activities of the excitatory populations when the active population is stimulated (100 trials, ratios indicate the number of No-switch or Switch trials). (B) Same as A but the silent population is stimulated.

fitted on a single trial of spontaneous activity (like in Section 5.2.1), predict the response of the SNN when an external perturbation is applied? As a preliminary step in that direction, we tested whether an external stimulation of the fitted model would generate the same response as that of the microscopic SNN when subjected to the same perturbation.

Using the same multi-population network as in Section 5.2 (Figure 3 A) and neuLVM fitted on a single trial of spontaneous activity (Figure 3 B), we compared the response of the ground truth SNN with that of neuLVM when one of the populations was stimulated by a current pulse of 4 ms mimicking a stimulation of a photo-genetically manipulated population by a short light pulse. We simulated 100 trials where the momentarily active excitatory population was stimulated, and 100 where the momentarily silent excitatory population was stimulated (Figure 5 A and B respectively). Each stimulation led to two possible outcomes: stimulation could trigger a state switch (Switch trials) or no state switch (No-switch trials). In both the ground truth SNN and the fitted neuLVM, we found that stimulating the silent population triggered more frequent state switches (Figure 5 B) than stimulating the active population (Figure 5 A). Moreover, in both the ground truth and the fitted neuLVM, we could induce 'excitatory rebound' switches by stimulating the active population (Figure 5 A, lower half).

6 Discussion

Understanding the neural dynamics underlying computation in the brain is one of the main goals of latent variable modeling of multi-neuronal recordings [43, 45, 46, 48, 82, 83]. We contribute to this effort by proposing here a bottom-up, mechanistic LVM – the neuronally-grounded latent variable model (neuLVM) – which can be mapped to a multi-population SNN. Using SNN-based generative models, which are more biological than RNN-based models [43], could allow systems neuroscientists to test hypothesis about the architecture of the probed microcircuit, and provide a neuronally-grounded understanding of computation. While this work shows the potential of the neuLVM approch, two methodological questions should be addressed before an application of the neurLVM on real data.

First, although the neuLVM could recover the connectivity parameter (Section 5.1), our method cannot always recover the SNN parameters when many parameters are unknown (Section 5.2). This is most likely due to the non-identifiablity of the generative SNN model when the external stimulus is constant. Using Bayesian inference to circumvent the problem of non-identifiability (as in [49]) is left for future work.

Second, in the case of real data, choosing the good generative SNN model is a nontrivial task. As discussed in Section 5.2, preprocessing the recorded spike trains could guide the design of the generative model. Note the same question is encountered in biophysical modeling of neuroimaging data and ideas could be borrowed from that field [84–86].

References

[1] Nicholas A Steinmetz, Christof Koch, Kenneth D Harris, and Matteo Carandini. Challenges and opportunities for large-scale electrophysiology with neuropixels probes. *Current opinion in neurobiology*, 50:92–100, 2018.

[2] Peiran Gao and Surya Ganguli. On simplicity and complexity in the brave new world of large-scale neuroscience. *Current opinion in neurobiology*, 32:148–155, 2015.

[3] Jonathan W Pillow and Peter Latham. Neural characterization in partially observed populations of spiking neurons. *Adv Neural Information Processing Systems*, 20(3.5), 2008.

[4] Johanni Brea, Walter Senn, and Jean-Pascal Pfister. Sequence learning with hidden units in spiking neural networks. *Advances in neural information processing systems*, 24, 2011.

[5] Danilo Rezende, Daan Wierstra, and Wulfram Gerstner. Variational learning for recurrent spiking networks. *Advances in neural information processing systems*, 24, 2011.

[6] Guillaume Bellec, Shuqi Wang, Alireza Modirshanechi, Johanni Brea, and Wulfram Gerstner. Fitting summary statistics of neural data with a differentiable spiking network simulator. *arXiv preprint arXiv:2106.10064*, 2021.

[7] Wulfram Gerstner. Time structure of the activity in neural network models. *Phys. Rev. E*, 51(1): 738, 1995.

[8] Nicolas Brunel and Vincent Hakim. Fast global oscillations in networks of integrate-and-fire neurons with low firing rates. *Neural computation*, 11(7):1621–1671, 1999.

[9] Wulfram Gerstner. Population dynamics of spiking neurons: fast transients, asynchronous states, and locking. *Neural Comput.*, 12(1):43–89, 2000.

[10] Wulfram Gerstner, Werner M Kistler, Richard Naud, and Liam Paninski. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.

[11] Tilo Schwalger, Moritz Deger, and Wulfram Gerstner. Towards a theory of cortical columns: From spiking neurons to interacting neural populations of finite size. *PLoS computational biology*, 13(4):e1005507, 2017.

[12] Tilo Schwalger and Anton V Chizhov. Mind the last spike—firing rate models for mesoscopic populations of spiking neurons. *Curr. Opin. Neurobiol.*, 58:155–166, 2019.

[13] Valentin Schmutz, Eva Löcherbach, and Tilo Schwalger. On a finite-size neuronal population equation. *arXiv preprint arXiv:2106.14721*, 2021.

[14] Sandrine Lefort, Christian Tomm, J-C Floyd Sarria, and Carl CH Petersen. The excitatory neuronal network of the c2 barrel column in mouse primary somatosensory cortex. *Neuron*, 61 (2):301–316, 2009.

[15] Tobias C Potjans and Markus Diesmann. The cell-type specific cortical microcircuit: relating structure and activity in a full-scale spiking network model. *Cerebral cortex*, 24(3):785–806, 2014.

[16] Yumiko Yoshimura, Jami LM Dantzker, and Edward M Callaway. Excitatory cortical neurons form fine-scale functional networks. *Nature*, 433(7028):868–873, 2005.

[17] Sen Song, Per Jesper Sjöström, Markus Reigl, Sacha Nelson, and Dmitri B Chklovskii. Highly nonrandom features of synaptic connectivity in local cortical circuits. *PLoS biology*, 3(3):e68, 2005.

[18] Claudia Clopath, Lars Büsing, Eleni Vasilaki, and Wulfram Gerstner. Connectivity reflects coding: a model of voltage-based stdp with homeostasis. *Nature neuroscience*, 13(3):344–352, 2010.

[19] Rodrigo Perin, Thomas K Berger, and Henry Markram. A synaptic organizing principle for cortical neuronal groups. *Proceedings of the National Academy of Sciences*, 108(13):5419–5424, 2011.

[20] Ho Ko, Sonja B Hofer, Bruno Pichler, Katherine A Buchanan, P Jesper Sjöström, and Thomas D Mrsic-Flogel. Functional specificity of local synaptic connections in neocortical networks. *Nature*, 473(7345):87–91, 2011.

[21] Ashok Litwin-Kumar and Brent Doiron. Slow dynamics and high variability in balanced cortical networks with clustered connections. *Nature neuroscience*, 15(11):1498–1505, 2012.

[22] Michael N Shadlen and William T Newsome. Noise, neural codes and cortical organization. *Current opinion in neurobiology*, 4(4):569–579, 1994.

[23] Michael N Shadlen and William T Newsome. The variable discharge of cortical neurons: implications for connectivity, computation, and information coding. *Journal of neuroscience*, 18(10):3870–3896, 1998.

[24] Saurabh Vyas, Matthew D Golub, David Sussillo, and Krishna V Shenoy. Computation through neural population dynamics. *Annual Review of Neuroscience*, 43:249–275, 2020.

[25] Paul Miller. Itinerancy between attractor states in neural systems. *Current opinion in neurobiology*, 40:14–22, 2016.

[26] Giancarlo La Camera, Alfredo Fontanini, and Luca Mazzucato. Cortical computations via metastable activity. *Current opinion in neurobiology*, 58:37–45, 2019.

[27] Braden AW Brinkman, Han Yan, Arianna Maffei, Il Memming Park, Alfredo Fontanini, Jin Wang, and Giancarlo La Camera. Metastable dynamics of neural circuits and networks. *Applied Physics Reviews*, 9(1):011313, 2022.

[28] Kenneth W Latimer, Jacob L Yates, Miriam LR Meister, Alexander C Huk, and Jonathan W Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making. *Science*, 349(6244):184–187, 2015.

[29] Tatiana A Engel, Nicholas A Steinmetz, Marc A Gieselmann, Alexander Thiele, Tirin Moore, and Kwabena Boahen. Selective modulation of cortical state during spatial attention. *Science*, 354(6316):1140–1144, 2016.

[30] Luca Mazzucato, Giancarlo La Camera, and Alfredo Fontanini. Expectation-induced modulation of metastable activity underlies faster coding of sensory stimuli. *Nature neuroscience*, 22(5): 787–796, 2019.

[31] Rubén Moreno-Bote, John Rinzel, and Nava Rubin. Noise-induced alternations in an attractor network model of perceptual bistability. *Journal of neurophysiology*, 98(3):1125–1139, 2007.

[32] Luca Mazzucato, Alfredo Fontanini, and Giancarlo La Camera. Dynamics of multistable states during ongoing and evoked cortical activity. *Journal of Neuroscience*, 35(21):8214–8231, 2015.

[33] Jakob H Macke, Lars Buesing, John P Cunningham, Byron M Yu, Krishna V Shenoy, and Maneesh Sahani. Empirical models of spiking in neural populations. In *Advances in Neural Information Processing Systems 24: 25th conference on Neural Information Processing Systems (NIPS 2011)*, pages 1350–1358, 2012.

[34] Scott Linderman, Matthew Johnson, Andrew Miller, Ryan Adams, David Blei, and Liam Paninski. Bayesian learning and inference in recurrent switching linear dynamical systems. In *Artificial Intelligence and Statistics*, pages 914–922. PMLR, 2017.

[35] J Nassar, SW Linderman, M Bugallo, and IM Park. Tree-structured recurrent switching linear dynamical systems for multi-scale modeling. In *International Conference on Learning Representations (ICLR)*, 2019.

[36] Joshua Glaser, Matthew Whiteway, John P Cunningham, Liam Paninski, and Scott Linderman. Recurrent switching dynamical systems models for multiple interacting neural populations. *Advances in neural information processing systems*, 33:14867–14878, 2020.

[37] David Zoltowski, Jonathan Pillow, and Scott Linderman. A general recurrent state space framework for modeling neural dynamics during decision-making. In *International Conference on Machine Learning*, pages 11680–11691. PMLR, 2020.

[38] Jayant E Kulkarni and Liam Paninski. Common-input models for multiple neural spike-train data. *Network: Computation in Neural Systems*, 18(4):375–407, 2007.

[39] Vernon Lawhern, Wei Wu, Nicholas Hatsopoulos, and Liam Paninski. Population decoding of motor cortical activity using a generalized linear model with hidden states. *Journal of neuroscience methods*, 189(2):267–280, 2010.

[40] Michael Vidne, Yashar Ahmadian, Jonathon Shlens, Jonathan W Pillow, Jayant Kulkarni, Alan M Litke, EJ Chichilnisky, Eero Simoncelli, and Liam Paninski. Modeling the impact of common noise inputs on the network activity of retinal ganglion cells. *Journal of computational neuroscience*, 33(1):97–121, 2012.

[41] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical neural population models through nonlinear embeddings. *Advances in neural information processing systems*, 29, 2016.

[42] Anqi Wu, Nicholas A Roy, Stephen Keeley, and Jonathan W Pillow. Gaussian process based nonlinear latent structure discovery in multivariate spike train data. *Advances in neural information processing systems*, 30, 2017.

[43] Chethan Pandarinath, Daniel J O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky, Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R Hochberg, et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature methods*, 15(10):805–815, 2018.

[44] Lea Duncker and Maneesh Sahani. Temporal alignment and latent gaussian process factor inference in population spike trains. *bioRxiv*, page 331751, 2018.

[45] Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable continuous-time models of latent stochastic dynamical systems. In *International Conference on Machine Learning*, pages 1726–1734. PMLR, 2019.

[46] Virginia Rutten, Alberto Bernacchia, Maneesh Sahani, and Guillaume Hennequin. Non-reversible gaussian processes for identifying latent dynamical structure in neural data. *Advances in Neural Information Processing Systems*, 2020.

[47] Stephen Keeley, Mikio Aoi, Yiyi Yu, Spencer Smith, and Jonathan W Pillow. Identifying signal and noise structure in neural population activity with gaussian process factor models. *Advances in Neural Information Processing Systems*, 33:13795–13805, 2020.

[48] Timothy D Kim, Thomas Z Luo, Jonathan W Pillow, and Carlos Brody. Inferring latent dynamics underlying neural population activity via neural differential equations. In *International Conference on Machine Learning*, pages 5551–5561. PMLR, 2021.

[49] Alexandre René, André Longtin, and Jakob H Macke. Inference of a mesoscopic population model from population spike trains. *Neural computation*, 32(8):1448–1498, 2020.

[50] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.

[51] Wilson Truccolo, Uri T Eden, Matthew R Fellows, John P Donoghue, and Emery N Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *J. Neurophysiol.*, 93(2):1074–1089, 2005.

[52] Jonathan W Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, EJ Chichilnisky, and Eero P Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

[53] David Zoltowski and Jonathan W Pillow. Scaling the poisson glm to massive neural datasets through polynomial approximations. *Advances in neural information processing systems*, 31, 2018.

[54] Regis C Lambert, Christine Tuleau-Malot, Thomas Bessaih, Vincent Rivoirard, Yann Bouret, Nathalie Leresche, and Patricia Reynaud-Bouret. Reconstructing the functional connectivity of multiple spike trains using Hawkes models. *J. Neurosci. Methods*, 297:9–21, 2018.

[55] Ryota Kobayashi, Shuhei Kurita, Anno Kurth, Katsunori Kitano, Kenji Mizuseki, Markus Diesmann, Barry J Richmond, and Shigeru Shinomoto. Reconstructing neuronal circuitry from parallel spike trains. *Nature communications*, 10(1):1–13, 2019.

[56] Werner M Kistler, Wulfram Gerstner, and J Leo van Hemmen. Reduction of the hodgkin-huxley equations to a single-variable threshold model. *Neural computation*, 9(5):1015–1045, 1997.

[57] Renaud Jolivet, Timothy J Lewis, and Wulfram Gerstner. Generalized integrate-and-fire models of neuronal activity approximate spike trains of a detailed model to a high degree of accuracy. *Journal of neurophysiology*, 92(2):959–976, 2004.

[58] Romain Brette and Wulfram Gerstner. Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *Journal of neurophysiology*, 94(5):3637–3642, 2005.

[59] Renaud Jolivet, Alexander Rauch, Hans-Rudolf Lüscher, and Wulfram Gerstner. Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *Journal of computational neuroscience*, 21(1):35–49, 2006.

[60] Ryota Kobayashi, Yasuhiro Tsubo, and Shigeru Shinomoto. Made-to-order spiking neuron model equipped with a multi-timescale adaptive threshold. *Frontiers in computational neuroscience*, 3:9, 2009.

[61] Christian Pozzorini, Skander Mensi, Olivier Hagens, Richard Naud, Christof Koch, and Wulfram Gerstner. Automated high-throughput characterization of single neurons by means of simplified spiking models. *PLoS Comput Biol*, 11(6):e1004275, 2015.

[62] Corinne Teeter, Ramakrishnan Iyer, Vilas Menon, Nathan Gouwens, David Feng, Jim Berg, Aaron Szafer, Nicholas Cain, Hongkui Zeng, Michael Hawrylycz, et al. Generalized leaky integrate-and-fire models classify multiple neuron types. *Nature communications*, 9(1):1–15, 2018.

[63] David Roxbee Cox. *Renewal theory*. Methuen, 1962.

[64] Hugh R Wilson and Jack D Cowan. Excitatory and inhibitory interactions in localized populations of model neurons. *Biophys. J.*, 12(1):1–24, 1972.

[65] Anna De Masi, Antonio Galves, Eva Löcherbach, and Errico Presutti. Hydrodynamic limit for interacting neurons. *J. Stat. Phys.*, 158(4):866–902, 2015. ISSN 0022-4715.

[66] Nicolas Fournier and Eva Löcherbach. On a toy model of interacting neurons. In *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, volume 52, pages 1844–1876. Institut Henri Poincaré, 2016.

[67] Julien Chevallier. Mean-field limit of generalized Hawkes processes. *Stochastic Process. Appl.*, 127(12):3870–3912, 2017. ISSN 0304-4149.

[68] Julien Chevallier et al. Fluctuations for mean-field interacting age-dependent hawkes processes. *Electronic Journal of Probability*, 22, 2017.

[69] Grégory Dumont, Alexandre Payeur, and André Longtin. A stochastic-field description of finite-size spiking neural networks. *PLoS computational biology*, 13(8):e1005691, 2017.

[70] Valentin Schmutz, Wulfram Gerstner, and Tilo Schwalger. Mesoscopic population equations for spiking neural networks with synaptic short-term plasticity. *The Journal of Mathematical Neuroscience*, 10(1):1–32, 2020.

[71] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[72] Anne C Smith and Emery N Brown. Estimating a state-space model from point process observations. *Neural computation*, 15(5):965–991, 2003.

[73] Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Journal of neurophysiology*, 102(1):614–635, 2009.

[74] Yariv Ephraim and Neri Merhav. Hidden markov processes. *IEEE Transactions on information theory*, 48(6):1518–1569, 2002.

[75] Armen Allahverdyan and Aram Galstyan. Comparative analysis of viterbi training and maximum likelihood estimation for hmms. *Advances in Neural Information Processing Systems*, 24, 2011.

[76] Wulfram Gerstner and J Leo van Hemmen. Coherence and incoherence in a globally coupled ensemble of pulse-emitting units. *Physical review letters*, 71(3):312, 1993.

[77] David Golomb and John Rinzel. Clustering in globally coupled inhibitory neurons. *Physica D: Nonlinear Phenomena*, 72(3):259–282, 1994.

[78] U Ernst, K Pawelzik, and T Geisel. Synchronization induced by temporal delays in pulse-coupled oscillators. *Physical review letters*, 74(9):1570, 1995.

[79] Kong-Fatt Wong and Xiao-Jing Wang. A recurrent network mechanism of time integration in perceptual decisions. *Journal of Neuroscience*, 26(4):1314–1328, 2006.

[80] Mark CW van Rossum. A novel spike distance. *Neural computation*, 13(4):751–763, 2001.

[81] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28 (2):129–137, 1982.

[82] Yuan Zhao and Il Memming Park. Interpretable nonlinear dynamic modeling of neural trajectories. *Advances in neural information processing systems*, 29, 2016.

[83] Matthew R Whiteway and Daniel A Butts. The quest for interpretable models of neural population activity. *Current opinion in neurobiology*, 58:86–93, 2019.

[84] William D Penny, Klaas E Stephan, Andrea Mechelli, and Karl J Friston. Comparing dynamic causal models. *Neuroimage*, 22(3):1157–1172, 2004.

[85] Will D Penny, Klaas E Stephan, Jean Daunizeau, Maria J Rosa, Karl J Friston, Thomas M Schofield, and Alex P Leff. Comparing families of dynamic causal models. *PLoS computational biology*, 6(3):e1000709, 2010.

[86] Kay H Brodersen, Thomas M Schofield, Alexander P Leff, Cheng Soon Ong, Ekaterina I Lomakina, Joachim M Buhmann, and Klaas E Stephan. Generative embedding for model-based classification of fmri data. *PLoS computational biology*, 7(6):e1002079, 2011.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code URL available upon acceptance or upon request.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes]

(b) Did you mention the license of the assets? [N/A]

(c) Did you include any new assets either in the supplemental material or as a URL? [N/A]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]