Neural Dynamics as Sampling: A Model for Stochastic Computation in Recurrent Networks of Spiking Neurons

Lars Buesing[¤]*, **Johannes Bill, Bernhard Nessler, Wolfgang Maass**

Institute for Theoretical Computer Science, Graz University of Technology, Graz, Austria

Abstract

The organization of computations in networks of spiking neurons in the brain is still largely unknown, in particular in view of the inherently stochastic features of their firing activity and the experimentally observed trial-to-trial variability of neural systems in the brain. In principle there exists a powerful computational framework for stochastic computations, probabilistic inference by sampling, which can explain a large number of macroscopic experimental data in neuroscience and cognitive science. But it has turned out to be surprisingly difficult to create a link between these abstract models for stochastic computations and more detailed models of the dynamics of networks of spiking neurons. Here we create such a link and show that under some conditions the stochastic firing activity of networks of spiking neurons can be interpreted as probabilistic inference via Markov chain Monte Carlo (MCMC) sampling. Since common methods for MCMC sampling in distributed systems, such as Gibbs sampling, are inconsistent with the dynamics of spiking neurons, we introduce a different approach based on non-reversible Markov chains that is able to reflect inherent temporal processes of spiking neuronal activity through a suitable choice of random variables. We propose a neural network model and show by a rigorous theoretical analysis that its neural activity implements MCMC sampling of a given distribution, both for the case of discrete and continuous time. This provides a step towards closing the gap between abstract functional models of cortical computation and more detailed models of networks of spiking neurons.

Citation: Buesing L, Bill J, Nessler B, Maass W (2011) Neural Dynamics as Sampling: A Model for Stochastic Computation in Recurrent Networks of Spiking Neurons. PLoS Comput Biol 7(11): e1002211. doi:10.1371/journal.pcbi.1002211

Editor: Olaf Sporns, Indiana University, United States of America

Received March 16, 2011; **Accepted** August 10, 2011; **Published** November 3, 2011

Funding: This paper was written under partial support by the European Union project #FP7-237955 (FACETS-ITN), project #FP7-269921 (BrainScaleS), project #FP7-216593 (SECO), project #FP7-506778 (PASCAL2) and project #FP7-243914 (BRAIN-I-NETS). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: lars@gatsby.ucl.ac.uk

¤ Current address: Gatsby Computational Neuroscience Unit, University College London, London, United Kingdom

Introduction

Attempts to understand the organization of computations in the brain from the perspective of traditional, mostly deterministic, models of computation, such as attractor neural networks or Turing machines, have run into problems: Experimental data suggests that neurons, synapses, and neural systems are inherently *stochastic* [1], especially in vivo, and therefore seem less suitable for implementing deterministic computations. This holds for ion channels of neurons [2], synaptic release [3], neural response to stimuli (trial-to-trial variability) [4,5], and perception [6]. In fact, several experimental studies arrive at the conclusion that external stimuli only modulate the highly stochastic spontaneous firing activity of cortical networks of neurons [7,8]. Furthermore, traditional models for neural computation have been challenged by the fact that typical sensory data from the environment is often noisy and ambiguous, hence requiring neural systems to take *uncertainty* about external inputs into account. Therefore many researchers have suggested that information processing in the brain carries out probabilistic, rather than logical, inference for making decisions and choosing actions [9–22]. Probabilistic inference has emerged in the 1960's [23], as a principled mathematical framework for reasoning in the face of uncertainty with regard to observations, knowledge, and causal relationships, which is characteristic for real-world inference tasks. This framework has become tremendously successful in real-world applications of artificial intelligence and machine learning. A typical computation that needs to be carried out for probabilistic inference on a high-dimensional joint distribution $p(z_1,\ldots,z_l,z_{l+1},\ldots,z_K)$ is the evaluation of the conditional distribution $p(z_1,\ldots,z_l|z_{l+1},\ldots,z_K)$ (or marginals thereof) over some variables of interest, say $z_1,\ldots,z_l$, given variables $z_{l+1},\ldots,z_K$. In the following, we will call the set of variables $z_{l+1},\ldots,z_K$, which we condition on, the *observed* variables and denote it by $\boldsymbol{o}$.

Numerous studies in different areas of neuroscience and cognitive science have suggested that probabilistic inference could explain a variety of computational processes taking place in neural systems (see [10,11]). In models of perception the observed variables $\boldsymbol{o}$ are interpreted as the sensory input to the central nervous system (or its early representation by the firing response of neurons, e.g., in the LGN in the case of vision), and the variables $z_1,\ldots,z_l$ model the interpretation of the sensory input, e.g., the texture and position of objects in the case of vision, which might be encoded in the response of neurons in various higher cortical areas [15]. Furthermore, in models for motor control the observed variables $\boldsymbol{o}$ often consist not only of sensory and proprioceptive inputs to the brain, but also of specific goals and constraints for a planned movement [24–26], whereas inference is carried out over

Author Summary

It is well-known that neurons communicate with short electric pulses, called action potentials or spikes. But how can spiking networks implement complex computations? Attempts to relate spiking network activity to results of deterministic computation steps, like the output bits of a processor in a digital computer, are conflicting with findings from cognitive science and neuroscience, the latter indicating the neural spike output in identical experiments changes from trial to trial, i.e., neurons are "unreliable". Therefore, it has been recently proposed that neural activity should rather be regarded as *samples* from an underlying *probability distribution* over many variables which, e.g., represent a model of the external world incorporating prior knowledge, memories as well as sensory input. This hypothesis assumes that networks of stochastically spiking neurons are able to emulate powerful algorithms for reasoning in the face of uncertainty, i.e., to carry out probabilistic inference. In this work we propose a detailed neural network model that indeed fulfills these computational requirements and we relate the spiking dynamics of the network to concrete probabilistic computations. Our model suggests that neural systems are suitable to carry out probabilistic inference by using stochastic, rather than deterministic, computing elements.

the variables $z_1,\ldots,z_l$ representing a motor plan or motor commands to muscles. Recent publications show that human reasoning and learning can also be cast into the form of probabilistic inference problems [27–29]. In these models learning of concepts, ranging from concrete to more abstract ones, is interpreted as inference in lower and successively higher levels of hierarchical probabilistic models, giving a consistent description of inductive learning within and across domains of knowledge.

In spite of this active research on the functional level of neural processing, it turned out to be surprisingly hard to relate the computational machinery required for probabilistic inference to experimental data on neurons, synapses, and neural systems. There are mainly two different approaches for implementing the computational machinery for probabilistic inference in "neural hardware". The first class of approaches builds on deterministic methods for evaluating exactly or approximately the desired conditional and/or marginal distributions, whereas the second class relies on sampling from the probability distributions in question. Multiple models in the class of deterministic approaches implement algorithms from machine learning called message passing or belief propagation [30–33]. By clever reordering of sum and product operators occurring in the evaluation of the desired probabilities, the total number of computation steps are drastically reduced. The results of subcomputations are propagated as "messages" or "beliefs" that are sent to other parts of the computational network. Other deterministic approaches for representing distributions and performing inference are probabilistic population code (PPC) models [34]. Although deterministic approaches provide a theoretically sound hypothesis about how complex computations can possibly be embedded in neural networks and explain aspects of experimental data, it seems difficult (though not impossible) to conciliate them with other aspects of experimental evidence, such as stochasticity of spiking neurons, spontaneous firing, trial-to-trial variability, and perceptual multistability.

Therefore other researchers (e.g., [16–18,35]) have proposed to model computations in neural systems as probabilistic inference based on a different class of algorithms, which requires stochastic, rather than deterministic, computational units. This approach, commonly referred to as sampling, focuses on drawing *samples*, i.e., concrete values for the random variables that are distributed according to the desired probability distribution. Sampling can naturally capture the effect of apparent stochasticity in neural responses and seems to be furthermore consistent with multiple experimental effects reported in cognitive science literature [17,18]. On the conceptual side, it has proved to be difficult to implement learning in message passing and PPC network models. In contrast, following the lines of [36], the sampling approach might be well suited to incorporate learning.

Previous network models that implement sampling in neural networks are mostly based on a special sampling algorithm called Gibbs (or general Metropolis-Hastings) sampling [9,17,18,37]. The dynamics that arise from this approach, the so-called Glauber dynamics, however are only superficially similar to spiking neural dynamics observed in experiments, rendering these models rather abstract. Building on and extending previous models, we propose here a family of network models, that can be shown to exactly sample from any arbitrary member of a well-defined class of probability distributions via their inherent network dynamics. These dynamics incorporate refractory effects and finite durations of postsynaptic potentials (PSPs), and are therefore more biologically realistic than existing approaches. Formally speaking, our model implements Markov chain Monte Carlo (MCMC) sampling in a spiking neural network. In contrast to prior approaches however, our model incorporates irreversible dynamics (i.e., no detailed balance) allowing for finite time PSPs and refractory mechanisms. Furthermore, we also present a continuous time version of our network model. The resulting stochastic dynamical system can be shown to sample from the correct distribution. In general, continuous time models arguably provide a higher amount of biological realism compared to discrete time models.

The paper is structured in the following way. First we provide a brief introduction to MCMC sampling. We then define the neural network model whose neural activity samples from a given class of probability distributions. The model will be first presented in discrete time together with some illustrative simulations. An extension of the model to networks of more detailed spiking neuron models which feature a relative refractory mechanism is presented. Furthermore, it is shown how the neural network model can also be formulated in continuous time. Finally, as a concrete simulation example we present a simple network model for perceptual multistability.

Results

Recapitulation of MCMC sampling

In machine learning, sampling is often considered the "gold standard" of inference methods, since, assuming that we can sample from the distribution in question, and assuming enough computational resources, any inference task can be carried out with arbitrary precision (in contrast to some deterministic approximate inference methods such as variational inference). However sampling from an arbitrary distribution can be a difficult problem in itself, as, e.g., many distributions can only be evaluated modulo a global constant (the partition function). In order to circumvent these problems, elaborate MCMC sampling techniques have been developed in machine learning and statistics [38]. MCMC algorithms are based on the following idea: instead of producing an ad-hoc sample, a process that is heuristically comparable to a global search over the whole state space of the random variables, MCMC methods produce a new sample via a “local search” around a point in the state space that is already (approximately) a sample from the distribution.

More formally, a Markov chain M (in discrete time) is defined by a set S of states (we consider for discrete time only the case where S has a finite size, denoted by $|S|$) together with a transition operator T. The operator T is a conditional probability distribution $T(s|s')$ over the next state s given a preceding state s'. The Markov chain M is started in some initial state $s(0)$, and moves through a trajectory of states $s(t)$ via iterated application of the stochastic transition operator T. More precisely, if $s(t-1)$ is the state at time $t-1$, then the next state $s(t)$ is drawn from the conditional probability distribution $T(s|s(t-1))$. An important theorem from probability theory (see, e.g., p. 232 in [39]) states that if M is irreducible (i.e., any state in S can be reached from any other state in S in finitely many steps with probability >0) and aperiodic (i.e., its state transitions cannot be trapped in deterministic cycles), then the probability $p(s(t)=s|s(0))$ converges for $t\to\infty$ to a probability $p(s)$ that does not depend on the initial state $s(0)$. This state distribution p is called the invariant distribution of M. The irreducibility of M implies that it is the only distribution over the states S that is invariant under its transition operator T, i.e.

$$p(s)=\sum_{s'\in S}T(s|s')\cdot p(s'). \tag{1}$$

Thus, in order to carry out probabilistic inference for a given distribution p, it suffices to construct an irreducible and aperiodic Markov chain M that leaves p invariant, i.e., satisfies equation (1). Then one can answer numerous probabilistic inference questions regarding p without any numerical computations of probabilities. Rather, one plugs in the observed values for some of the random variables (RVs) and simply collects samples from the conditional distribution over the other RVs of interest when the Markov chain approaches its invariant distribution.

A convenient and popular method for the construction of an operator T for a given distribution p is looking for operators T that satisfy the following detailed balance condition,

$$T(s|s')\cdot p(s')=T(s'|s)\cdot p(s) \tag{2}$$

for all $s,s'\in S$. A Markov chain that satisfies (2) is said to be reversible. In particular, the Gibbs and Metropolis-Hastings algorithms employ reversible Markov chains. A very useful property of (2) is that it implies the invariance property (1), and this is in fact the standard method for proving (1). However, as our approach makes use of irreversible Markov chains as explained below, we will have to prove (1) directly.

Neural sampling

Let $p(z_1,\ldots,z_K)$ be some arbitrary joint distribution over K binary variables $z_1,\ldots,z_K$ that only takes on values >0. We will show that under a certain computability assumption on p a network $\mathcal{N}$ consisting of K spiking neurons $\nu_1,\ldots,\nu_K$ can sample from p using its inherent stochastic dynamics. More precisely, we show that the stochastic firing activity of $\mathcal{N}$ can be viewed as a non-reversible Markov chain that samples from the given probability distribution p. If a subset $\boldsymbol{o}$ of the variables are observed, modelled as the corresponding neurons being “clamped” to the observed values, the remaining network samples from the conditional distribution of the remaining variables given the observables. Hence, this approach offers a quite natural implementation of probabilistic inference. It is similar to sampling approaches which have already been applied extensively, e.g., in Boltzmann machines, however our model is more biologically realistic as it incorporates aspects of the inherent temporal dynamics and spike-based communication of a network of spiking neurons. We call this approach *neural sampling* in the remainder of the paper.

In order to enable a network $\mathcal{N}$ of spiking neurons to sample from a distribution $p(z_1,\ldots,z_K)$ of binary variables z_k, one needs to specify how an assignment $(z_1,\ldots,z_K)\in\{0,1\}^K$ of values to these binary variables can be represented by the spiking activity of the network $\mathcal{N}$ and vice versa. A spike, or action potential, of a biological neuron ν_k has a short duration of roughly 1 ms. But the effect of such spike, both on the neuron ν_k itself (in the form of refractory processes) and on the membrane potential of other neurons (in the form of postsynaptic potentials) lasts substantially longer, on the order of 5 ms to 100 ms. In order to capture this temporally extended effect of each spike, we fix some parameter τ that models the average duration of these temporally extended processes caused by a spike. We say that a binary vector $(z_1,\ldots,z_K)$ is represented by the firing activity of the network $\mathcal{N}$ at time t for $k=1,\ldots,K$ iff:

$$z_k(t)=1 \Leftrightarrow \nu_k \text{ has fired within the time interval } (t-\tau,t]. \tag{3}$$

In other words, any spike of neuron ν_k sets the value of the associated binary variable z_k to 1 for a duration of length τ.

An obvious consequence of this definition is that the binary vector $(z_1,\ldots,z_K)$ that is defined by the activity of $\mathcal{N}$ at time t does not fully capture the internal state of this stochastic system. Rather, one needs to take into account additional non-binary variables $(\zeta_1,\ldots,\zeta_K)$, where the value of ζ_k at time t specifies *when* within the time interval $(t-\tau,t]$ the neuron ν_k has fired (if it has fired within this time interval, thereby causing $z_k=1$ at time t). The neural sampling process has the Markov property only with regard to these more informative auxiliary variables $\zeta_1,\ldots,\zeta_K$. Therefore our analysis of neural sampling will focus on the temporal evolution of these auxiliary variables. We adopt the convention that each spike of neuron ν_k sets the value of ζ_k to its maximal value τ, from which it linearly decays back to 0 during the subsequent time interval of length τ.

For the construction of the sampling network $\mathcal{N}$, we assume that the membrane potential $u_k(t)$ of neuron ν_k at time t equals the log-odds of the corresponding variable z_k to be active, and refer to this property as *neural computability condition*:

$$u_k(t)=\log\frac{p(z_k=1|\mathbf{z}_{\setminus k})}{p(z_k=0|\mathbf{z}_{\setminus k})}, \tag{4}$$

where we write z_k for $z_k(t)$ and $\mathbf{z}_{\setminus k}$ for the current values $z_i(t)$ of all other variables z_i with $i\neq k$. Under the assumption we make in equation (4), i.e., that the neural membrane potential reflects the log-odds of the corresponding variable z_k, it is required that each single neuron in the network can actually compute the right-hand side of equation (4), i.e., that it fulfills the neural computability condition.

A concrete class of probability distributions, that we will use as an example in the remainder, are Boltzmann distributions:

$$p(\mathbf{z})=\frac{1}{Z}\exp\left(\sum_{i,j}\frac{1}{2}W_{ij}z_iz_j+\sum_i b_iz_i\right) \tag{5}$$

with arbitrary real valued parameters b_i, W_{ij} which satisfy $W_{ij} = W_{ji}$ and $W_{ii} = 0$ (the constant Z ensures the normalization of $p(\mathbf{z})$). For the Boltzmann distribution, condition (4) is satisfied by neurons v_k with the standard membrane potential

$$u_k(t) = b_k + \sum_{i=1}^{K} W_{ki} z_i(t), \tag{6}$$

where b_k is the bias of neuron v_k (which regulates its excitability), W_{ki} is the strength of the synaptic connection from neuron v_i to v_k, and $W_{ki} z_i(t)$ approximates the time course of the postsynaptic potential in neuron v_k caused by a firing of neuron v_i with a constant signal of duration τ (i.e., a square pulse). As we will describe below, spikes of neuron v_k are evoked stochastically depending on the current membrane potential u_k and the auxiliary variable ζ_k.

The neural computability condition (4) links classes of probability distributions to neuron and synapse models in a network of spiking neurons. As shown above, Boltzmann distributions satisfy the condition if one considers point neuron models which compute a linear weighted sum of the presynaptic inputs. The class of distributions can be extended to include more complex distributions using a method proposed in [40] which is based on the following idea. Neuron v_k representing the variable z_k is not directly influenced by the activities $\mathbf{z}_{\backslash k}$ of the presynaptic neurons, but via intermediate nonlinear preprocessing elements. This preprocessing might be implemented by dendrites or other (inter-) neurons and is assumed to compute nonlinear combinations of the presynaptic activities $\mathbf{z}_{\backslash k}$ (similar to a kernel). This allows the membrane potential u_k, and therefore the log-odds ratio on the right-hand side of (4), to represent a more complex function of the activities $\mathbf{z}_{\backslash k}$, giving rise to more complex joint distributions $p(\mathbf{z})$. The concrete implementation of non-trivial directed and undirected graphical models with the help of preprocessing elements in the neural sampling framework is subject of current research. For the examples given in this study, we focus on the standard form of the membrane potential (6) of point neurons. As shown below, these spiking network models can emulate any Boltzmann machine (BM) [36].

A substantial amount of preceding studies has demonstrated that BMs are very powerful, and that the application of suitable learning algorithms for setting the weights W_{ij} makes it possible to learn and represent complex sensory processing tasks by such distributions [37,41]. In applications in statistics and machine learning using such Boltzmann distributions, sampling is typically implemented by Gibbs sampling or more general *reversible* MCMC methods. However, it is difficult to model some neural processes, such as an absolute refractory period or a postsynaptic potential (PSP) of fixed duration, using a reversible Markov chain, but they are more conveniently modelled using an irreversible one. As we wish to keep the computational power of BMs and at the same time to augment the sampling procedure with aspects of neural dynamics (such as PSPs with fixed durations, refractory mechanisms) to increase biological realism, we focus in the following on irreversible MCMC methods (keeping in mind that this might not be the only possible way to achieve these goals).

Neural sampling in discrete time

Here we describe neural dynamics in discrete time with an absolute refractory period τ. We interpret one step of the Markov chain as a time step dt in biological real time. The dynamics of the variable ζ_k, that describes the time course of the effect of a spike of neuron v_k, are defined in the following way. ζ_k is set to the value τ when neuron v_k fires, and decays by 1 at each subsequent discrete time step. The parameter τ is chosen to be some integer, so that ζ_k decays back to 0 in exactly τ time steps. The neuron can only spike (with a probability that is a function of its current membrane potential u_k) if its variable $\zeta_k \leq 1$. If however, $\zeta_k > 1$, the neuron is considered refractory and it cannot spike, but its ζ_k is reduced by 1 per time step. To show that these simple dynamics do indeed sample from the given distribution $p(\mathbf{z})$, we proceed in the following way. We define a joint distribution $p(\boldsymbol{\zeta}, \mathbf{z})$ which has the desired marginal distribution $\sum_{\boldsymbol{\zeta}} p(\boldsymbol{\zeta}, \mathbf{z}) = p(\mathbf{z})$. Further we formalize the dynamics informally described above as a transition operator T operating on the state vector $(\boldsymbol{\zeta}, \mathbf{z})$. Finally, in the Methods section, we show that $p(\boldsymbol{\zeta}, \mathbf{z})$ is the unique invariant distribution of this operator T, i.e., that the dynamics described by T produce samples $\mathbf{z}$ from the desired distribution $p(\mathbf{z})$. We refer to sampling through networks with this stochastic spiking mechanism as *neural sampling with absolute refractory period* due to the persistent refractory process.

Given the distribution $p(\mathbf{z})$ that we want to sample from, we define the following joint distribution $p(\boldsymbol{\zeta}, \mathbf{z})$ over the neural variables:

$$p(\boldsymbol{\zeta}, \mathbf{z}) := p(\boldsymbol{\zeta}|\mathbf{z}) \cdot p(\mathbf{z}) \text{ with } p(\boldsymbol{\zeta}|\mathbf{z}) := \prod_{k=1}^{K} p(\zeta_k|z_k)$$
$$\text{where} \quad p(\zeta_k|z_k) := \begin{cases} \tau^{-1} & \text{for} \quad z_k = 1 \ \wedge \ \zeta_k > 0 \\ 1 & \text{for} \quad z_k = 0 \ \wedge \ \zeta_k = 0 \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

This definition of $p(\zeta_k|z_k)$ simply expresses that if $z_k = 1$, then the auxiliary variable ζ_k can assume any value in $\{1, 2, \ldots, \tau\}$ with equal probability. On the other hand ζ_k necessarily assumes the value 0 if $z_k = 0$ (i.e., when the neuron is in its resting state).

The state transition operator T can be defined in a transparent manner as a composition of K transition operators, $T = T^1 \circ \ldots \circ T^K$, where T^k only updates the variables ζ_k and z_k of neuron v_k, i.e., the neurons are updated sequentially in the same order (this severe restriction will become obsolete in the case of continuous time discussed below). We define the composition as $(T^k \circ T^l)(\cdot) = (T^k(T^l(\cdot))$, i.e., T^l is applied prior to T^k. The new values of ζ_k and z_k only depend on the previous value ζ'_k and on the current membrane potential $u_k(\mathbf{z}_{\backslash k})$. The interesting dynamics take place in the variable ζ_k. They are illustrated in Figure 1 where the arrows represent transition probabilities greater than 0.

If the neuron v_k is not refractory, i.e., $\zeta'_k \leq 1$, it can spike (i.e., a transition from $\zeta'_k \leq 1$ to $\zeta_k = \tau$) with probability

$$T^k(\zeta_k = \tau | \zeta'_k, \mathbf{z}_{\backslash k}) = \sigma(u_k - \log \tau), \tag{8}$$

where $\sigma(x) = (1 + e^{-x})^{-1}$ is the standard sigmoidal activation function and the $\log$ denotes the natural logarithm. The term u_k is the current membrane potential, which depends on the current values of the variables z_i for $i \neq k$. The term $\log \tau$ in (8) reflects the granularity of a chosen discrete time scale. If it is very fine (say one step equals one microsecond), then τ is large, and the firing probability at each specific discrete time step is therefore reduced. If the neuron in a state with $\zeta'_k \leq 1$ does not spike, ζ_k relaxes into the resting state $\zeta_k = 0$ corresponding to a non-refractory neuron.

If the neuron is in a refractory state, i.e., $\zeta'_k > 1$, its new variable ζ_k assumes deterministically the next lower value $\zeta_k = \zeta'_k - 1$, reflecting the inherent temporal process:

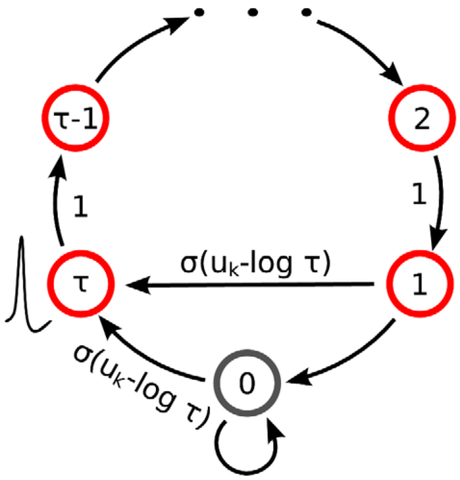

Figure 1. Neuron model with absolute refractory mechanism. The figure shows a schematic of the transition operator T^k for the internal state variable ζ_k of a spiking neuron ν_k with an absolute refractory period. The neuron can fire in the resting state $\zeta_k = 0$ and in the last refractory state $\zeta_k = 1$.
doi:10.1371/journal.pcbi.1002211.g001

$$T^k(\zeta_k = \zeta'_k - 1 | \zeta_{k'}, \mathbf{z}_{\setminus k}) = 1. \qquad (9)$$

After the transition of the auxiliary variable ζ_k, the binary variable z_k is deterministically set to a consistent state, i.e., $z_k = 1$ if $\zeta_k \geq 1$ and $z_k = 0$ if $\zeta_k = 0$.

It can be shown that each of these stochastic state transition operators T^k leaves the given distribution p invariant, i.e., satisfies equation (1). This implies that any composition or mixture of these operators T^k also leaves p invariant, see, e.g., [38]. In particular, the composition $T = T^1 \circ \ldots \circ T^K$ of these operators T^k leaves p invariant, which has a quite natural interpretation as firing dynamics of the spiking neural network $\mathcal{N}$: At each discrete time step the variables ζ_k, z_k are updated for all neurons ν_k, where the update of ζ_k, z_k takes preceding updates for ζ_i, z_i with $i > k$ into account. Alternatively, one could also choose at each discrete time step a different order for updates according to [38]. The assumption of a well-regulated updating policy will be overcome in the continuous-time limit, i.e., in case where the neural dynamics are described as a Markov jump process. In the methods section we prove the following central theorem:

Theorem 1. *$p(\boldsymbol{\zeta}, \mathbf{z})$ is the unique invariant distribution of operator T, i.e., T is aperiodic and irreducible and satisfies*

$$p(\boldsymbol{\zeta}, \mathbf{z}) = \sum_{\mathbf{z}', \mathbf{z}'} T(\boldsymbol{\zeta}, \mathbf{z} | \boldsymbol{\zeta}', \mathbf{z}') \cdot p(\boldsymbol{\zeta}', \mathbf{z}'). \qquad (10)$$

The proof of this Theorem is provided by Lemmata 1 – 3 in the Methods section. The statement that T (which is composed of the operators T^k) is irreducible and aperiodic ensures that p is the *unique* invariant distribution of the Markov chain defined by T, i.e., that irrespective of the initial network state the successive application of T explores the whole state space in a non-periodic manner.

This theorem guarantees that after a sufficient "burn-in" time (more precisely in the limit of an infinite "burn-in" time), the dynamics of the network, which are given by the transition operator T, produce samples from the distribution $p(\boldsymbol{\zeta}, \mathbf{z})$. As by construction $\sum_{\boldsymbol{\zeta}} p(\boldsymbol{\zeta}, \mathbf{z}) = p(\mathbf{z})$, the Markov chain provides samples from the given distribution $p(\mathbf{z})$. Furthermore, the network $\mathcal{N}$ can carry out probabilistic inference for this distribution. For example, $\mathcal{N}$ can be used to sample from the posterior distribution $p(z_1 \ldots, z_l | z_{l+1}, \ldots, z_K)$ over $z_1 \ldots, z_l$ given $z_{l+1}, \ldots, z_K$. One just needs to clamp those neurons $\nu_{l+1}, \ldots, \nu_K$ to the corresponding observed values. This could be implemented by injecting a strong positive (negative) current into the units with $z_j = 1$ ($z_j = 0$). Then, as soon as the stochastic dynamics of $\mathcal{N}$ has converged to its invariant distribution, the averaged firing rate of neuron ν_1 is proportional to the following desired marginal probability

$$p(z_1 = 1 | z_{l+1}, \ldots, z_K) = \sum_{z_2, \ldots, z_l} p(z_1 = 1, z_2, \ldots, z_l | z_{l+1}, \ldots, z_K).$$

In a biological neural system this result of probabilistic inference could for example be read out by an integrator neuron that counts spikes from this neuron ν_1 within a behaviorally relevant time window of a few hundred milliseconds, similarly as the experimentally reported integrator neurons in area LIP of monkey cortex [20,21]. Another readout neuron that receives spike input from ν_k could at the same time estimate $p(z_k = 1 | z_{l+1}, \ldots, z_K)$ for another RV z_k. But valuable information for probabilistic inference is not only provided by firing rates or spike counts, but also by spike correlations of the neurons $\nu_1, \ldots, \nu_l$ in $\mathcal{N}$. For example, the probability $p(z_1 = 1, z_2 = 1 | z_{l+1}, \ldots, z_K)$ can be estimated by a readout neuron that responds to superpositions of EPSPs caused by near-coincident firing of neurons ν_1 and ν_2 within a time interval of length τ. Thus, a large number of different probabilistic inferences can be carried out efficiently in parallel by readout neurons that receive spike input from different subsets of neurons in the network $\mathcal{N}$.

Variation of the discrete time model with a relative refractory mechanism. For the previously described simple neuron model, the refractory process was assumed to last for τ time steps, exactly as long as the postsynaptic potentials caused by each spike. In this section we relax this assumption by introducing a more complex and biologically more realistic neuron model, where the duration of the refractory process is decoupled from the duration τ of a postsynaptic potential. Thus, this model can for example also fire bursts of spikes with an interspike interval $< \tau$. The introduction of this more complex neuron model comes at the price that one can no longer prove that a network of such neurons samples from the desired distribution p. Nevertheless, if the sigmoidal activation function σ is replaced by a different activation function f, one can still prove that the sampling is "locally correct", as specified in equation (12) below. Furthermore, our computer simulations suggest that also globally the error introduced by the more complex neuron model is not functionally significant, i.e. that statistical dependencies between the RVs $\mathbf{z}$ are still faithfully captured.

The neuron model with a relative refractory period is defined in the following way. Consider some arbitrary refractory function $g : [0, \ldots, \tau] \rightarrow \mathbb{R}$ with $g(\tau) = 0$, $g(0) = 1$, and $g(l) \geq 0$ for $l = 1, \ldots, \tau - 1$. The idea is that $g(\zeta_k)$ models the readiness of the neuron to fire in its state ζ_k. This readiness has value 0 when the neuron has fired at the preceding time step (i.e., $\zeta_k = \tau$), and assumes the resting state 1 when ζ_k has dropped to 0. In between, the readiness may take on any non-negative value according to the function $g(\zeta_k)$. The function g does not need to be monotonic, allowing for example that it increases to high values in between, yielding a preferred interspike interval of a oscillatory neuron. The firing probability of neuron v_k in state ζ_k is given by $g(\zeta_k)\cdot f(u_k)$, where $f(u_k)$ is an appropriate function of the membrane potential as described below. Thus this function g is closely related to the function η (called afterpotential) in the spike response model [5] as well as to the self-excitation kernel in Generalized Linear Models [42]. In general, different neurons in the network may have different refractory profiles, which can be modeled by a different refractory function for each neuron v_k. However for the sake of notational simplicity we assume a single refractory function in the following.

In the presence of this refractory function g one needs to replace the sigmoidal activation function $\sigma(u_k - \log\tau)$ by a suitable function $f(u_k)$ that satisfies the condition

$$exp(u) = f(u)\frac{\sum_{\eta=1}^{\tau}\prod_{\zeta=\eta+1}^{\tau}(1-g(\zeta)\cdot f(u))}{\prod_{\zeta=1}^{\tau}(1-g(\zeta)\cdot f(u))} \qquad (11)$$

for all real numbers u. This equation can be derived (see Methods section Lemma 0) if one requires each neuron v_k to represent the correct distribution $p(z_k|\mathbf{z}_{\backslash k})$ over z_k conditioned the variables $\mathbf{z}_{\backslash k}$. One can show that, for any g as above, there always exists a continuous, monotonic function f which satisfies this equation (see Lemma 0 in Methods). Unfortunately (11) cannot be solved analytically for f in general. Hence, for simulations we approximate the function f for a given g by numerically solving (11) on a grid and interpolating between the grid points with a constant function. Examples for several functions g and the associated f are shown in Figure 2B and Figure 2C respectively. Furthermore, spike trains emitted by single neurons with these refractory functions g and the corresponding functions f are shown in Figure 2D for the case of piecewise constant membrane potentials. This figure indicates, that functions g that define a shorter refractory effect lead to higher firing rates and more irregular firing. It is worth noticing that the standard activation function $\sigma(u_k - \log\tau)$ is the solution of equation (11) for the absolute refractory function, i.e., for $g(0)=g(1)=1$ and $g(l)=0$ for $1<l\leq\tau$.

The transition operator T^k is defined for this model in a very similar way as before. However, for $1<\zeta'_k\leq\tau$, when the variable ζ'_k was deterministically reduced by 1 in the simpler model (yielding $\zeta_k=\zeta'_k-1$), this reduction occurs now only with probability $1-g(\zeta'_k)\cdot f(u_k)$. With probability $g(\zeta'_k)\cdot f(u_k)$ the operator T^k sets $\zeta_k=\tau$, modeling the firing of another spike of neuron v_k at this time point. The neural computability condition (4) remains unchanged, e.g., $u_k=b_k+\sum_{i=1}^{K}W_{ki}z_i$ for a Boltzmann distribution. A schema of the stochastic dynamics of this local state transition operator $T^k(\zeta_k|\zeta'_k,\mathbf{z}'_{\backslash k})$ is shown in Figure 2A.

This transition operator T^k has the following properties. In Lemma 0 in Methods it is proven that the unique invariant distribution of T^k, denoted as $q_k^*(\zeta_k,z_k|\boldsymbol{\zeta}_{\backslash k},\mathbf{z}_{\backslash k})$, gives rise to the correct marginal distribution over z_k, i.e.

$$\sum_{\zeta_k=0}^{\tau} q_k^*(\zeta_k,z_k|\boldsymbol{\zeta}_{\backslash k},\mathbf{z}_{\backslash k}) = p(z_k|\mathbf{z}_{\backslash k}).$$

This means that a neuron whose dynamics is described by T^k samples from the correct distribution $p(z_k|\mathbf{z}_{\backslash k})$ if it receives a static input from the other neurons in the network, i.e., as long as its membrane potential u_k is constant. Hence the “local” computa-

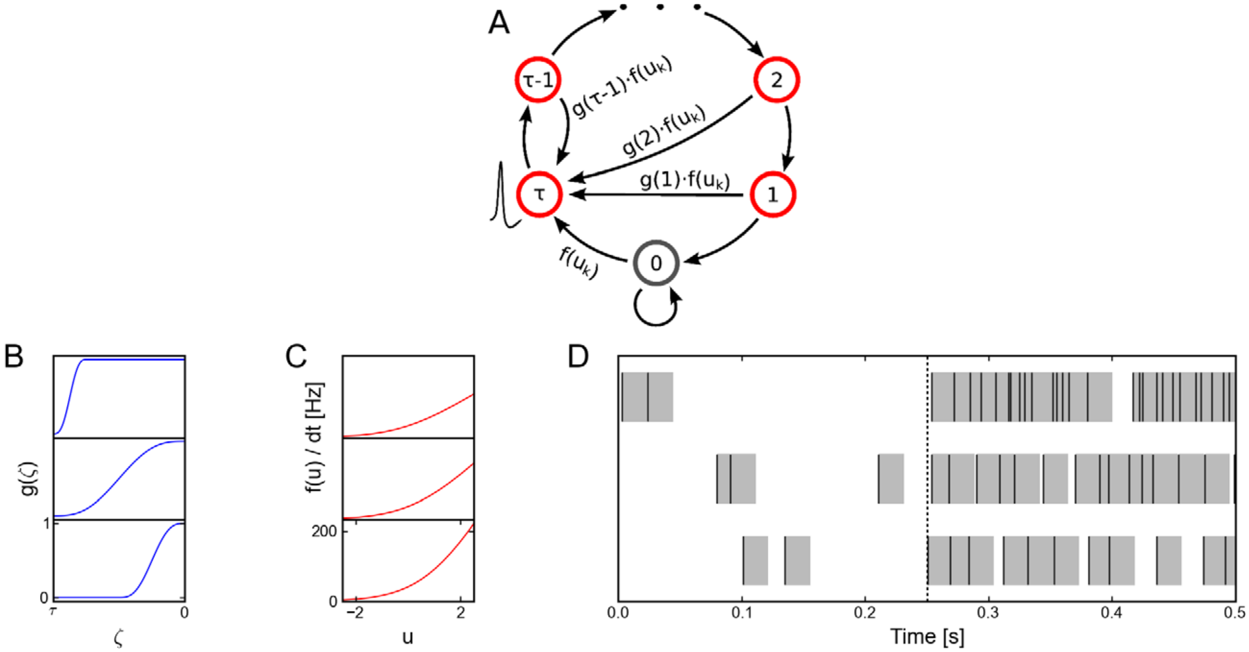

Figure 2. Neuron model with relative refractory mechanism. The figure shows the transition operator T^k, refractory functions g and activation functions f for the neuron model with relative refractory mechanism. (A) Transition probabilities of the internal variable ζ_k given by T^k. (B) Three examples of possible refractory functions g. They assume value 0 when the neuron cannot spike, and return to value 1 (full readiness to fire again) with different time courses. The value of g at intermediate time points regulates the current probability of firing of neuron v_k (see A). The x-axis is equivalent to the number of time steps since last spike (running from 0 to τ from left to right). (C) Associated activation functions f according to (11). (D) Spike trains produced by the resulting three different neuron models with (hypothetical) membrane potentials that jump at time $[0.25]s$ from a constant low value to a constant high value. Black horizontal bars indicate spikes, and the active states $z_k=1$ are indicated by gray shaded areas of duration $\tau\cdot dt=20ms$ after each spike. It can be seen from this example that different refractory mechanisms give rise to different spiking dynamics.
doi:10.1371/journal.pcbi.1002211.g002

tion performed by such neuron can be considered as correct. If however, several neurons in the network change their states in a short interval of time, the joint distribution over $\boldsymbol{z}$ is in general not the desired one, i.e., $\sum_{\zeta} q^{*}(\boldsymbol{\zeta},\boldsymbol{z}) \neq p(\boldsymbol{z})$, where $q^{*}(\boldsymbol{\zeta},\boldsymbol{z})$ denotes the invariant distribution of $T = T^{1} \circ \ldots \circ T^{K}$. In the Methods section, we present simulation results that indicate that the error of the approximation to the desired Boltzmann distributions introduced by neural sampling with relative refractory mechanism is rather minute. It is shown that the neural sampling approximation error is orders of magnitudes below the one introduced by a fully factorized distribution (which amounts to assuming correct marginal distributions $p(z_k)$ and independent neurons).

To illustrate the sampling process with the relative refractory mechanism, we examine a network of $K = 40$ neurons. We aim to sample from a Boltzmann distribution (5) with parameters W_{ij}, b_i being randomly drawn from normal distributions. For the neuron model, we use the relative refractory mechanism shown in the mid row of Figure 2B. A detailed description of the simulation and the parameters used is given in the Methods section. A spike pattern of the resulting sampling network is shown in Figure 3A. The network features a sparse, irregular spike response with average firing rate of 13.9 Hz. For one neuron ν_{26}, indicated with orange spikes, the internal dynamics are shown in Figure 3B. After each action potential the neuron's refractory function $g(\zeta_{26})$ drops to zero and reduces the probability of spiking again in a short time interval. The influence of the remaining network $\boldsymbol{z}_{\backslash 26}$ is transmitted to neuron ν_{26} via PSPs of duration $\tau \cdot dt = 20$ ms and sums up to the fluctuating membrane potential u_{26}. As reflected in the highly variable membrane potential even this small network exhibits rich interactions. To represent the correct distribution $p(z_{26}|\boldsymbol{z}_{\backslash 26})$ over z_{26} conditioned on $\boldsymbol{z}_{\backslash 26}$, the neuron ν_{26} continuously adapts its instantaneous firing rate. To quantify the precision with which the spiking network draws samples from the target distribution (5), Figure 3C shows the joint distribution of 5 neurons. For comparison we accompany the distribution of sampled network states with the result obtained from the standard Gibbs sampling algorithm (considered as the ground truth). Since the number of possible states $\boldsymbol{z}$ grows exponentially in the number of neurons, we restrict ourselves for visualization purposes to the distribution $p(z_{24}, \ldots, z_{28})$ of the gray shaded units and marginalize over the remaining network. The probabilities are estimated from 10^7 samples, i.e., from 10^7 successive states $\boldsymbol{z}$ of the Markov chain. Stochastic deviations of the estimated probabilities due to the finite number of samples are quite small (typical errors $\Delta p(\boldsymbol{z})/\sqrt{p(\boldsymbol{z})} \approx 10^{-3}$) and are comparable to systematic deviations due to the only locally correct computation of neurons with relative refractory mechanism. In the Methods section, we present further simulation results showing that the proposed networks consisting of neurons with relative refractory mechanism approximate the desired target distributions faithfully over a large range of distribution parameters.

In order to illustrate that the proposed sampling networks feature biologically quite realistic spiking dynamics, we present in the Methods section several neural firing statistics (e.g., the interspike interval histogram) of the network model. In general, the statistics computed from the model match experimentally observed statistics well. The proposed network models are based on the assumption of rectangular-shaped, renewal PSPs. More precisely, we define renewal (or non-additive) PSPs in the following way. Renewal PSPs evoked by a single synapse do not add up but are merely prolonged in their duration (according to equation (6)); renewal PSPs elicited at different synapses nevertheless add up in the normal way. In Methods we investigate the impact of replacing the theoretically ideal rectangular-shaped, renewal PSPs with biologically more realistic alpha-shaped, additive PSPs. Simulation results suggest that the network model with alpha-shaped PSPs does not capture the target distribution as accurately as with the theoretically ideal PSP shapes, statistical dependencies between the RVs $\boldsymbol{z}$ are however still approximated reasonably well.

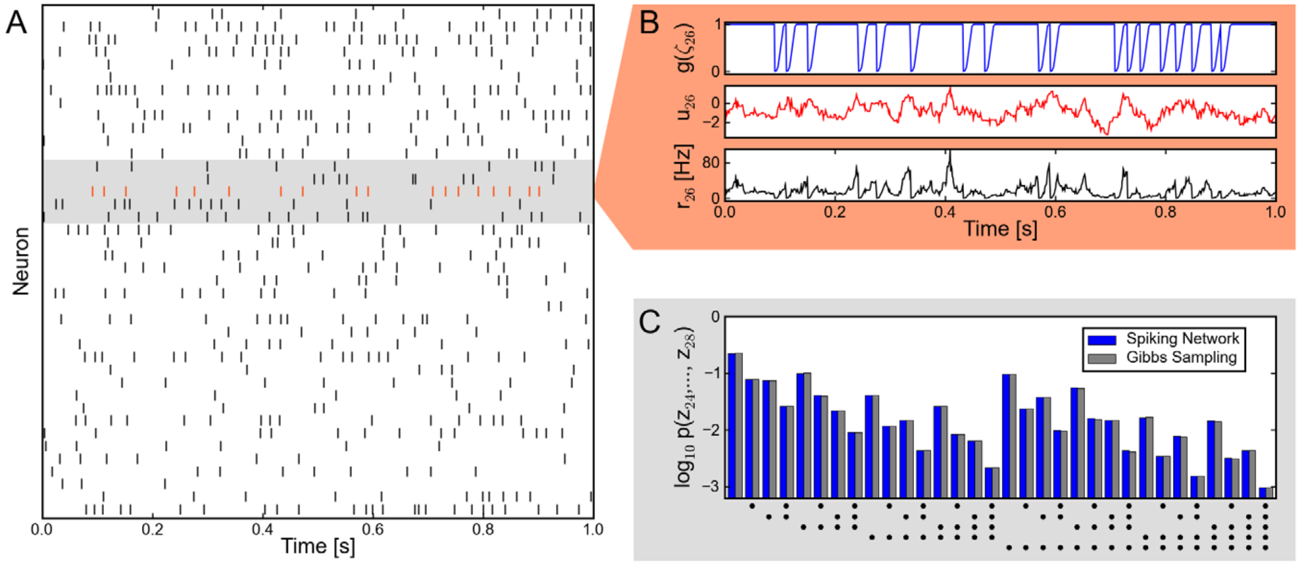

Figure 3. Sampling from a Boltzmann distribution by spiking neurons with relative refractory mechanism. (A) Spike raster of the network. (B) Traces of internal state variables of a neuron (# 26, indicated by orange spikes in A). The rich interaction of the network gives rise to rapidly changing membrane potentials and instantaneous firing rates. (C) Joint distribution of 5 neurons (gray shaded area in A) obtained by the spiking neural network and Gibbs sampling from the same distribution. Active states $z_i = 1$ are indicated by a black dot, using one row for each neuron ν_i, the columns list all $2^5 = 32$ possible states $(z_{24}, \ldots, z_{28})$ of these 5 neurons. The tight match between both distributions suggests that the spiking network represents the target probability distribution p with high accuracy.
doi:10.1371/journal.pcbi.1002211.g003

Neural sampling in continuous time

The neural sampling model proposed above was formulated in discrete time of step size dt, inspired by the discrete time nature of MCMC techniques in statistics and machine learning as well as to make simulations possible on digital computers. However, models in continuous time (e.g., ordinary differential equations) are arguably more natural and "realistic" descriptions of temporally varying biological processes. This gives rise to the question whether one can find a sensible limit of the discrete time model in the limit $dt \to 0$, yielding a sampling network model in continuous time. Another motivation for considering continuous time models for neural sampling is the fact that many mathematical models for recurrent networks are formulated in continuous time [5], and a comparison to these existing models would be facilitated. Here we propose a stochastically spiking neural network model in continuous time, whose states still represent correct samples from the desired probability distribution $p(\mathbf{z})$ at any time t. These types of models are usually referred to as Markov jump processes. It can be shown that discretizing this continuous time model yields the discrete time model defined earlier, which thus can be regarded as a version suitable for simulations on a digital computer.

We define the continuous time model in the following way. Let t_k^l, for $l = 0, 1, \ldots$, denote the firing times of neuron ν_k. The refractory process of this neuron, in analogy to Figure 1 and equation (8)-(9) for the case of discrete time, is described by the following differential equation for the auxiliary variable ζ_k, which may now assume any nonnegative real number $0 \leq \zeta_k \leq 1$

$$\frac{d}{dt}\zeta_k(t) = \begin{cases} -\frac{1}{\tau} & \text{for} \quad \zeta_k > 0 \\ \sum_l \delta(t - t_k^l) & \text{for} \quad \zeta_k = 0. \end{cases} \tag{12}$$

Here $\delta(t - t_k^l)$ denotes Dirac's Delta centered at the spike time t_k^l. This differential equation describes the following simple dynamics. The auxiliary variable $\zeta_k(t)$ decays linearly with time constant τ when the neuron is refractory, i.e., $\zeta_k(t) > 0$. Once $\zeta_k(t)$ arrives at its resting state 0 it remains there, corresponding to the neuron being ready to spike again (more precisely, in order to avoid point measures we set it to a random value in $[-2\epsilon, -\epsilon]$, see Methods). In the resting state, the neuron has the probability density $\frac{1}{\tau}\exp(u_k(t))$ to fire at every time t. If it fires at t_k^l, this results in setting $\zeta_k(t_k^l) = 1$, which is formalized in equation (12) by the sum of Dirac Delta's $\sum_l \delta(t - t_k^l)$. Here the current membrane potential $u_k(t)$ at time t is defined as in the discrete time case, e.g., by $u_k = b_k + \sum_{i=1}^K W_{ki} z_i(t)$ for the case of a Boltzmann distribution (5). The binary variable $z_k(t)$ is defined to be 1 if $\zeta_k(t) > 0$ and 0 if the neuron is in the resting state $\zeta_k(t) = 0$. Biologically, the term $W_{ki} z_i(t)$ can again be interpreted as the value at time t of a rectangular-shaped PSP (with a duration of τ) that neuron ν_i evokes in neuron ν_k. As the spikes are discrete events in continuous time, the probability of two or more neurons spiking at the same time is zero. This allows for updating all neurons in parallel using a differential equation.

In analogy to the discrete time case, the neural network in continuous time can be shown to sample from the desired distribution $p(\mathbf{z})$, i.e., $p(\mathbf{z})$ is an invariant distribution of the network dynamics defined above. However, to establish this fact, one has to rely on a different mathematical framework. The probability distribution $p_t(\boldsymbol{\zeta})$ of the auxiliary variables $\zeta_1(t), \ldots, \zeta_K(t)$ as a function of time t, which describes the evolution of the network, obeys a partial differential equation, the so-called Differential-Chapman-Kolmogorov equation (see [43]):

$$\partial_t p_t(\boldsymbol{\zeta}) = (T p_t)(\boldsymbol{\zeta}), \tag{13}$$

where the operator T, which captures the dynamics of the network, is implicitly defined by the differential equations (12) and the spiking probabilities. This operator T is the continuous time equivalent to the transition operator T in the discrete time case. The operator T consists here of two components. The *drift term* captures the deterministic decay process of $\zeta_k(t)$, stemming from the term $-1/\tau$ in equation (12). The *jump term* describes the non-continuous aspects of the path $\zeta_k(t)$ associated with "jumping" from $\zeta_k(t_k^l - dt) = 0$ to $\zeta_k(t_k^l) = 1$ at the time t_k^l when the neuron fires.

In the Methods section we prove that the resulting time invariant distribution, i.e., the distribution that solves $\partial_t p_t(\boldsymbol{\zeta}) = 0$, now denoted $p(\boldsymbol{\zeta})$ as it is not a function of time, gives rise to the desired marginal distribution $p(\mathbf{z})$ over $\mathbf{z}$:

$$\int d\boldsymbol{\zeta}\, \delta(\mathbf{z}, \boldsymbol{\zeta}^{>0}) p(\boldsymbol{\zeta}) = p(\mathbf{z}), \tag{14}$$

where $\delta(\mathbf{z}, \boldsymbol{\zeta}^{>0}) = (\delta(z_1, \zeta_1^{>0}), \ldots, \delta(z_K, \zeta_K^{>0}))$ and $\zeta_k^{>0} = 1$ if $\zeta_k > 0$ and $\zeta_k^{>0} = 0$ otherwise. $\delta(z_k, \zeta_k^{>0}) = 1$ denotes Kronecker's Delta with $\delta(z_k, \zeta_k^{>0}) = 1$ if $z_k = \zeta_k^{>0}$ and $\delta(z_k, \zeta_k^{>0}) = 0$ otherwise. Thus, the function $\delta(\mathbf{z}, \boldsymbol{\zeta}^{>0})$ simply reflects the definition that $z_k(t) = 1$ if $\zeta_k(t) > 0$ and 0 otherwise. For an explicit definition of T, a proof of the above statement, and some additional comments see the Methods section.

The neural samplers in discrete and continuous time are closely related. The model in discrete time provides an increasingly more precise description of the inherent spike dynamics when the duration dt of the discrete time step is reduced, causing an increase of τ (such that $\tau \cdot dt$ is constant) and therefore a reduced firing probability of each neuron at any discrete time step (see the term $\log \tau$ in equation (8)). In the limit of dt approaching 0, the probability that two or more neurons will fire at the same time approaches 0, and the discrete time sampler becomes equal to the continuous time system defined above, which updates all units in parallel.

It is also possible to formulate a continuous time version of the neural sampler based on neuron models with relative refractory mechanisms. In the Methods section the resulting continuous time neuron model with a relative refractory mechanism is defined. Theoretical results similar to the discrete time case can be derived for this sampler (see Lemmata 9 and 10 in Methods): It is shown that each neuron "locally" performs the correct computation under the assumption of static input from the remaining neurons. However one can no longer prove in general that the global network samples from the target distribution p.

Demonstration of probabilistic inference with recurrent networks of spiking neurons in an application to perceptual multistability

In the following we present a network model for perceptual multistability based on the neural sampling framework introduced above. This simulation study is aimed at showing that the proposed network can indeed sample from a desired distribution and also perform inference, i.e., sample from the correct corresponding posterior distribution. It is not meant to be a highly realistic or exhaustive model of perceptual multistability nor of biologically plausible learning mechanisms. Such models would naturally require considerably more modelling work.

Perceptual multistability evoked by ambiguous sensory input, such as a 2D drawing (e.g., Necker cube) that allows for different consistent 3D interpretations, has become a frequently studied perceptual phenomenon. The most important finding is that the perceptual system of humans and nonhuman primates does not produce a superposition of different possible percepts of an ambiguous stimulus, but rather switches between different self-consistent global percepts in a spontaneous manner. Binocular rivalry, where different images are presented to the left and right eye, has become a standard experimental paradigm for studying this effect [44–47]. A typical pair of stimuli are the two images shown in Figure 4A. Here the percepts of humans and nonhuman primates switch (seemingly stochastically) between the two presented orientations. [16–18] propose that several aspects of experimental data on perceptual multistability can be explained if one assumes that percepts correspond to samples from the conditional distribution over interpretations (e.g., different 3D shapes) given the visual input (e.g., the 2D drawing). Furthermore, the experimentally observed fact that percepts tend to be stable on the time scale of seconds suggests that perception can be interpreted as probabilistic inference that is carried out by MCMC sampling which produces successively correlated samples. In [18] it is shown that this MCMC interpretation is also able to qualitatively reproduce the experimentally observed distribution of dominance durations, i.e., the distribution of time intervals between perceptual switches. However, in lack of an adequate model for sampling by a recurrent network of spiking neurons, theses studies could describe this approach only on a rather abstract level, and pointed out the open problem to relate this algorithmic approach to neural processes. We have demonstrated in a computer simulation that the previously described model for neural sampling could in principle fill this gap, providing a modelling framework that is on the one hand consistent with the dynamics of networks of spiking neurons, and which can on the other hand also be clearly understood from the perspective of probabilistic inference through MCMC sampling.

In the following we model some essential aspects of an experimental setup for binocular rivalry with grating stimuli (see Figure 4A) in a recurrent network of spiking neurons with the previously described relative refractory mechanism. We assigned to each of the 217 neurons in the network $\mathcal{N}$ a tuning curve $V_k(\varphi)$, centered around its preferred orientation $\bar{\varphi}_k$ as shown in Figure 4B. The preferred orientations $\bar{\varphi}_k$ of the neurons were chosen to cover the entire interval $[0,\pi)$ of possible orientations and were randomly assigned to the neurons. The neurons were arranged on a hexagonal grid as depicted in Figure 4F. Any two neurons with distance ≤ 8 were synaptically connected (neighboring units had distance 1). We assume that these neurons represent neurons in the visual system that have roughly the same or neighboring receptive field, and that each neuron receives visual input from either the left or the right eye. The network connections were chosen such that neurons that have similar (very different) preferred orientations are connected with positive (negative) weights (for details see Methods section).

We examined the resulting distribution $p(\mathbf{z})$ over the 217 dimensional network states. To provide an intuitive visualization of these high dimensional network states $\mathbf{z}$, we resort to a 2-dimensional projection, the population vector of a state $\mathbf{z}$ (see Methods for details of the applied population vector decoding scheme). Only the endpoints of the population vectors are drawn (as colored points) in Figure 4D,E. The orientation of the population vector is assumed to correspond to the dominant orientation of the percept, and its distance from the origin encodes the strength of this percept. We also, somewhat informally, call the strength of a percept its coherence and a network state which represents a coherent percept a coherent network state. A coherent network state hence results in a population vector of large magnitude. Each direction of a population vector is color coded in Figure 4D,E, using the color code for directions shown on the right hand side of Figure 4F. In Figure 4D the distribution $p(\mathbf{z})$ of the network is illustrated by sampling of the network for $[20]s$, with samples $\mathbf{z}$ taken every millisecond. Each dot equals a sampled network state $\mathbf{z}$. In a biological interpretation the spike response of the freely evolving network reflects spontaneous activity, since no observations, i.e., no external input, was added to the system. Figure 4D shows that the spontaneous activity of this simple network of spiking neurons moves preferably through coherent network states for all possible orientations due to the chosen recurrent network connections (being positive for neurons with similar preferred orientation and negative otherwise). This can directly be seen from the rare occurrence of population vectors with small magnitude (vectors close to the "center") in Figure 4D.

To study percepts elicited by ambiguous stimuli, where inputs like in Figure 4A are shown simultaneously to the left and right eye during a binocular rivalry experiment, we provided ambiguous input to the network. Two cells with preferred orientation $\bar{\varphi}_k \approx 45^\circ$ and two cells with $\bar{\varphi}_k \approx 135^\circ$ were clamped to 1. Additionally four neurons with $\bar{\varphi}_k \approx 0^\circ$ resp. 90° were muted by clamping to 0. This ambiguous input is incompatible with a coherent percept, as it corresponds to two orthogonal orientations presented at the same time. The resulting distribution over the state of the 209 remaining neurons is shown for a time span of 20 s of simulated biological time (with samples taken every millisecond) in Figure 4E. One clearly sees that the network spends most of the time in network states that correspond to one of the two simultaneously presented input orientations (45^o and 135^o), and virtually no time on orientations in between. This implements a sampling process from a bimodal conditional distribution. The black line marks a 500 ms trace of network states $\mathbf{z}$ around a perceptual switch: The network remained in one mode of high probability – corresponding to one percept – for some period of time, and then quickly traversed the state space to another mode – corresponding to a different percept.

Three of the states $\mathbf{z}$ around this perceptual switch ($\mathbf{z}(t_1)$, $\mathbf{z}(t_2)$ and $\mathbf{z}(t_3)$ in Figure 4E) are explicitly shown in Figure 4F. Neurons v_k that fired during the preceding interval of 20 ms (marked in gray in Figure 4G) are drawn in the respective color of their preferred orientation. Inactive neurons are drawn in white, and clamped neurons are marked by a black dot (●).

Figure 4G shows the action potentials of the 209 non-clamped neurons during the same 500 ms trace around the perceptual switch. One sees that the sampling process is expressed in this neural network model by a sparse, asynchronous and irregular spike response. It is worth mentioning that the average firing rate when sampling from the posterior distribution is only slightly higher than the average firing rate of spontaneous activity (16.1 Hz and 15.4 Hz respectively), which is reminiscent of related experimental data [7]. Thus on the basis of the overall network activity it is indistinguishable whether the network carries out an inference task or freely samples from its prior distribution. It is furthermore notable, that a focus of the network activity on the two orientations that are given by the external input can be achieved in this model, in spite of the fact that only two of the 217 neurons were clamped for each of them. This numerical relationship is reminiscent of standard data on the weak input from LGN to V1 that is provided in the brain [48,49], and raises the question whether the proposed neural sampling model could provide a possible mechanism (under the modelling assumptions

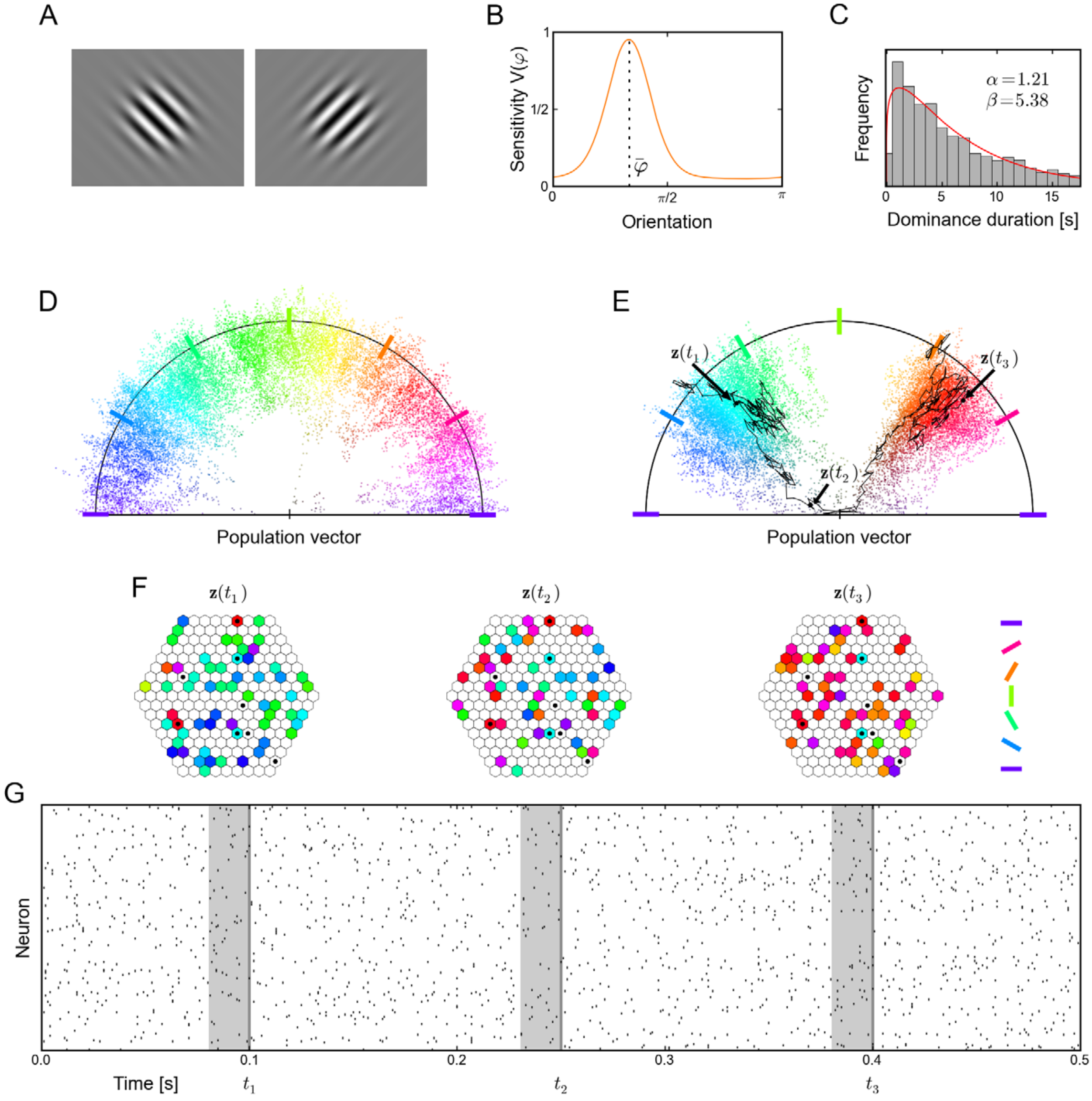

Figure 4. Modeling perceptual multistability as probabilistic inference with neural sampling. (A) Typical visual stimuli for the left and right eye in binocular rivalry experiments. (B) Tuning curve of a neuron with preferred orientation $\bar{\varphi}$. (C) Distribution of dominance durations in the trained network under ambiguous input. The red curve shows the Gamma distribution with maximum likelihood on the data. (D) 2-dimensional projection (via population vector) of the distribution $p(z)$ encoded in the spiking network showing that it strongly favors coherent global states of arbitrary orientation to incoherent ones (corresponding to population vectors of small magnitude). (E) 2-dimensional projection of the bimodal posterior distribution under an ambiguous input consisting of two different orientations reminiscent of the stimuli shown in A. The black trace shows the temporal evolution of the network state z for 500 ms around a perceptual switch. (F) Network states at 3 time points t_1, t_2, t_3 marked in E. Neurons that fired in the preceding 20 ms (see gray bar in G) are plotted in the color of their preferred orientation. Inactive neurons are shown in white. While states $z(t_1)$ and $z(t_3)$ represent rather coherent orientations, $z(t_2)$ shows an incoherent state corresponding to a perceptual switch. Clamped neurons (which the posterior is condition on) are marked by a black dot. (G) Spike raster of the unclamped neurons during a 500 ms epoch marked by the black trace in E. Gray bars indicate the 20 ms time intervals that define the network states shown in F. Altogether this figure shows that a theoretically rigorous probabilistic inference process can be carried out by a network of spiking neurons with a spike raster that is similar to generic recorded data.

doi:10.1371/journal.pcbi.1002211.g004

made above) for cortical processing of such numerically weak external inputs.

The distribution of the resulting dominance durations, i.e., the time between perceptual switches, for the previously described setup with ambiguous input is shown for a continuous run of 10^4 s in Figure 4C (a similar method as in [18] was used to measure dominance durations, see Methods). This distribution can be approximated quite well by a Gamma distribution, which also provides a good fit to experimental data (see the discussion in [18]). We expect that also other features of the more abstract

MCMC model for biological vision of [17,18], such as contextual biases and traveling waves, will emerge in larger and more detailed implementations of the MCMC approach through the proposed neural sampling method in networks of spiking neurons.

Discussion

We have presented a spiking neural network that samples from a given probability distribution via its inherent network dynamics. In particular the network is able to carry out probabilistic inference through sampling. The model, based on assumptions about the underlying probability distribution (formalized by the neural computability condition) as well as on certain assumptions regarding the underlying MCMC model, provides one possible neural implementation of the ''inference-by-sampling paradigm'' emerging in computational neuroscience.

During inference the observations (i.e., the variables which we wish to condition on) are modeled in this study by clamping the corresponding neurons by strong external input to the observed binary value. Units which receive no input or input with vanishing contrast (stimulus intensity) are treated as unobserved. Using this admittedly quite simplistic model of the input, we observed in simulations that our network model exhibits the following property: The onset of a sensory stimulus reduces the variability of the firing activity, which represents (after stimulus onset) a conditional distribution, rather than the prior distribution (see the difference between panels **D** and **E** of Figure 5. It is tempting to compare these results to the experimental finding of reduced firing rate variability after stimulus onset observed in several cortical areas [50]. We wish to point out however, that a consistent treatment of zero contrast stimuli requires more thorough modelling efforts (e.g., by explicitly adding a random variable for the stimulus intensity [35,51]), which is not the focus of the presented work.

Virtually all high-level computational tasks that a brain has to solve can be formalized as optimization problems, that take into account a (possibly large) number of soft or hard constraints. In typical applications of probabilistic inference in science and engineering (see e.g. [52,53]) such constraints are encoded in e.g., conditional probability tables or factors. In a biological setup they could possibly be encoded through the synaptic weights of a recurrent network of spiking neurons. The solution of such optimizations problems in a probabilistic framework via sampling, as implemented in our model, provides an alternative to deterministic solutions, as traditionally implemented in neural networks (see, e.g., [54] for the case of constraint satisfaction problems). Whereas an attractor neural network converges to *one* (possibly approximate) solution of the problem, a stochastic network may alternate between different approximate solutions and stay the longest at those approximate solutions that provide the best fit. This might be advantageous, as given more time a stochastic network can explore more of the state space and avoid shallow local minima. Responses to ambiguous sensory stimuli [44–47] might be interpreted as an optimization with soft constraints. The interpretation of human thinking as sampling process solving an inference task, recently proposed in cognitive science [28,55,56], further emphasizes that considering neural activity as an inferential process via sampling promises to be a fruitful approach.

Our approach builds on, and extends, previous work where recurrent networks of non-spiking stochastic neurons (commonly considered in artificial neural networks) were shown to be able to carry out probabilistic inference through Gibbs sampling [36]. In [57] a first extension of this approach to a network of recurrently connected spiking neurons had been presented. The dynamics of the recurrently connected spiking neurons are described as stepwise sampling from the posterior of a temporal Restricted Boltzmann Machine (tRBM) by introducing a clever interpretation of the temporal spike code as time varying parameters of a multivariate Gaussian distribution. Drawing one sample from the posterior of a RBM is, by construction, a trivial one-step task. In contrast to our model, the model of [57] does not produce multiple samples from a fixed posterior distribution, given the fixed input, but produces exactly one sample consisting of the temporal sequence of the hidden nodes, given a temporal input sequence. Similar temporal models, sometimes called Bayesian filtering, also underlie the important contributions of [58] and [32]. In [32] every single neuron is described as hidden Markov Model (HMM) with two states. Instead of drawing samples from the instantaneous posterior distribution using stochastic spikes, [32] presents a deterministic spike generation with the intention to convey the analog probability value rather than discrete samples. The approach presented here can be interpreted as a biologically more realistic version of Gibbs sampling for a specific class of

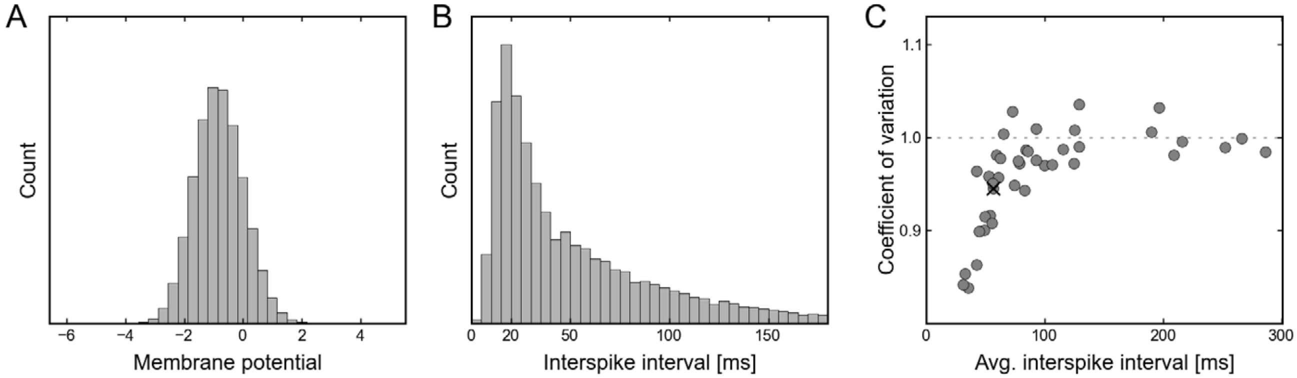

Figure 5. Firing statistics of neural sampling networks. (A) Shown is the membrane potential histogram of a typical neuron during sampling. The data is that of neuron v_{26} from the simulation shown in Figure 3 (the membrane potential and spike trace of v_{26} are highlighted in Figure 3). (B) The plot shows the ISI distribution of a typical neuron (again v_{26} from Figure 3) during sampling. The distribution is roughly gamma-shaped, reminiscent of experimentally observed ISI distributions. (C) A scatter plot of the coefficient of variation (CV) versus the average interspike interval (ISI) of each neuron taken from the simulation shown in Figure 3. The value of neuron v_{26} from Figure 3 is marked by a cross. The simulated data is in accordance with experimentally observed data.
doi:10.1371/journal.pcbi.1002211.g005

probability distributions by taking into account a spike-based communication, finite duration PSPs and refractory mechanisms. Other implementations based on different distributions (e.g., directed graphical models) and different sampling methods (e.g., reversible MCMC methods) are of course conceivable and worth exploring.

In a computer experiment (see Figure 4, we used our proposed network to model aspects of biological vision as probabilistic inference along the lines of argumentation put forward in [16–18]. Our model was chosen to be quite simplistic, just to demonstrate that a number of experimental data on the dynamics of spontaneous activity [51,59,60] and binocular rivalry [44–47] can in principle be captured by this approach. The main point of the modelling study is to show that rather realistic neural dynamics can support computational functions rigorously formalized as inference via sampling.

We have also presented a model of spiking dynamics in continuous time that performs sampling from a given probability distribution. Although computer simulations of biological networks of neurons often actually use discrete time, it is desirable to also have a sound approach for understanding and describing the network sampling dynamics in continuous time, as the latter is arguable a natural framework for describing temporal processes in biology. Furthermore comparison to many existing continuous time neuron and network models of neurons is facilitated.

We have made various simplifying assumption regarding neural processes, e.g., simple symbolic postsynaptic potentials in the form of step-functions (reminiscent of plateau potentials caused by dendritic NMDA spikes [61]). More accurate models for neurons have to integrate a multitude of time constants that represent different temporal processes on the physical, molecular, and genetic level. Hence the open problem arises, to which extent this multitude of time constants and other complex dynamics can be integrated into theoretical models of neural sampling. We have gone one first step in this direction by showing that in computer simulations the two temporal processes that we have considered (refractory processes and postsynaptic potentials) can approximately be decoupled. Furthermore, we have presented simulation results suggesting that more realistic alpha-shaped, additive EPSPs are compatible with the functionality of the proposed network model.

Finally, we want to point out that the prospect of using networks of spiking neurons for probabilistic inference via sampling suggests new applications for energy-efficient spike-based and massively parallel electronic hardware that is currently under development [62,63].

Methods

We first provide details and proofs for the neural sampling models, followed by details for the computer simulations. Then we investigate typical firing statistics of individual neurons during neural sampling and examine the approximation quality of neural sampling with different neuron and synapse models.

Mathematical details

Notation. To keep the derivations in a compact form, we introduce the following notations. We define the function $\zeta_k^{>0}$ of ζ_k to be 1 if $\zeta_k>0$ and 0 otherwise. Analogously we define $\boldsymbol{\zeta}_{\backslash k}^{>0}=(\zeta_1^{>0},\ldots,\zeta_{k-1}^{>0},\zeta_{k+1}^{>0},\ldots,\zeta_K^{>0})$. Let $\delta(\cdot,\cdot)$ denote Kronecker's Delta, i.e., $\delta(x,y)=1$ if $x=y$ and 0 whereas $\delta(\cdot)$ denotes Dirac's Delta, i.e., $\int f(x)\delta(x)dx=f(0)$. Furthermore $\chi_I(x)$ is the indicator function of the set I, i.e., $\chi_I(x)=1$ if $x\in I$ and $\chi_I(x)=0$ if $x\notin I$.

Details to neural sampling with absolute refractory period in discrete time. The following Lemmata 1 – 3 provide a proof of Theorem 1. For completeness we begin this paragraph with a recapitulation of the definitions stated in Results. We then identify some central properties of the joint probability distribution $p(\boldsymbol{\zeta},\boldsymbol{z})$ and proof that the proposed network samples from the desired invariant distribution.

For a given distribution $p(\boldsymbol{z})$ over the binary variables $\boldsymbol{z}\in\{0,1\}^K$ with $\forall \boldsymbol{z}\in\{0,1\}^K\ p(\boldsymbol{z})\neq 0$, the joint distribution over $(\boldsymbol{\zeta},\boldsymbol{z})$ with $\boldsymbol{\zeta}\in\{0,1,\ldots,\tau\}^K$ is defined in the following way (see equation 7):

$$p(\zeta_k|z_k):=\begin{cases}\tau^{-1} & \text{for} \quad z_k=1\ \wedge\ \zeta_k>0\\ 1 & \text{for} \quad z_k=0\ \wedge\ \zeta_k=0\\ 0 & \text{otherwise}\end{cases}$$

$$p(\boldsymbol{\zeta}|\boldsymbol{z}):=\prod_{k=1}^{K}p(\zeta_k|z_k)$$

$$p(\boldsymbol{\zeta},\boldsymbol{z}):=p(\boldsymbol{\zeta}|\boldsymbol{z})p(\boldsymbol{z}).$$

The assumption $p(\boldsymbol{z})\neq 0$ for all $\boldsymbol{z}$ is required to show the irreducibility of the Markov chain, a prerequisite to ensure the uniqueness of the invariant distribution of the MCMC dynamics. Furthermore, for the given distribution $p(\boldsymbol{z})$ we define the functions $u_k:\{0,1\}^{K-1}\rightarrow\mathbb{R}$ for $k\in\{1,\ldots,K\}$ which map $\boldsymbol{z}_{\backslash k}\mapsto u_k(\boldsymbol{z}_{\backslash k})$:

$$u_k(\boldsymbol{z}_{\backslash k}):=\text{logit}(p(z_k=1|\boldsymbol{z}_{\backslash k}))=\log\frac{p(z_k=1|\boldsymbol{z}_{\backslash k})}{p(z_k=0|\boldsymbol{z}_{\backslash k})}.$$

Instead of $u_k(\boldsymbol{z}_{\backslash k})$ we simply write u_k in the following.

Lemma 1. The distribution $p(\boldsymbol{\zeta},\boldsymbol{z})$ has conditional distributions of the following form:

$$p(\zeta_k|\boldsymbol{\zeta}_{\backslash k},\boldsymbol{z}_{\backslash k})=p(\zeta_k|\boldsymbol{z}_{\backslash k})=\begin{cases}\dfrac{\sigma(u_k)}{\tau} & \text{for} \quad \zeta_k>0\\ 1-\sigma(u_k) & \text{otherwise}\end{cases}$$

$$p(z_k|\boldsymbol{\zeta},\boldsymbol{z}_{\backslash k})=p(z_k|\zeta_k)=\begin{cases}1 & \text{for} \quad \zeta_k>0\ \wedge z_k=1\\ 1 & \text{for} \quad \zeta_k=0\ \wedge z_k=0\\ 0 & \text{otherwise.}\end{cases}$$

These results can also be written more compactly in the following form: $p(\zeta_k|\boldsymbol{z}_{\backslash k})=\sigma(u_k)\chi_{\{1,\ldots,\tau\}}(\zeta_k)\frac{1}{\tau}+(1-\sigma(u_k))\delta(\zeta_k,0)$ and $p(z_k|\zeta_k)=\delta(z_k,\zeta_k^{>0})$.

Proof. Here we use the fact that the logistic function σ is the inverse of the logit function, i.e., $p(z_k=1|\boldsymbol{z}_{\backslash k})=\sigma(u_k)$.

$$p(\zeta_k|\boldsymbol{\zeta}_{\backslash k},\boldsymbol{z}_{\backslash k})=\sum_{z_k=0}^{1}\frac{p(\boldsymbol{\zeta},\boldsymbol{z})}{p(\boldsymbol{\zeta}_{\backslash k},\boldsymbol{z}_{\backslash k})}=\sum_{z_k=0}^{1}\frac{p(\boldsymbol{\zeta},\boldsymbol{z})}{p(\boldsymbol{\zeta}_{\backslash k}|\boldsymbol{z}_{\backslash k})p(\boldsymbol{z}_{\backslash k})}$$

$$=\sum_{z_k=0}^{1}\frac{(\Pi_{l\neq k}\,p(\zeta_l|z_l))p(\zeta_k|z_k)p(\boldsymbol{z})}{(\Pi_{l\neq k}\,p(\zeta_l|z_l))p(\boldsymbol{z}_{\backslash k})}$$

$$=\sum_{z_k=0}^{1}p(\zeta_k|z_k)p(z_k|\boldsymbol{z}_{\backslash k})=\sigma(u_k)\chi_{\{1,\ldots,\tau\}}(\zeta_k)\frac{1}{\tau}+(1-\sigma(u_k))\delta(\zeta_k,0).$$

This also shows that ζ_k is independent from $\boldsymbol{\zeta}_{\setminus k}$ given $\boldsymbol{z}_{\setminus k}$, i.e., $p(\zeta_k|\boldsymbol{\zeta}_{\setminus k},\boldsymbol{z}_{\setminus k})=p(\zeta_k|\boldsymbol{z}_{\setminus k})$. Now we show the second relation using Bayes' rule:

$$
\begin{aligned}
p(z_k|\boldsymbol{\zeta},\boldsymbol{z}_{\setminus k}) &= \frac{p(\zeta_k|\boldsymbol{\zeta}_{\setminus k},\boldsymbol{z})}{p(\zeta_k|\boldsymbol{\zeta}_{\setminus k},\boldsymbol{z}_{\setminus k})}p(z_k|\boldsymbol{\zeta}_{\setminus k},\boldsymbol{z}_{\setminus k})\\
&= \frac{z_k\chi_{\{1,\ldots,\tau\}}(\zeta_k)\frac{1}{\tau}+(1-z_k)\delta(\zeta_k,0)}{\sigma(u_k)\chi_{\{1,\ldots,\tau\}}(\zeta_k)\frac{1}{\tau}+(1-\sigma(u_k))\delta(\zeta_k,0)}p(z_k|\boldsymbol{z}_{\setminus k})\\
&= \begin{cases} z_k & \text{for} \quad \zeta_k>0\\ 1-z_k & \text{for} \quad \zeta_k=0\end{cases}\\
&= \delta(z_k,\zeta_k^{>0}).
\end{aligned}
$$

In order to facilitate the verification of the next two Lemmata, we first restate the definition of the operators T^k in a more concise way:

$$
\begin{aligned}
&T := T^1 \circ \ldots \circ T^K\\
&T^k(\boldsymbol{\zeta},\boldsymbol{z}|\boldsymbol{\zeta}',\boldsymbol{z}') := T^k(\zeta_k,z_k|\boldsymbol{\zeta}',\boldsymbol{z}')\delta(\boldsymbol{\zeta}_{\setminus k},\boldsymbol{\zeta}'_{\setminus k})\delta(\boldsymbol{z}_{\setminus k},\boldsymbol{z}'_{\setminus k})\\
&T^k(\zeta_k,z_k|\boldsymbol{\zeta}',\boldsymbol{z}') := \delta(z_k,\zeta_k^{>0})\cdot T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})\\
&T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k}) := \begin{cases} \sigma(u'_k-\log\tau) & \text{for} \quad \zeta_k=\tau \wedge \zeta'_k=0,1\\ 1-\sigma(u'_k-\log\tau) & \text{for} \quad \zeta_k=0 \wedge \zeta'_k=0,1\\ 1 & \text{for} \quad \zeta_k=\zeta'_k-1 \wedge \zeta'_k>1\\ 0 & \text{otherwise}\end{cases},
\end{aligned}
$$

where $u'_k := u_k(\boldsymbol{z}'_{\setminus k})=\mathrm{logit}(p(z_k=1|\boldsymbol{z}'_{\setminus k}))$.

Lemma 2. For all $k=1,\ldots,K$ the operator $T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})$ leaves the conditional distribution $p(\zeta_k|\boldsymbol{z}'_{\setminus k})$ invariant.

Proof. For sake of simplicity, denote $T^k(\zeta_k=i|\zeta'_k=j,\boldsymbol{z}'_{\setminus k})=T^k_{ij}$ for $i,j\in\{0,1,\ldots,\tau\}$ and $p(\zeta_k=i|\boldsymbol{z}'_{\setminus k})=p_i$. We have to show $p_i \overset{!}{=} \sum_{j=0}^{\tau} T^k_{ij}p_j$ for $i\in\{0,1,\ldots,\tau\}$.

First we show $p_\tau=\sum_{j=0}^{\tau} T^k_{\tau j}p_j$ using $p_0=1-\sigma(u_k)$ and $p_1=p_2=\ldots=p_\tau=\sigma(u_k)\tau^{-1}$ (which results from Lemma 1):

$$
\begin{aligned}
&\sum_{j=0}^{\tau} T^k_{\tau j}p_j = T^k_{\tau 0}p_0+T^k_{\tau 1}p_1\\
&=\sigma(u_k-\log\tau)(1-\sigma(u_k))+\sigma(u_k-\log\tau)\sigma(u_k)\tau^{-1}\\
&=\sigma(u_k-\log\tau)\sigma(u_k)\tau^{-1}(\tau\exp(-u_k)+1)\\
&=\sigma(u_k-\log\tau)\sigma(u_k)\tau^{-1}(\sigma(u_k-\log\tau))^{-1}=\sigma(u_k)\tau^{-1}\overset{!}{=}p_\tau.
\end{aligned}
$$

Here we used the definition of the logistic function $\sigma(x)=(1+\exp(-x))^{-1}$ and $\sigma(x)(1-\sigma(x))^{-1}=\exp(x)$.

Now we show $p_0=\sum_{j=0}^{\tau} T^k_{0j}p_j$:

$$
\begin{aligned}
&\sum_{j=0}^{\tau} T^k_{0j}p_j = T^k_{00}p_0+T^k_{01}p_1\\
&=(1-\sigma(u_k-\log\tau))(1-\sigma(u_k))+(1-\sigma(u_k-\log\tau))\sigma(u_k)\tau^{-1}\\
&=(1-\sigma(u_k-\log\tau))(1-\sigma(u_k))\left(1+\exp(u_k)\tau^{-1}\right)\\
&=\sigma(-u_k+\log\tau)(1-\sigma(u_k))(\sigma(-u_k+\log\tau))^{-1}=1-\sigma(u_k)\overset{!}{=}p_0.
\end{aligned}
$$

Here we used $1-\sigma(x)=\sigma(-x)$.

It is trivial to show $p_i=\sum_{j=0}^{\tau} T^k_{ij}p_j$ for $i=1,\ldots,\tau-1$ as $\sum_{j=0}^{\tau} T^k_{ij}p_j=T^k_{i,i+1}p_{i+1}=p_{i+1}=p_i$. Here we used the facts that $T^k_{i,i+1}=1$ and $p_i=p_{i+1}$ for $i=1,\ldots,\tau-1$ by definition.

Lemma 3. For all $k=1,\ldots,K$ the operator $T^k(\boldsymbol{z},\boldsymbol{\zeta}|\boldsymbol{\zeta}',\boldsymbol{z}')$ leaves the distribution $p(\boldsymbol{\zeta},\boldsymbol{z})$ invariant.

Proof. We start from Lemma 2, which states that $T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})$ leaves the conditional distribution $p(\zeta_k|\boldsymbol{z}'_{\setminus k})$ invariant:

$$
\begin{aligned}
&\sum\nolimits_{\zeta'_k} T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})p(\zeta'_k|\boldsymbol{z}'_{\setminus k})=p(\zeta_k|\boldsymbol{z}'_{\setminus k})\\
&\Leftrightarrow \sum\nolimits_{\zeta'_k,\boldsymbol{z}_{\setminus k}'} \delta(\boldsymbol{z}_{\setminus k},\boldsymbol{z}'_{\setminus k})T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})p(\zeta'_k|\boldsymbol{z}'_{\setminus k})\\
&\quad = \sum_{\boldsymbol{z}'_{\setminus k}} \delta(\boldsymbol{z}_{\setminus k},\boldsymbol{z}'_{\setminus k})p(\zeta_k|\boldsymbol{z}'_{\setminus k})=p(\zeta_k|\boldsymbol{z}_{\setminus k})\\
&\Leftrightarrow \sum\nolimits_{\zeta'_k,\boldsymbol{z}_{\setminus k}'} \delta(z_k,\zeta_k^{>0})\delta(\boldsymbol{z}_{\setminus k},\boldsymbol{z}'_{\setminus k})T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})p(\zeta'_k|\boldsymbol{z}'_{\setminus k})\\
&\quad = \delta(z_k,\zeta_k^{>0})p(\zeta_k|\boldsymbol{z}_{\setminus k})=p(z_k|\zeta_k)p(\zeta_k|\boldsymbol{z}_{\setminus k})\\
&\Leftrightarrow \sum\nolimits_{\zeta'_k,\boldsymbol{z}'} \delta(z_k,\zeta_k^{>0})\delta(\boldsymbol{z}_{\setminus k},\boldsymbol{z}'_{\setminus k})T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})p(z'_k,\zeta'_k|\boldsymbol{z}'_{\setminus k})=p(z_k,\zeta_k|\boldsymbol{z}_{\setminus k})\\
&\Leftrightarrow \sum\nolimits_{\zeta'_k,\boldsymbol{z}'} \delta(z_k,\zeta_k^{>0})\delta(\boldsymbol{z}_{\setminus k},\boldsymbol{z}'_{\setminus k})T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})p(z'_k,\zeta'_k|\boldsymbol{z}'_{\setminus k})p(\boldsymbol{z}_{\setminus k}|\boldsymbol{z}'_{\setminus k})p(\boldsymbol{z}'_{\setminus k})\\
&\quad = p(z_k,\zeta_k|\boldsymbol{z}_{\setminus k})p(\boldsymbol{\zeta}_{\setminus k}|\boldsymbol{z}_{\setminus k})p(\boldsymbol{z}_{\setminus k})\\
&\Leftrightarrow \sum\nolimits_{\zeta'_k,\boldsymbol{z}'} \delta(z_k,\zeta_k^{>0})\delta(\boldsymbol{z}_{\setminus k},\boldsymbol{z}'_{\setminus k})T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\setminus k})p(\zeta'_k,\boldsymbol{z}_{\setminus k},\boldsymbol{z}')=p(\boldsymbol{\zeta},\boldsymbol{z})\\
&\Leftrightarrow \sum\nolimits_{\boldsymbol{\zeta}',\boldsymbol{z}'} T^k(z_k,\zeta_k|\boldsymbol{\zeta}',\boldsymbol{z}')\delta(\boldsymbol{\zeta}_{\setminus k},\boldsymbol{\zeta}'_{\setminus k})\delta(\boldsymbol{z}_{\setminus k},\boldsymbol{z}'_{\setminus k})p(\boldsymbol{z}',\boldsymbol{\zeta}')=p(\boldsymbol{\zeta},\boldsymbol{z})\\
&\Leftrightarrow \sum\nolimits_{\boldsymbol{\zeta}',\boldsymbol{z}'} T^k(\boldsymbol{z},\boldsymbol{\zeta}|\boldsymbol{\zeta}',\boldsymbol{z}')p(\boldsymbol{z}',\boldsymbol{\zeta}')=p(\boldsymbol{\zeta},\boldsymbol{z}).
\end{aligned}
$$

Here we used the relations $\delta(z_k,\zeta_k^{>0})=p(z_k|\zeta_k)$ and $p(\zeta_k,z_k|\boldsymbol{z}_{\setminus k})=p(z_k|\zeta_k)p(\zeta_k|\boldsymbol{z}_{\setminus k})$ as well as $p(\zeta_k|\boldsymbol{z}_{\setminus k})=p(\zeta_k|\boldsymbol{\zeta}_{\setminus k},\boldsymbol{z}_{\setminus k})$ which directly follow from the definitions of $T^k(\boldsymbol{\zeta},\boldsymbol{z},|\boldsymbol{\zeta}',\boldsymbol{z}')$ and $p(\boldsymbol{\zeta},\boldsymbol{z})$.

Finally, we can verify that the composed operator $T=T^1\circ\ldots\circ T^K$ samples from the given distribution p.

Theorem 1. $p(\boldsymbol{\zeta},\boldsymbol{z})$ is the unique invariant distribution of operator T.

Proof. As all T^k leave $p(\boldsymbol{\zeta},\boldsymbol{z})$ invariant, so does the concatenation $T=T^1\circ\ldots\circ T^K$. To ensure that $p(\boldsymbol{\zeta},\boldsymbol{z})$ is the *unique* invariant distribution, we have to show that T is irreducible and aperiodic. T is aperiodic as the transition probabilities $T^k_{00}=1-\sigma(u_k-\log\tau)>0$ and $T^k_{00}<1$ (this follows from the assumption $\forall\boldsymbol{z}\ p(\boldsymbol{z})\neq 0$ made above).

The operator T is also irreducible for the following reason. First we see that from any state $(\boldsymbol{\zeta}',\boldsymbol{z}')$ in at most τ steps we can get to the zero-state $(\boldsymbol{\zeta},\boldsymbol{z})=0^{2K}$ (and stay there) with non-zero probability, as $T^k_{i,i+1}=1$ for $i=1,\ldots,\tau-1$ and $T^k_{01}=1-\sigma(u_k-\log\tau)>0$. Furthermore, it can be seen that any state $(\hat{\boldsymbol{\zeta}},\hat{\boldsymbol{z}})$ can be reached from the zero-state $(\boldsymbol{\zeta},\boldsymbol{z})=0^{2K}$ in at most τ steps since $T^k_{N0}=\sigma(u_k-\log\tau)>0$ for any value of u_k. Hence every final state $(\hat{\boldsymbol{\zeta}},\hat{\boldsymbol{z}})$ can be reached from every starting state $(\boldsymbol{\zeta}',\boldsymbol{z}')$ in at most 2τ steps with non-vanishing probability.

Details to neural sampling with a relative refractory period in discrete time. We augment the neuron model with a relative refractory period described by a function $g(\zeta_k)$. We first ensure existence of the corresponding function $f(u_k)$. Based on these functions we then introduce the transition operator T of the Markov chain. This operator is shown to entail correct "local" computations.

Lemma 4. Let $(g_1,\ldots,g_\tau)\in(\mathbb{R}_0^+)^\tau$ *be a tuple of non-negative real numbers, with* $g_\tau=0$ *and at least one element* $g_i\geq 1$. *This defines the refractory function via* $g(\zeta_k):=g_{\zeta_k}$. *There exists a unique* $\mathcal{C}^\infty$ *function* $f:\mathbb{R}\rightarrow(0,1)$ *with the following property* $\forall u\in\mathbb{R}$*:*

$$f(u)\frac{\sum_{i=1}^{\tau}\prod_{j=i+1}^{\tau}(1-g_jf(u))}{\prod_{j=1}^{\tau}(1-g_jf(u))}=\exp(u). \qquad (15)$$

Furthermore, the function f *has the property:*

$$\forall i\in\{1,\ldots,\tau\}\ \forall u\in\mathbb{R}:\quad 0\leq g_if(u)<1$$
$$\exists i\in\{1,\ldots,\tau\}\ \forall u\in\mathbb{R}:\quad 0<g_if(u)<1.$$

Proof. Let $g_{\max}:=max_{j\in\{1,\ldots\tau\}}g_j$; we know that $g_{\max}\geq 1$. We define the function $F:(0,1/g_{\max})\rightarrow\mathbb{R}^+$:

$$F(x):=x\sum_{i=1}^{\tau}\left(\frac{1}{\prod_{j=1}^{i}(1-g_jx)}\right)$$

We can see that F is a positive $\mathcal{C}^\infty$ function on $(0,1/g_{\max})$. Furthermore, $F(x)/x$ is defined as a sum of functions of the form $\dfrac{1}{\prod_{j=1}^{i}(1-g_jx)}$. Each factor $1/(1-g_jx)$ is positive and strictly monotonous. Therefore, F is strictly monotonous on $(0,1/g_{\max})$ with the limits:

$$\lim_{x\rightarrow 0}F(x)=0$$
$$\lim_{x\rightarrow 1/g_{\max}}F(x)=\infty.$$

Hence the equation $F(x)=\exp(u)$ has a unique solution for x called $f(u)\in(0,1/g_{\max})$ for all $u\in\mathbb{R}$. From applying the implicit function theorem to $F(x,u):=F(x)-\exp(u)$ it follows that f is $\mathcal{C}^\infty$.

From here on, with the letter f we will denote the function characterized by the above Lemma for the given tuple g (which denotes the chosen refractory function).

Definition 1. Define $g_0=1$. *The transition operator* T^k *is defined in the following way for all* $k=1,\ldots,K$*:*

$$T^k(\zeta_k,z_k|\boldsymbol{\zeta}',\boldsymbol{z}'):=\delta(z_k,\zeta_k^{>0})T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\backslash k})$$

$$T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\backslash k}):=\begin{cases}g_{\zeta_{k'}}f(u_k) & \text{for}\quad \zeta_k=\tau\\ 1-g_{\zeta_{k'}}f(u_k) & \text{for}\quad \zeta_k=\zeta_{k'}-1\ \wedge\ \zeta_{k'}>0\\ 1-f(u_k) & \text{for}\quad \zeta_k=0\ \wedge\ \zeta_{k'}=0\\ 0 & \text{otherwise}\end{cases},$$

with $u_k=u_k(\boldsymbol{z}'_{\backslash k})$.

Lemma 5. For all $k=1,\ldots,K$ *the unique invariant distribution* $q^*(z_k,\zeta_k|\boldsymbol{\zeta}'_{\backslash k},\boldsymbol{z}'_{\backslash k})$ *of the operator* $T^k(z_k,\zeta_k|\boldsymbol{\zeta}',\boldsymbol{z}')$ *fulfills* $\sum_{\zeta_k}q^*(z_k,\zeta_k|\boldsymbol{\zeta}'_{\backslash k},\boldsymbol{z}'_{\backslash k})=p(z_k|\boldsymbol{z}'_{\backslash k})$. *This means, for a constant configuration* $\boldsymbol{z}'_{\backslash k}$, *the operator* T^k *produces samples* z_k^* *from the correct conditional distribution* $p(z_k|\boldsymbol{z}'_{\backslash k})$.

Proof. We define:

$$q^*(z_k,\zeta_k|\zeta'_{\backslash k},\mathrm{z}'_{\backslash k}):=\delta(z_k,\zeta_k^{>0})q(\zeta_k|\mathrm{z}'_{\backslash k}):=$$
$$\delta(z_k,\zeta_k^{>0})\big(\sigma(u_k)h(\zeta_k|\mathrm{z}'_{\backslash k})+(1-\sigma(u_k))\delta(\zeta_k,0)\big),$$

where the function $h(\zeta_k|\boldsymbol{z}'_{\backslash k})$ is defined as:

$$h(\zeta_k|\boldsymbol{z}'_{\backslash k}):=\begin{cases}\dfrac{\prod_{j=\zeta_k+1}^{\tau}(1-g_jf(u_k))}{\sum_{\alpha=1}^{\tau}\prod_{j=\alpha+1}^{\tau}(1-g_jf(u_k))} & \text{for}\qquad \zeta_k>0\\ 0 & \text{otherwise}\end{cases}.$$

It is trivial to see that q^* has the correct marginal distribution over z_k:

$$\begin{aligned}&\sum_{\zeta_k}q^*(z_k,\zeta_k|\boldsymbol{\zeta}'_{\backslash k},\boldsymbol{z}'_{\backslash k})\\ &=\sum_{\zeta_k}\delta(z_k,\zeta_k^{>0})\big(\sigma(u_k)h(\zeta_k|\boldsymbol{z}'_{\backslash k})+(1-\sigma(u_k))\delta(\zeta_k,0)\big)\\ &=\sigma(u_k)^{z_k}(1-\sigma(u_k))^{1-z_k}=p(z_k|\boldsymbol{z}'_{\backslash k}).\end{aligned}$$

We now show that q^* is the unique invariant distribution of T^k. Because of the definition of T^k, we only have to show that $q^*(\zeta_k|\boldsymbol{z}'_{\backslash k})$ is the unique invariant distribution of $T^k(\zeta_k|\zeta'_k,\boldsymbol{z}'_{\backslash k})$. We denote $q^*(\zeta_k=i|\boldsymbol{z}'_{\backslash k})=:q_i$ and $T^k(\zeta_k=i|\zeta'_k=j,\boldsymbol{z}'_{\backslash k})=:T_{ij}$, i.e., we have to show $\forall i\in\{0,1,\ldots,\tau\}:\ q_i=\sum_j T_{ij}q_j$.

It is trivial to show $q_i=\sum_j T_{ij}q_j$ for $1\leq i\leq\tau-1$, as there is only one non-vanishing element of transition operator, namely $T_{i,i+1}$:

$$\begin{aligned}\sum_{j=0}^{\tau}T_{ij}q_j&=T_{i,i+1}q_{i+1}=(1-g_{i+1}f(u_k))q_{i+1}\\ &=(1-g_{i+1}f(u_k))h(\zeta_k=i+1|\mathrm{z}_{\backslash k})\sigma(u_k)\\ &=h(\zeta_k=i|\mathrm{z}_{\backslash k})p(z_k=1|\mathrm{z}_{\backslash k})\overset{!}{=}q_i.\end{aligned}$$

Here we used $q_i=h(\zeta_k=i|\boldsymbol{z}_{\backslash k})\sigma(u_k)$ for $i>0$ and the definition of $h(\zeta_k|\boldsymbol{z}_{\backslash k})$.

Now we show $q_0=\sum_j T_{0j}q_j$ starting from equation (15) and additionally using the relations $\exp(u_k)=\sigma(u_k)/(1-\sigma(u_k))$ and $q_0=1-\sigma(u_k)$ as well as the definition of q_1. We define for the sake of simplicity $\psi:=\sum_{\alpha=1}^{\tau}\prod_{j=\alpha+1}^{\tau}(1-g_jf(u_k))$:

$$\begin{aligned}\sum_{j=0}^{\tau}T_{0j}q_j&=(1-f(u_k))q_0+(1-g_1f(u_k))q_1\\ &=(1-f(u_k))(1-\sigma(u_k))+\frac{\sigma(u_k)}{\psi}\prod_{j=1}^{\tau}(1-g_jf(u_k))\\ &=(1-f(u_k))(1-\sigma(u_k))+\sigma(u_k)f(u_k)\exp(-u_k)\\ &=(1-f(u_k))(1-\sigma(u_k))+f(u_k)(1-\sigma(u_k))\overset{!}{=}q_0.\end{aligned}$$

We finally show $q_\tau=\sum_j T_{\tau j}q_j$, using the definition of $q_\tau=\sigma(u_k)h(\zeta_k=\tau|\boldsymbol{z}_{\backslash k})=\dfrac{\sigma(u_k)}{\psi}$:

$$\begin{aligned}
\sum_{i=0}^{\tau} T_{\tau i} q_i &= \sum_{i=1}^{\tau} g_i f(u_k) q_i + f(u_k) q_0 \\
&= \sum_{i=1}^{\tau} g_i f(u_k) \prod_{j=i+1}^{\tau} (1-g_j f(u_k)) \frac{\sigma(u_k)}{\psi} + f(u_k) q_0 \\
&= \frac{\sigma(u_k)}{\psi} \left(\sum_{i=1}^{\tau} g_i f(u_k) \prod_{j=i+1}^{\tau} (1-g_j f(u_k)) + (1-g_1 f(u_k)) \prod_{j=2}^{\tau} (1-g_j f(u_k)) \right) \\
&= \frac{\sigma(u_k)}{\psi} \left(-\sum_{i=1}^{\tau} (1-g_i f(u_k)) \prod_{j=i+1}^{\tau} (1-g_j f(u_k)) \right. \\
&\left. + \sum_{i=1}^{\tau} \prod_{j=i+1}^{\tau} (1-g_j f(u_k)) + \prod_{j=1}^{\tau} (1-g_j f(u_k)) \right) \\
&= \frac{\sigma(u_k)}{\psi} \left(-\sum_{i=1}^{\tau} \prod_{j=i}^{\tau} (1-g_j f(u_k)) + \sum_{i=0}^{\tau} \prod_{j=i+1}^{\tau} (1-g_j f(u_k)) \right) \\
&= \frac{\sigma(u_k)}{\psi} \left(-\sum_{i=0}^{\tau-1} \prod_{j=i+1}^{\tau} (1-g_j f(u_k)) + \sum_{i=0}^{\tau} \prod_{j=i+1}^{\tau} (1-g_j f(u_k)) \right) \\
&= \frac{\sigma(u_k)}{\psi} \left(\prod_{j=\tau+1}^{\tau} (1-g_j f(u_k)) \right) = \frac{\sigma(u_k)}{\psi} \overset{!}{=} q_\tau.
\end{aligned}$$

The argument that the transition operator T^k is aperiodic and irreducible is similar to the one presented in Lemma 1.

Details to neural sampling with an absolute refractory period in continuous time. In contrast to the discrete time model we define the state space of ζ_k to be $\mathbb{R}^+ \bigcup [-2\epsilon, -\epsilon]$ for $\epsilon > 0$, i.e., as the union of the positive real numbers and a small interval $[-2\epsilon, -\epsilon]$. We will define the sampling operator in such a way that after neuron k was refractory for exactly its refractory period τ, its refractory variable ζ_k is uniformly placed in the small interval $[-2\epsilon, -\epsilon]$, which represents now the resting state and replaces $\zeta_k = 0$. This avoids point measures (Dirac's Delta) on the value $\zeta_k = 0$. This system is still exactly equivalent to the system discussed in the main paper, as all spike-transition probabilities of T for $\zeta_k < 0$ are constant. Hence, it does not matter which values ζ_k assumes with respect to the spike mechanism during its non-refractory period as long as $\zeta_k < 0$.

Definition 2. For a given distribution $p(\boldsymbol{z})$ over the binary variables $\boldsymbol{z} \in \{0,1\}^K$ with $\forall \boldsymbol{z} \in \{0,1\}^K \; p(\boldsymbol{z}) \neq 0$, we define a joint distribution over $(\boldsymbol{\zeta}, \boldsymbol{z})$ with $\boldsymbol{\zeta} \in \mathbb{R}^K$ in the following way:

$$\begin{aligned}
p(\zeta_k | z_k) &:= \begin{cases} 1 & \text{for} \quad 1 \geq \zeta_k > 0 \;\wedge\; z_k = 1 \\ \epsilon^{-1} & \text{for} \quad \zeta_k \in I_\varepsilon \;\wedge\; z_k = 0 \\ 0 & \text{otherwise} \end{cases} \\
p(\boldsymbol{\zeta} | \boldsymbol{z}) &:= \prod_{k=1}^{K} p(\zeta_k | z_k) \\
p(\boldsymbol{\zeta}, \boldsymbol{z}) &:= p(\boldsymbol{\zeta} | \boldsymbol{z}) p(\boldsymbol{z}),
\end{aligned}$$

where $I_\epsilon := [-2\epsilon, -\epsilon]$ is the refractory resting state interval. In accordance with this definition we can also write $p(\zeta_k | z_k) = z_k \chi_{[0,1]}(\zeta_k) +$ $(1 - z_k)\epsilon^{-1} \chi_{I_\epsilon}(\zeta_k)$.

Lemma 6. The distribution $p(\boldsymbol{\zeta}, \boldsymbol{z})$ has the following marginal distribution:

$$\begin{aligned}
p(\zeta_k | \boldsymbol{\zeta}_{\backslash k}) &= \sigma(u_k) \chi_{[0,1]}(\zeta_k) + (1 - \sigma(u_k)) \epsilon^{-1} \chi_{I_\epsilon}(\zeta_k) \\
&= \begin{cases} \sigma(u_k) & \text{for} \quad 1 \geq \zeta_k > 0 \\ (1 - \sigma(u_k)) \epsilon^{-1} & \text{for} \quad \zeta_k \in I_\epsilon \end{cases},
\end{aligned}$$

where $u_k := u_k(\boldsymbol{\zeta}_{\backslash k}^{>0})$.

Definition 3. For $k \in \{1, \ldots, K\}$ and $x \in \mathbb{R}$ the operator T_x^k is defined in the following way for a function $q : \mathbb{R} \to \mathbb{R}$:

$$(T_x^k q)(\zeta_k) := \tau^{-1} (\partial_{\zeta_k} (q(\zeta_k) \chi_{\mathbb{R}^+}(\zeta_k)) - \delta(\zeta_k) F(q) + \exp(x) \delta(\zeta_k - 1) \int_{I_\epsilon} q(\zeta'_k) d\zeta'_k$$

$$+ \; \chi_{I_\epsilon}(\zeta_k) \big(\epsilon^{-1} F(q) - \exp(x) q(\zeta_k) \big) \big).$$

where the functional F is defined as the one-sided limit from above at 0:

$$F(q) := \lim_{x \to 0^+} q(x).$$

The operator T is defined in the following way for a probability distribution $q(\boldsymbol{\zeta})$ on $\mathbb{R}^K$:

$$(Tq)(\boldsymbol{\zeta}) := \sum_{k=1}^{K} (T_{u_k}^k q(\zeta_1, \ldots, \zeta_{k-1}, \cdot, \zeta_{k+1}, \zeta_K))(\zeta_k),$$

where $q(\zeta_1, \ldots, \zeta_{k-1}, \cdot, \zeta_{k+1}, \zeta_K) : \mathbb{R} \to \mathbb{R}$ denotes the function $q(\boldsymbol{\zeta})$ of ζ_k where $\boldsymbol{\zeta}_{\backslash k}$ is held constant and $u_k := u_k(\boldsymbol{\zeta}_{\backslash k}^{>0})$.

The transition operator T defines the following Fokker-Planck equation for a time-dependent distribution $q_t(\boldsymbol{\zeta})$:

$$\partial_t q_t(\boldsymbol{\zeta}) = (T q_t)(\boldsymbol{\zeta}).$$

The jump and drift functions $W^k(\boldsymbol{\zeta} | \boldsymbol{\zeta}')$ and $A^k(\boldsymbol{\zeta})$ associated to the operator T are given by:

$$\begin{aligned}
W^k(\boldsymbol{\zeta} | \boldsymbol{\zeta}') &= \big((\epsilon\tau)^{-1} \chi_{I_\epsilon}(\zeta_k) \delta(\zeta_{k'}) + \delta(\zeta_k - 1) \exp(u_k(\boldsymbol{\zeta}'_{\backslash k}) - \log\tau) \chi_{I_\epsilon}(\zeta_{k'}) \big) \delta(\boldsymbol{\zeta}_{\backslash k} - \boldsymbol{\zeta}'_{\backslash k}) \\
A^k(\boldsymbol{\zeta}) &= -\tau^{-1} \chi_{\mathbb{R}^+}(\zeta_k) \\
\Rightarrow \; (Tq_t)(\boldsymbol{\zeta}) &= -\sum_{k=1}^{K} \partial_{\zeta_k} (A^k(\boldsymbol{\zeta}) q_t(\boldsymbol{\zeta})) + \sum_{k=1}^{K} \int (W^k(\boldsymbol{\zeta} | \boldsymbol{\zeta}') p(\boldsymbol{\zeta}') - W^k(\boldsymbol{\zeta}' | \boldsymbol{\zeta}) p(\boldsymbol{\zeta})) d\boldsymbol{\zeta}'.
\end{aligned}$$

Lemma 7. The operator $T_{u_k}^k$ leaves the conditional distribution $p(\zeta_k | \boldsymbol{\zeta}_{\backslash k})$ invariant with $u_k = u_k(\boldsymbol{\zeta}_{\backslash k}^{>0})$, i.e.:

$$(T_{u_k}^k p(\cdot | \boldsymbol{\zeta}_{\backslash k}))(\zeta_k) = 0.$$

Proof. This is easy to proof using calculus and the relations $\partial_{\zeta_k} \chi_{\mathbb{R}^+}(\zeta_k) = \delta(\zeta_k)$ and $F(p(\cdot | \boldsymbol{\zeta}_{\backslash k})) = \sigma(u_k) = \exp(u_k)(1 - \sigma(u_k))$.

Lemma 8. $p(\boldsymbol{\zeta})$ is an invariant distribution of T, i.e., it is a solution to the invariant Fokker-Planck equation:

$$\partial_t p(\boldsymbol{\zeta}) = (Tp)(\boldsymbol{\zeta}) = 0.$$

Proof. We observe that $T^k(\alpha p) = \alpha T^k p$ for a constant $\alpha \in \mathbb{R}$ (which is not a function of ζ_k). Hence:

$$\begin{aligned}
T_{u_k}^k p(\zeta_1, \ldots, \zeta_{k-1}, \cdot, \zeta_{k+1}, \ldots, \zeta_K) &= T_{u_k}^k (p(\cdot | \boldsymbol{\zeta}_{\backslash k}) p(\boldsymbol{\zeta}_{\backslash k})) \\
&= p(\boldsymbol{\zeta}_{\backslash k}) (T_{u_k}^k p(\cdot | \boldsymbol{\zeta}_{\backslash k})) \\
&= 0.
\end{aligned}$$

The Lemma follows then from the definition of $T := \sum_k T^k_{u_k}$.

Details to neural sampling with a relative refractory period in continuous time. As already assumed in the case of the absolute refractory sampler in continuous time, we define the state space of ζ_k to be $\mathbb{R}^+ \bigcup [-2\epsilon, -\epsilon]$ for $\epsilon > 0$.

Lemma 9. Let g be a continuous, non-negative function $g : [0,1] \to \mathbb{R}^+_0$ with $g(\zeta_k) = 1$ for $\zeta_k \leq 0$. There exists a unique $\mathcal{C}^\infty$ function $f : \mathbb{R} \to \mathbb{R}^+$ with the following property $\forall u \in \mathbb{R}$:

$$f(u) \int_0^1 \exp\left(f(u) \int_0^{\zeta_k} g(\zeta_{k'}) d\zeta_{k'}\right) d\zeta_k = \exp(u). \qquad (16)$$

Proof. We define the function $F : \mathbb{R}^+_0 \to \mathbb{R}$ in the following way:

$$F(x) := x \int_0^1 \exp(x\alpha(\zeta_k)) d\zeta_k,$$

where $\alpha(r) := \int_0^r g(\zeta_{k'}) d\zeta_{k'}$. From $g(\zeta_k) \geq 0$ we can follow that $\alpha : [0,1] \to \mathbb{R}^+_0$ is non-negative. $F(x)$ is differentiable with the derivative:

$$F'(x) = \int_0^1 \exp(x\alpha(\zeta_k)) d\zeta_k + x \int_0^1 \exp(x\alpha(\zeta_k))\alpha(\zeta_k) d\zeta_k$$
$$\Rightarrow F'(x) > 0.$$

Hence F is strictly monotonously increasing. Furthermore, the following relations hold:

$$F(0) = 0$$
$$F(x) \geq x.$$

Therefore the equation:

$$F(x) = \exp(u),$$

has exactly one solution $f(u)$ with $F(f(u)) = \exp(u)$ in $\mathbb{R}^+$. From applying the implicit function theorem to $F(x,u) := F(x) - \exp(u)$ it follows that f is $\mathcal{C}^\infty$.

Definition 4. For all $k \in \{1, \ldots, K\}$ and $x \in \mathbb{R}$ the operator T^k_x is defined in the following way for a function $q : \mathbb{R} \to \mathbb{R}$:

$$\begin{aligned}(T^k_x q)(\zeta_k) := \tau^{-1}(&\partial_{\zeta_k}(q(\zeta_k)\chi_{\mathbb{R}^+}(\zeta_k)) - \delta(\zeta_k)q(\zeta_k) \\ &+ f(x)\delta(\zeta_k - 1) \int_{\mathbb{R}} g(\zeta_{k'})q(\zeta_{k'}) d\zeta_{k'} \\ &+ \chi_{I_\varepsilon}(\zeta_k)\varepsilon^{-1}F(q) - f(x)q(\zeta_k)g(\zeta_k)).\end{aligned}$$

The transition operator T^k_x defines the following Fokker-Planck equation for a time-dependent distribution $q_t(\zeta_k)$:

$$\partial_t q_t(\zeta_k) = (T^k_x q_t)(\zeta_k).$$

The jump and drift functions $W^k(\zeta_k|\zeta_{k'})$ and $A^k(\zeta_k)$ associated to the operator T^k_x are given by:

$$W^k(\zeta_k|\zeta_{k'}) = (\varepsilon\tau)^{-1}\chi_{I_\varepsilon}(\zeta_k)\delta(\zeta_{k'}) + \tau^{-1}\delta(\zeta_k - 1)f(x)g(\zeta_{k'})$$
$$A^k(\zeta_k) = -\tau^{-1}\chi_{\mathbb{R}^+}(\zeta_k)$$
$$\Rightarrow \ (T^k_x q_t)(\zeta_k) = -\partial_{\zeta_k}(A^k(\zeta_k)q_t(\zeta_k)) + \int \left(W^k(\zeta_k|\zeta_{k'})p(\zeta_{k'}) - W^k(\zeta_{k'}|\zeta_k)p(\zeta_k)\right) d\zeta_{k'}.$$

Lemma 10. For all $k = 1, \ldots, K$ the invariant distribution $q^*(\zeta_k|\mathbf{z}_{\setminus k})$ of the operator $T^k_{u_k}$ fulfills $\int \delta(z_k, \zeta_k^{>0}) q^*(\zeta_k|\mathbf{z}_{\setminus k}) d\zeta_k = p(z_k|\mathbf{z}_{\setminus k})$.

Proof. We define the distribution $q^*(\zeta_k|\mathbf{z}_{\setminus k})$ as:

$$q^*(\zeta_k|\mathbf{z}_{\setminus k}) = (1 - \sigma(u_k))\left(f(u_k)\chi_{[0,1]}(\zeta_k)\exp(f(u_k)\alpha(\zeta_k)) + \epsilon^{-1}\chi_{I_\epsilon}(\zeta_k)\right),$$

where $\alpha(\zeta_k) := \int_0^1 g(\zeta'_k) d\zeta'_k$. By applying the operator $T^k_{u_k}$ to q^* one can verify that $T^k_{u_k} q^* = 0$ holds using the definition of $f(u_k)$ given in (16). Furthermore we can compute the ratio:

$$\frac{\int_0^1 q^*(\zeta_k|\mathbf{z}_{\setminus k}) d\zeta_k}{\int_{I_\epsilon} q^*(\zeta_k|\mathbf{z}_{\setminus k}) d\zeta_k} = \frac{p(z_k = 1|\mathbf{z}_{\setminus k})}{p(z_k = 0|\mathbf{z}_{\setminus k})}$$
$$= f(u_k) \int_0^1 \exp\left(f(u_k) \int_0^{\zeta_k} g(\zeta_{k'}) d\zeta_{k'}\right) d\zeta_k = \exp(u_k).$$

Details to the computer simulations

The simulation results shown in Figure 2, Figure 3 and Figure 4 used the biologically more realistic neuron model with the relative refractory mechanism. During all experiments the first second of simulated time was discarded as burn-in time. The full list of parameters defining the experimental setup is given in Table 1. All occurring joint probability distributions are Boltzmann distributions of the form given in equation (5). Example Python [64] scripts for neural sampling from Boltzmann distributions are available on request and will be provided on our webpage. The example code comprises networks with both absolute and relative refractory mechanism. It requires standard Python packages only and is readily executable.

Details to Figure 2: Neuron model with relative refractory mechanism. The three refractory functions $g(\zeta)$ of panel (B) as well as all other simulation parameters are listed in Table 1. Panel (C) shows the corresponding functions $f(u)$, which result from numerically solving equation (11). The spike patterns in panel (D) show the response of the neurons when the membrane potential is low ($u_k = -1$ for $0 < t < 250$ ms) or high ($u_k = +2$ for 250 ms $< t < 500$ ms). These membrane potentials encode $p(z_k = 1) = 0.269$ and $p(z_k = 1) = 0.881$, respectively according to (3) and (4). The binary state $z_k = 1$ is indicated by gray shaded areas of duration $\tau \cdot dt = 20$ ms after each spike.

Details to Figure 3: Sampling from a Boltzmann distribution by spiking neurons with relative refractory mechanism. We examined the spike response of a network of 40 randomly connected neurons which sampled from a Boltzmann distribution. The excitabilities b_k as well as the synaptic weights $W_{ki} (= W_{ik})$ were drawn from Gaussian distributions (with diagonal elements $W_{ii} = 0$). For the full list of parameters please refer to Table 1. One second of the arising spike pattern is shown in panel (A). The average firing rate of the network was 13.9 Hz.

Table 1. List of parameters of the computer simulations.

Description	Variable	Value	Figure	Comment
Simulation Time				
Simulation step size	dt	$[1]ms$	2–7	interpretation of an MCMC step
Burn-in time	t_{burn}	$[1]s$	2–7	before recording spikes
Simulation time	t_{sim}	$[0.5]s$	2	
		$[10^4]s$	3,5–7	
		$[20]s$	4	$[10^4]s$ for Figure 3C
Network				
Number of neurons	K	3	2	unconnected
		40	3,5,6	randomly connected
		217	4	
		10	7	100 networks
Connection radius		0	2	
		∞	3,5–7	
		8	4	
Recurrent weights	W_{ki}	$\mathcal{N}(0,0.3^2)$	3,5–7	from Gaussian distribution
Falling edge	τ_+	[20]ms	6,7	for realistic PSP shapes
Rising edge	τ_-	[3]ms	6,7	
Scaling factor	λ	20/17	6,7	
Neuron Model				
Number recovery steps	τ	20	2–7	PSP duration $=\tau\cdot dt=[20]ms$
Refractory function	$g(\zeta)$	$\left[4(1-\zeta)+\frac{1}{2\pi}\sin(8\pi\zeta)\right]$	2↑	normalized to $\zeta\in[0,1]$,
		$\left[1-\zeta+\frac{1}{2\pi}\sin(2\pi\zeta)\right]$	2–7	$[x] := \min\{1,\max\{0,x\}\}$
		$\left[1-2\zeta+\frac{1}{2\pi}\sin(4\pi\zeta)\right]$	2↓,7	
Excitability	b_k	-1 or 2	2	defines membrane potential u_k
		$\mathcal{N}(-1.5,0.5^2)$	3,5–7	from Gaussian distribution
		0	4	initial value
Tuning Function, Training and Inference (Figure 4)				
Peakedness	κ	3	4	measured: 1.78 ± 0.15
Base sensitivity	v_0	0.05	4	measured: 0.017 ± 0.009
Sensitivity contrast	C	0.9	4	measured: 0.760 ± 0.020
Training samples	N_{train}	10^5	4	
Decorrelation steps		20	4	for contrastive divergence
Learning rate	η	10^{-4}	4	
Number of neurons clamped on/off		4/4	4	

doi:10.1371/journal.pcbi.1002211.t001

To highlight the internal dynamics of the neuron model, the values of the refractory function $g(\zeta_{26})$, the membrane potential u_{26} and the instantaneous firing rate r_{26} of neuron v_{26} (indicated with red spikes) are shown in panel (B). Here, the instantaneous firing rate r_{26} is defined for the discrete time Markov chain as

$$r_{26} = p(\text{spike})/dt = T^{26}(\tau|\zeta_{26}, z_{\setminus 26})/dt = g(\zeta_{26})\cdot f(u_{26})/dt. \quad (17)$$

As stated before, the neuron model with relative refractory mechanism $g_k(\zeta)$ does not entail the correct overall invariant distribution $p(\mathbf{z})$. To estimate the impact of this approximation on the joint network dynamics, we compared the distribution $p(z_{24},\ldots,z_{28})$ over five neurons (indicated by gray background in A) in the spiking network with the correct distribution obtained from Gibbs sampling. The probabilities were estimated from 10^7 samples. A more quantitative analysis of the approximation quality of neural sampling with a relative refractory mechanism is provided below.

Details to Figure 4: Modeling perceptual multistability as probabilistic inference with neural sampling. We demonstrate probabilistic inference and learning in a network of orientation selective neurons. As a simple model we consider a network of 217 neurons on a hexagonal grid as shown in panel (F). Any two neurons with distance ≤ 8 were synaptically connected (neighboring units had distance 1). For the remaining parameters of the network and neuron model please refer to Table 1. Each neuron featured a π-periodic tuning curve as depicted in panel (B):

$$V_k(\varphi) = v_0 + C\cdot\exp[\kappa\cdot\cos(2(\varphi-\bar{\varphi}_k))-\kappa] \quad (18)$$

with base sensitivity v_0, contrast C, peakedness κ and preferred orientation $\bar{\varphi}_k$. The preferred orientations $\bar{\varphi}_k$ of the neurons were chosen to cover the entire interval $[0,\pi)$ of possible orientations with equal spacing and were randomly assigned to the neurons.

For simplicity we did not incorporate the input dynamics in our probabilistic model, but rather trained the network directly like a fully visible Boltzmann machine. We used for this purpose a standard Boltzmann machine learning rule known as contrastive divergence [41,65]. This learning rule requires posterior samples $\tilde{\boldsymbol{z}}$, i.e., network states under the influence of the present input, and approximate prior samples $\boldsymbol{z}^\star$, which reflect the probability distribution of the network in the absence of stimuli. The update rules for synaptic weights and neuronal excitabilities read:

$$\begin{aligned}\Delta W_{ki} &= \eta_{ki}\cdot(\tilde{z}_k\tilde{z}_i - z_k^\star z_i^\star)\\ \Delta b_k &= \eta\cdot(\tilde{z}_k - z_k^\star)\\ \eta_{ki} &= \begin{cases}\eta & \text{if } v_k \text{ and } v_i \text{ are connected}\\ 0 & \text{otherwise}.\end{cases}\end{aligned} \tag{19}$$

While more elaborate policies can speed up convergence, we simply used a global learning rate η which was constant in time. The values of W_{ki} and b_k were initialized at 0. We generated binary training patterns in the following way:

1. A global orientation φ was drawn uniformly from $[0,\pi)$,
2. each neuron was independently set to be active with probability $p(z_k = 1) = V_k(\varphi)$,
3. the resulting network state $\tilde{\boldsymbol{z}}$ was taken as posterior sample.

To obtain an approximate prior sample $\boldsymbol{z}^\star$ we let the network run for a short time freely starting from $(\tilde{\boldsymbol{\zeta}},\tilde{\boldsymbol{z}})$. The variables $\tilde{\mathbf{z}}$ were also assumed to be observed with $\tilde{\zeta}_k \sim$ iid. uniformly in $\{1,\ldots,\tau\}$ if $\tilde{z}_k = 1$ and $\tilde{\zeta}_k = 0$ otherwise. After evolving freely for 20 time steps, the resulting network state $\boldsymbol{z}^\star$ was taken as approximate prior sample and $\boldsymbol{W}$ and $\boldsymbol{b}$ were updated according to (19). This process was repeated $N_{train} = 10^5$ times. As a result, neurons with similar preferred orientations featured excitatory synaptic connections ($W_{ki} = 6.4\cdot10^{-3} \pm 6.7\cdot10^{-3}$ = mean ± standard deviation of weight distribution), those with dissimilar orientations maintained inhibitory synapses ($W_{ki} = -4.9\cdot10^{-3} \pm 5.2\cdot10^{-3}$). Here, preferred orientations $\bar{\varphi}_i$ and $\bar{\varphi}_j$ are defined as similar if $V_i(\bar{\varphi}_j) - v_0 = V_j(\bar{\varphi}_i) - v_0 > 0.5\,C$, otherwise they are dissimilar. Neuronal biases converged to $b_k = -0.08 \pm 0.03$.

We illustrate the learned prior distribution $p(\boldsymbol{z})$ of the network through sampled states when the network evolved freely. As seen in panel (D), the population vector – a 2-dimensional projection of the high dimensional network state – typically reflected an arbitrary, yet coherent, orientation (for the definition of the population vector see below). Each dot represents a sampled network state $\boldsymbol{z}$.

To apply an ambiguous cue, we clamped 8 out of 217 neurons: Two units with $\bar{\varphi}_k \approx \pi/4$ and two with $\bar{\varphi}_k \approx 3\pi/4$ were set active, two units with $\bar{\varphi}_k \approx 0$ and two with $\bar{\varphi}_k \approx \pi/2$ were set inactive. This led to a bimodal posterior distribution as shown in panel (E). The sampling network represented this distribution by encoding either global perception separately: The trace of network states $\boldsymbol{z}(t)$ roamed in one mode for multiple steps before quickly crossing the state space towards the opposite percept.

We define the population vector $\mathbf{x}$ of a network state $\boldsymbol{z}$ as a function of the preferred orientations of all active units:

$$\boldsymbol{x} = (x_0, x_{\pi/4}) = \sum_{k=1}^{K} z_k\cdot(\cos 2\tilde{\varphi}_k, \sin 2\tilde{\varphi}_k). \tag{20}$$

This definition of $\boldsymbol{x}$ is not based on the preferred orientations $\bar{\varphi}_k$ which are used for generating external input to the network from a given stimulus with orientation φ. It is rather based on the preferred orientations $\tilde{\varphi}_k$ measured from the network response. We used population vector decoding based on the measured values $\tilde{\varphi}_k$, as they are conceptually closer to experimentally measurable preferred orientations, and this decoding hence does not require knowledge of the (unobservable) $\bar{\varphi}_k$. For every neuron v_k the preferred orientation $\tilde{\varphi}_k$ was measured in the following way. We estimated a tuning curve $\tilde{V}_k(\varphi)$ by a van-Mises fit (of the form (18)) to data from stimulation trials in which neuron v_k was not clamped, i.e., where v_k was only stimulated by recurrent input (feedforward input was modeled by clamping 8 out of 217 neurons as a function of stimulus orientation φ as before). Due to the structured recurrent weights, the experimentally measured tuning curves $\tilde{V}_k(\varphi)$ were found to be reasonably close to the tuning curves $V_k(\varphi)$ used for external stimulation. $\tilde{\varphi}_k$ was set to the preferred orientation of $\tilde{V}_k(\varphi)$ (localization parameter of the van-Mises fit). The measured values $\tilde{\varphi}_k$ turned out to be consistent with the preferred orientations $\bar{\varphi}_k$ ($\bar{\varphi}_k - \tilde{\varphi}_k = 6\cdot10^{-4} \pm 8.3\cdot10^{-3}$ averaged over all K neurons). The mean and standard deviation of the remaining parameter values v_0, C and κ of the fitted tuning curves $\tilde{V}_k(\varphi)$ are listed in Table 1 next to the ones used for stimulation.

The population vector $\boldsymbol{x}$ was defined in (20) with the argument $2\tilde{\varphi}_k$ (instead of $\tilde{\varphi}_k$) as orthogonal orientations should cancel each other and neighborhood relations should be respected. For example neurons with $\tilde{\varphi}_k = \epsilon$ and $\tilde{\varphi}_k = \pi - \epsilon$ contribute similarly to the population vector for small ε. But counter to intuition the population vector of a state $\boldsymbol{z}$ with dominant orientation φ_z will point into direction $\varphi_x = 2\varphi_z$. For visualization in panel (D) and (E) we therefore rescaled the population vector: If $(x_0, x_{\pi/4}) \mapsto (r_x, \varphi_x)$ in polar coordinates, then the dot is located at $(r_x, \varphi_x/2)$ in accord with intuition. The black semicircles equal $|\boldsymbol{x}| = r_x = 45$.

The population vector $(x_0, x_{\pi/4}) \in \mathbb{R}^2$ was also used for measuring the dominance durations shown in panel (C). To this $\mathbb{R}^2$ was divided into 3 areas: (a) $x_{\pi/4} < -35$, (b) $-35 \le x_{\pi/4} \le 35$, (c) $35 < x_{\pi/4}$. We detected a perceptual switch when the network state entered area (a) or (c) while the previous perception was (c) or (a), respectively.

In panel (F) neurons v_k with $z_k = 1$ are plotted with their preferred orientation color code, inactive neurons are displayed in white. Cells marked by a dot (●) were part of the observed variables $\boldsymbol{o}$. The three network states correspond to $\boldsymbol{z}(t_i)$ with $t_1 = 100$ ms, $t_2 = 250$ ms and $t_3 = 400$ ms in the spike pattern in panel (G). The spike pattern shows the response of the freely evolving units around a perceptual switch during sampling from the posterior distribution. The corresponding trace of the population vector is drawn as black line in panel (E). The width of the light-gray shaded areas in the spike pattern equals the PSP duration $\tau\cdot dt$, i.e., neurons that spiked in these intervals were active in the corresponding state in (F).

Firing statistics of neural sampling networks

In previous sections it was shown that a spiking neural network can draw samples from a given joint distribution which is in a well-defined class of probability distributions (see the neural computability condition (4)). Here, we examine some statistics of individual neurons in a sampling network which are commonly used to analyze experimental data from recordings. The spike trains and membrane potential data are taken from the simulation presented in Figure 3.

Figure 5A,B exemplarily show the distribution of the membrane potential u_k and the interspike interval (ISI) histogram of a single neuron, namely neuron v_{26} which was already considered in Figure 3B. The responses of other neurons yield qualitatively similar statistics. The bell-shaped distribution of the membrane potential is commonly observed in neurons embedded in an active network [66]. The ISI histogram reflects the reduced spiking probability immediately after an action potential due the refractory mechanism. Interspike intervals larger than the refractory time constant $\tau{\cdot}dt = 20$ ms roughly follow an exponential distribution. Similar ISI distributions were observed during in-vivo recordings in awake, behaving monkeys [67].

Figure 5C shows a scatterplot of the coefficient of variation (CV) of the ISIs versus the average ISI for each neuron in the network. The neurons exhibited a variety of average firing rates between 3.5 Hz and 31.5 Hz. Most of the neurons responded in a highly irregular manner with a CV ≈ 1. Neurons with high firing rates had a slightly lower CV due to the increased influence of the refractory mechanism The dashed line marks the CV of a Poisson process, i.e., a memoryless spiking behavior. The CV of neuron v_{26} is marked by a cross. The structure of this plot resembles, e.g., data from recordings in behaving macaque monkeys [68] (but note the lower average firing rate).

Approximation quality of neural sampling with different neuron and synapse models

The theory of the neuron model with absolute refractory mechanism guarantees sampling form the correct distribution. In contrast, the theory for the neuron model with a relative refractory mechanism only shows that the sampling process is "locally correct", i.e., that it would yield correct conditional distributions $p(z_k|\mathbf{z}_{\backslash k})$ for each individual neuron if the state of the remaining network $\mathbf{z}_{\backslash k}$ stayed constant. Therefore, the stationary distribution of the sampling process with relative refractory mechanism only provides an approximation to the target distribution. In the following we examine the approximation quality and robustness of sampling networks with different refractory mechanisms for target Boltzmann distributions with parameters randomly drawn from different distributions. Furthermore, we investigate the effect of additive PSP shapes with more realistic time courses.

We generated target Boltzmann distributions with randomly drawn weights W_{ki} and biases (excitabilities) b_k and computed the similarity between these reference distributions and the corresponding neural sampling approximations. The setup of these simulations is the same as for the simulation presented in Figure 3. As we aimed to compare the distribution $q^*(\mathbf{z})$ sampled by the network with the exact Boltzmann distribution $p(\mathbf{z})$, we reduced the number of neurons per network to $K = 10$. This resulted in a state space of 2^{10} possible network states $\mathbf{z}$ for which the normalization constant for the target Boltzmann distribution could be computed exactly. The weight matrix W was constraint to be symmetric with vanishing diagonal. Off-diagonal elements were drawn from zero-mean normal distributions with three different standard deviations $\sigma = 0.03$, $\sigma = 0.3$ and $\sigma = 3$, whereas the b_k were sampled from the same distribution as in Figure 3. For every value of the hyperparameter σ we generated 100 random distributions. For Boltzmann distributions with small weights ($\sigma = 0.03$), the RVs are nearly independent, whereas distributions with intermediate weights ($\sigma = 0.3$) show substantial statistical dependencies between RVs. For very large weights ($\sigma = 3$), the probability mass of the distributions is concentrated on very few states (usually 90% on less than 10 out of the 2^{10} states). Hence, the range of the hyperparameter $0.03 \leq \sigma \leq 3$ considered here covers a range a very different distributions.

The approximation quality of the sampled distribution was measured in terms of the Kullback-Leibler divergence between the target distribution p and the neural approximation q^*

$$D_{\mathrm{KL}}(p||q^*) = \sum_{\mathbf{z}} p(\mathbf{z}) \log \frac{p(\mathbf{z})}{q^*(\mathbf{z})}. \tag{21}$$

We estimated q^* from 10^7 samples for each simulation trial using a Laplace estimator, i.e., we added a priori 1 to the number of occurrences of each state $\mathbf{z}$.

Table 2 shows the means and the standard deviations of the Kullback-Leibler divergences between the target Boltzmann distributions and the estimated approximations stemming from neural sampling networks with three different neuron and synapse models: the exact model with absolute refractory mechanism and two models with different relative refractory mechanisms shown in the bottom and middle row in Figure 2B. Additionally, as a reference, we provide the (analytically calculated) Kullback-Leibler divergences for fully factorized distributions, i.e., $q^*(\mathbf{z}) = \Pi_k\, q^*(z_k)$ with correct marginals $q^*(z_k) = p(z_k)$ but independent variables z_i, z_j for $i \neq j$.

The absolute refractory model provides the best results as we expected due to the theoretical guarantee to sample from the correct distribution (the non-zero Kullback-Leibler divergence is caused by the estimation from a finite number of samples). The models with relative refractory mechanism provide faithful approximations for all values of the hyperparameter σ considered here. These relative refractory models are characterized by the theory to be "locally correct" and turn out to be much more

Table 2. Approximation quality of networks with different refractory mechanisms.

σ	**Absolute refractory**	**Rel. late recovery**	**Rel. moderate recovery**	**Prod. of marginals**
0.03	$(3.10 \pm 0.18){\cdot}10^{-4}$	$(3.21 \pm 0.15){\cdot}10^{-4}$	$(3.33 \pm 0.17){\cdot}10^{-4}$	$(4.65 \pm 1.28){\cdot}10^{-4}$
0.3	$(2.98 \pm 0.19){\cdot}10^{-4}$	$(3.20 \pm 0.15){\cdot}10^{-4}$	$(3.58 \pm 0.3){\cdot}10^{-4}$	$(4.94 \pm 1.91){\cdot}10^{-2}$
3.0	$(1.32 \pm 0.45){\cdot}10^{-4}$	$(4.20 \pm 8.70){\cdot}10^{-3}$	$(1.00 \pm 1.82){\cdot}10^{-2}$	$(5.36 \pm 6.71){\cdot}10^{-1}$

Mean and standard deviation of the Kullback-Leibler divergence $D_{\mathrm{KL}}(p||q^*)$ between reference Boltzmann distributions p and neural sampling approximations q^* for three different neuron models (corresponding to columns) and three different values for the reference distribution hyperparameter σ (corresponding to rows). The parameter σ controls the standard deviation of the weights of the reference distributions $p(\mathbf{z})$. In case of very strong synaptic interactions (leading to sharply peaked distributions, $\sigma = 3$) the approximation quality of the spiking network degrades, if the neurons feature a relative refractory mechanism. The data was computed from 100 randomly generated Boltzmann distributions and their neural approximations for each value of σ.

doi:10.1371/journal.pcbi.1002211.t002

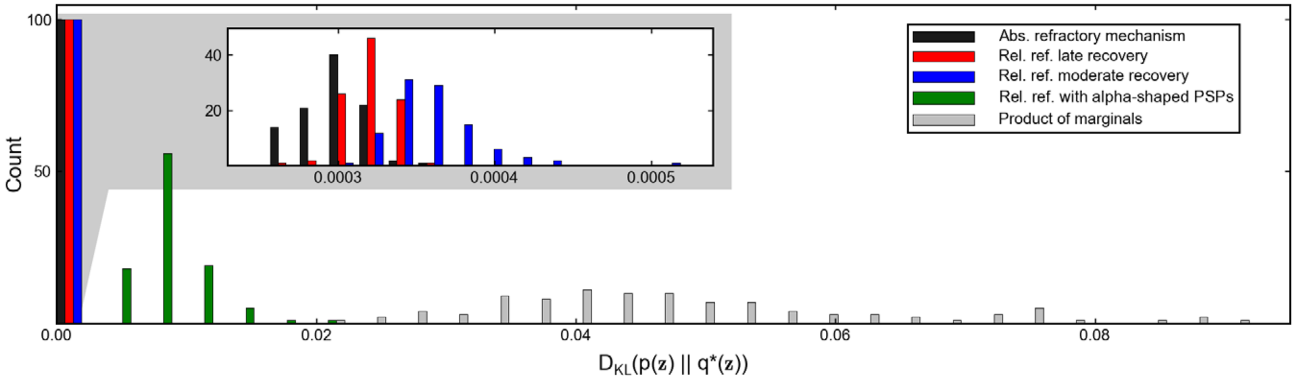

Figure 6. Comparison of neural sampling with different neuron and synapse models. The figure shows a histogram of the Kullback-Leibler divergence between 100 different Boltzmann distributions over K = 10 variables (with parameters randomly drawn, see setup of Figure 3) and approximations stemming from different neural sampling networks. Networks with absolute refractory mechanism provide the best approximation (as expected from theoretical guarantees). Networks consisting of neurons with relative refractory mechanisms, with only "locally" correct sampling, also provide a close fit to the true distribution (see inset) compared to a fully factorized approximation (assuming correct marginals and independent variables). Furthermore, it can be seen that sampling networks with more realistic, alpha-shaped, additive PSPs still fit the true distribution reasonably well.
doi:10.1371/journal.pcbi.1002211.g006

accurate approximations than fully factorized distributions if substantial statistical dependencies between the RVs are present (i.e., $\sigma = 0.3$, $\sigma = 3$). As expected, a late recovery of the refractory function $g(\zeta)$ is beneficial for the approximation quality of the model as it is closer to an absolute refractory mechanism. Figure 6 shows the full histograms of the Kullback-Leibler divergences for the intermediate weights group ($\sigma = 0.3$). Systematic deviations due to the relative refractory mechanism are on the same order as the effect of estimating from finite samples (as can be seen, e.g., from a comparison with the absolute refractory model which has 0 systematic error). For completeness, we mention that the divergences of the fully factorized distributions of 2 out of the 100 networks with $D_{\mathrm{KL}} > 0.1$ are not shown in the plot.

The theorems presented in this article assumed renewed (i.e., non-additive), rectangular PSPs. In the following we examine the effect of additive PSPs with more realistic time courses. We define additive, alpha-shaped PSPs in the following way. The influence Δu_{ki} of each presynaptic neuron v_i on the postsynaptic membrane potential u_k is modeled by convolving the input spikes with a kernel κ:

$$\Delta u_{ki}(t) = W_{ki} \cdot \sum_f \kappa(t - t_i^f) \qquad (22)$$

where $\kappa(s) = \lambda \cdot (e^{-s/\tau_+} - e^{-s/\tau_-})$ for $s \geq 0$ and $\kappa(s) = 0$ for $s < 0$, and t_i^f for $f \in \mathbb{N}$ are the spike times of the presynaptic neuron v_i. The time constant governing the rising edge of the PSPs was set to $\tau_- = 3$ ms. The time constant controlling the falling edge was chosen equal to the duration of rectangular PSPs, $\tau_+ = \tau \cdot dt = 20$ ms. The scaling parameter λ was set such that the time integral over a single PSP matches the time integral over the theoretically optimal rectangular PSP, i.e., $\lambda = \tau \cdot dt / (\tau_+ - \tau_-) = 20/17$. These parameters display a simple and reasonable choice for the purpose of this study (an optimization of λ, τ_+ and τ_- is likely to yield an improved approximation quality). Figure 7A shows the resulting shape of the non-rectangular PSP. Furthermore the time course of the function $g(\zeta_k(t))$ caused by a single spike of neuron v_k is shown in order to illustrate that the time constants of g and of a PSP are closely related due to the

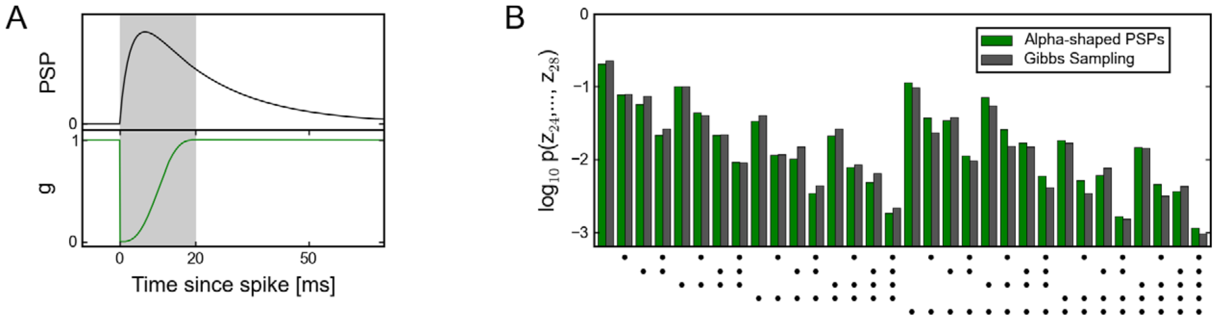

Figure 7. Sampling from a Boltzmann distribution with more realistic PSP shapes. (A) The upper panel shows the shape of a single PSP elicited at time $t = 0$. The lower panel shows the time course of the refractory function $g(\zeta_k(t))$ caused by a single spike of neuron v_k at $t = 0$. The grey-shaded area of length $\tau \cdot dt = 20ms$ indicates the interval of neuron v_k being active (i.e., $z_k = 1$) due to a single spike of neuron v_k at time $t = 0$. (B) Shown is the probability distribution of 5 out of 40 neurons. The plot is similar to Figure 3C, however it is generated with a sampling network that features alpha-shaped, additive PSPs. It can be seen that the network still produces a reasonable approximation to the true Boltzmann distribution (determined by Gibbs sampling).
doi:10.1371/journal.pcbi.1002211.g007

assumption $\tau_+ = \tau \cdot dt$ made above. Preliminary and non-exhaustive simulations seem to suggest that the choice $\tau_+ = \tau \cdot dt$ yields better approximation quality than setting $\tau_+ \gg \tau \cdot dt$ or $\tau_+ \ll \tau \cdot dt$; however it is very well possible that a mismatch between τ_+ and $\tau \cdot dt$ can be compensated for by adapting other parameters, e.g., the PSP magnitude or a specific choice of the refractory function g. Figure 7B shows the results of an experiment, similar to the one presented in Figure 3C , with additive, alpha-shaped PSPs and relative refractory mechanism. While differences to Gibbs sampling results are visible, the spiking network still captures dependencies between the binary random variables quite well.

For a quantitative analysis of the approximation quality, we repeated the experiment of Figure 6 with additive, alpha-shaped PSPs (shown as green bars). The Kullback-Leibler divergence $D_{\mathrm{KL}}(p||q^*)$ to the true distribution is clearly higher compared to the case of renewed, rectangular PSPs. Still networks with this more realistic synapse model account for dependencies between the random variables $\boldsymbol{z}$ and yield a better approximation of $p(\boldsymbol{z})$ than fully factorized distributions.

Acknowledgments

We would like to thank Mihai Petrovici, Robert Legenstein and Samuel Gershman for helpful discussions.

Author Contributions

Conceived and designed the experiments: LB JB WM. Performed the experiments: JB. Analyzed the data: LB JB. Contributed reagents/materials/analysis tools: LB JB BN WM. Wrote the paper: LB JB BN WM. Designed network model and its theoretical analysis: LB.

References

1. Rolls ET, Deco G (2010) The Noisy Brain: Stochastic Dynamics as a Principle of Brain Function. Oxford University Press.
2. Cannon R, O'Donnell C, Nolan M (2010) Stochastic ion channel gating in dendritic neurons: morphology dependence and probabilistic synaptic activation of dendritic spikes. PLoS Comput Biol 6: e1000886.
3. Flight M (2010) Synaptic transmission: On the probability of release. Nat Rev Neurosci 9: 736–737.
4. Azouz R, Gray CM (1999) Cellular mechanisms contributing to response variability of cortical neurons in vivo. J Neurosci 19: 2209–2223.
5. GerstnerW, KistlerWM (2002) Spiking Neuron Models. Cambridge: Cambridge University Press.
6. Brascamp JW, van Ee R, Noest AJ, Jacobs RHAH, van den Berg AV (2006) The time course of binocular rivalry reveals a fundamental role of noise. J Vis 6: 1244–1256.
7. Fiser J, Chiu C, Weliky M (2004) Small modulation of ongoing cortical dynamics by sensory input during natural vision. Nature 431: 573–583.
8. Ringach DL (2009) Spontaneous and driven cortical activity: implications for computation. Curr Opin Neurobiol 19: 1–6.
9. Geman S, Geman D (1984) Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Trans Pattern Anal Mach Intell 6: 721–741.
10. Rao RPN, Olshausen BA, Lewicki MS (2002) Probabilistic Models of the Brain. MIT Press.
11. Doya K, Ishii S, Pouget A, Rao RPN (2007) Bayesian Brain: Probabilistic Approaches to Neural Coding. MIT-Press.
12. Körding KP, Wolpert DM (2004) Bayesian integration in sensorimotor learning. Nature 427: 244–247.
13. Kersten D, Mamassian P, Yuille A (2004) Object perception as Bayesian inference. Annu Rev Psychol 55: 271–304.
14. Gopnik A, Tenenbaum JB (2007) Bayesian special section: Introduction; Bayesian networks, Bayesian learning and cognitive development. Dev Sci 10: 281–287.
15. Lee TS, Mumford D (2003) Hierarchical Bayesian inference in the visual cortex. J Opt Soc Am A 20: 1434–1448.
16. Hoyer P, Hyvärinen A (2003) Interpreting neural response variability as Monte Carlo sampling of the posterior. In: Proceedings of the 16th Conference on Advances in Neural Information Processing Systems; December 2002. Vancouver, Canada, NIPS 2002.
17. Sundareswara R, Schrater PR (2008) Perceptual multistability predicted by search model for bayesian decisions. J Vis 8: 1–19.
18. Gershman SJ, Vul E, Tenenbaum J (2009) Perceptual multistability as Markov chain Monte Carlo inference. In: Proceedings of the 22nd Conference on Advances in Neural Information Processing Systems; December 2008. Vancouver, Canada, NIPS 2008.
19. Griffiths TL, Kemp C, Tenenbaum JB (2008) Bayesian models of cognition. In: Sun R, ed. Handbook of Computational Cognitive Modeling Cambridge University Press. pp 59–100.
20. Yang T, Shadlen MN (2007) Probabilistic reasoning by neurons. Nature 447: 1075–1080.
21. Gold JI, Shadlen MN (2007) The neural basis of decision making. Annu Rev Neurosci 30: 535–574.
22. Sadaghiani S, Hesselmann G, Friston KJ, Kleinschmidt A (2010) The relation of ongoing brain activity, evoked neural responses, and cognition. Front Syst Neurosci 4: Artikel 20.
23. Pearl J (1988) Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann.
24. Friston KJ, Daunizeau J, Kilner J, Kiebel SJ (2010) Action and behavior: a free-energy formulation. Biol Cybern 102: 227–260.
25. Toussaint M (2009) Probabilistic inference as a model of planned behavior. Künstliche Intelligenz 3: 23–29.
26. Toussaint M, Goerick C (2010) A Bayesian view on motor control and planning. In: Sigaud O, Peters J, eds. From motor to interaction learning in robots Studies in Computational Intelligence. Springer. pp 227–252.
27. Tenenbaum JB, Griffiths TL, Kemp C (2006) Theory-based bayesian models of inductive learning and reasoning. Trends Cogn Sci 10: 309–318.
28. Griffiths TL, Tenenbaum JB (2006) Optimal predictions in everyday cognition. Psychol Sci 17: 767–773.
29. Oaksford M, Chater N (2007) Bayesian Rationality: The Probabilistic Approach to Human Reasoning. Oxford University Press.
30. Rao RPN (2007) Neural models of Bayesian belief propagation. In: Doya K, Ishii S, Pouget A, Rao RPN, eds. Bayesian Brain. Cambridge, MA: MIT-Press. pp 239–267.
31. Steimer A, Maass W, Douglas R (2009) Belief-propagation in networks of spiking neurons. Neural Comput 21: 2502–2523.
32. Deneve S (2008) Bayesian spiking neurons I: Inference. Neural Comput 20: 91–117.
33. Litvak S, Ullman S (2009) Cortical circuitry implementing graphical models. Neural Comput 21: 1–47.
34. Sahani M, Dayan P (2003) Doubly distributional population codes: Simultaneous representation of uncertainty and multiplicity. Neural Comput 15: 2255–2279.
35. Fiser J, Berkes P, Orbn G, Lengyel M (2010) Statistically optimal perception and learning: from behavior to neural representations. Trends Cogn Sci 14: 119–130.
36. Ackley DH, Hinton GE, Sejnowski TJ (1985) A learning algorithm for boltzmann machines. Cogn Sci 9: 147–169.
37. Hinton GE, Osindero S, Teh YW (2006) A fast learning algorithm for deep belief nets. Neural Comput 18: 1527–1554.
38. Andrieu C, Freitas ND, Doucet A, Jordan MI (2003) An introduction to MCMC for machine learning. Mach Learn 50: 5–43.
39. Grimmett GR, Stirzaker DR (2001) Probability and Random Processes. Oxford University Press, 3rd edition.
40. Nessler B, Pfeiffer M, MaassW (2009) Hebbian learning of Bayes optimal decisions. In: Proceedings of the 21th Conference on Advances in Neural Information Processing Systems; December 2008. Vancouver, Canada, NIPS 2008.
41. Hinton GE (2010) Learning to represent visual input. Philos Trans R Soc Lond B Biol Sci 365: 177–184.
42. Pillow JW, Shlens J, Paninski L, Sher A, Litke AM, et al. (2008) Spatio-temporal correlations and visual signalling in a complete neuronal population. Nature 454: 995–999.
43. Gardiner C (2004) Handbook of Stochastic Methods. 3rd ed. Springer.
44. Leopold DA, Wilke M, Maier A, Logothetis NK (2002) Stable perception of visually ambiguous patterns. Nat Neurosci 5: 605–609.
45. Blake R, Logothetis NK (2002) Visual competition. Nat Rev Neurosci 3: 13–21.
46. Alais D, Blake R (2005) Binocular Rivalry. MIT Press.
47. Bartels A, Logothetis NK (2010) Binocular rivalry: a time dependence of eye and stimulus contributions. J Vis 10: 3.
48. Binzegger T, Douglas RJ, Martin KA (2004) A quantitative map of the circuit of cat primary visual cortex. J Neurosci 24: 8441–8453.
49. Binzegger T, Douglas R, Martin K (2009) Topology and dynamics of the canonical circuit of cat V1. Neural Netw 22: 1071–1078.
50. Churchland MM, Yu BM, Cunningham JP, Sugrue LP, Cohen MR, et al. (2010) Stimulus onset quenches neural variability: a widespread cortical phenomenon. Nat Neurosci 13: 369–378.
51. Berkes P, Orbán G, Lengyel M, Fiser J (2011) Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment. Science 331: 83.
52. Bishop CM (2006) Pattern Recognition and Machine Learning. New York: Springer.

53. Koller D, Friedman N (2009) Probabilistic Graphical Models: Principles and Techniques. MIT Press.
54. Hopfield JJ, Tank DW (1985) ''Neural'' computation of decisions in optimization problems. Biol Cybern 52: 141–152.
55. Vul E, Pashler H (2008) Measuring the crowd within: Probabilistic representations within individuals. Psychol Sci 19: 645–647.
56. Denison S, Bonawitz E, Gopnik A, Griffiths T (2009) Preschoolers sample from probability distributions. In: Proceedings of the 32nd Annual Conference of the Cognitive Science Society; 29 July - 1 August 2009; Amsterdam, Netherlands. CogSci 2009.
57. Hinton G, Brown A (2000) Spiking Boltzmann machines. In: Proceedings of the 13th Conference on Advances in Neural Information Processing Systems; December 1999. Vancouver, Canada: NIPS 1999.
58. Zemel R, Huys QJM, Natarajan R, Dayan P (2005) Probabilistic computation in spiking populations. In: Proceedings of the 17th Conference on Advances in Neural Information Processing Systems; December 2004. Vancouver, Canada: NIPS 2004.
59. Kenet T, Bibitchkov D, Tsodyks M, Grinvald A, Arieli A (2003) Spontaneously emerging cortical representations of visual attributes. Nature 425: 954–956.
60. Fox MD, Raichle ME (2007) Spontaneous fluctuations in brain activity observed with functional magnetic resonance imaging. Nat Rev Neurosci 8: 700–711.
61. Antic SD, ZhouWL, Moore AR, Short SM, Ikonomu KD (2010) The decade of the dendritic NMDA spike. J Neurosci Res 88: 2991–3001.
62. Merolla P, Arthur J, Shi BE, Boahen K (2007) Expandable networks for neuromorphic chips. IEEE Trans Circuits Syst I Regul Pap 54: 301–311.
63. Bruederle D, Bill J, Kaplan B, Kremkow J, Meier K, et al. (2010) Live demonstration: Simulatorlike exploration of cortical network architectures with a mixed-signal VLSI system. In: Proceedings of the IEEE International Symposium on Circuits and Systems; 30May - 2 June 2010. Paris, France: ISCAS 2010.
64. Python (2011) The python language reference. Available: http://docs.python.org/reference/.
65. Hinton GE (2002) Training products of experts by minimizing contrastive divergence. Neural Comput 14: 1771–1800.
66. Pospischil M, Piwkowska Z, Bal T, Destexhe A (2009) Characterizing neuronal activity by describing the membrane potential as a stochastic process. J Physiol Paris 103: 98–106.
67. Shinomoto S, Kim H, Shimokawa T, Matsuno N, Funahashi S, et al. (2009) Relating neuronal firing patterns to functional differentiation of cerebral cortex. PLoS Comput Biol 5: e1000433.
68. Softky W, Koch C (1993) The highly irregular firing of cortical cells is inconsistent with temporal integration of random epsps. J Neurosci 13: 334–350.