



Ensembles of Spiking Neurons with Noise Support Optimal Probabilistic Inference in a Dynamically Changing Environment

Robert Legenstein*, Wolfgang Maass

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

Abstract

It has recently been shown that networks of spiking neurons with noise can emulate simple forms of probabilistic inference through “neural sampling”, i.e., by treating spikes as samples from a probability distribution of network states that is encoded in the network. Deficiencies of the existing model are its reliance on single neurons for sampling from each random variable, and the resulting limitation in representing quickly varying probabilistic information. We show that both deficiencies can be overcome by moving to a biologically more realistic encoding of each salient random variable through the stochastic firing activity of an ensemble of neurons. The resulting model demonstrates that networks of spiking neurons with noise can easily track and carry out basic computational operations on rapidly varying probability distributions, such as the odds of getting rewarded for a specific behavior. We demonstrate the viability of this new approach towards neural coding and computation, which makes use of the inherent parallelism of generic neural circuits, by showing that this model can explain experimentally observed firing activity of cortical neurons for a variety of tasks that require rapid temporal integration of sensory information.

Citation: Legenstein R, Maass W (2014) Ensembles of Spiking Neurons with Noise Support Optimal Probabilistic Inference in a Dynamically Changing Environment. *PLoS Comput Biol* 10(10): e1003859. doi:10.1371/journal.pcbi.1003859

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

Received: April 22, 2014; **Accepted:** August 16, 2014; **Published:** October 23, 2014

Copyright: © 2014 Legenstein, Maass. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability: The authors confirm that all data underlying the findings are fully available without restriction. All relevant data are within the paper and its Supporting Information files.

Funding: Written under partial support by the European Union project #604102 (HBP) and the Austrian Science Fund FWF #1753-N23 (PNEUMA). 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.

* Email: robert.legenstein@igi.tugraz.at

Introduction

Humans and animals are confronted with various situations where the state of some behaviorally relevant time-varying random variable $v(t)$ is only accessible through noisy observations $Y(t)$. It is then essential to estimate the current value of that random variable $v(t)$, and to update this belief on the basis of further evidence. Over the last three decades we have learned from various experiments that monkeys are able to perform such operations. In the classical random-dot motion task, monkeys are confronted with dots on a screen moving in random directions, where a random subset of dots moves coherently. Monkeys are able to determine the direction of coherent motion even for low coherency levels [1]. A more recent study has shown that the firing rate of neurons in parietal cortex are proportional to the momentary log-likelihood ratio of a rewarded action for the given sensory evidence [2], suggesting that cortical circuits perform some form of probabilistic inference to determine the value of the hidden variable that represents rewarded actions. In yet another experiment, Cisek and Kalaska [3] studied macaque monkeys in an ambiguous target task. A visual spatial cue and a color cue, which were separated by a memory epoch, determined the rewarded direction of an arm movement, see Figure 1A. Ambiguity about the rewarded action after the first cue was reflected in the firing activity of dorsal premotor cortex (PMd) neurons, see Figure 1B. When the second cue determined the single rewarded

action, only neurons tuned to the rewarded movement direction remained active. This finding suggests that estimates for the value of a salient time-varying random variable $v(t)$ (getting rewarded for carrying out a specific action) are represented and updated through the current firing activity of different ensembles E_i of neurons, one for each possible value i of the random variable $v(t)$.

We show that despite of their diversity, all these tasks can be viewed as probabilistic inference tasks, where some internal belief about the current value i of a hidden random variable $v(t)$ (e.g., which action is most likely to be rewarded at the end of a trial) needs to be updated based on often ambiguous sensory evidence $Y(t)$ (moving dots, visual cues, etc.). We will distinguish 5 different classes of such tasks (labeled A - E in Results) that differ for example with regard to the time scale on which the hidden variable $v(t)$ changes, or prior knowledge about the expected change of $v(t)$. These tasks can not be solved adequately through a Hidden Markov Model (HMM). The reason is that a HMM generates at each moment in time just a single guess for the current value of an unknown variable. It is therefore not able to work with more complex temporary guesses, say that an unknown variable has probably value 1 or 2, but definitely not value 3. Obviously such advanced representations are necessary in order to make decisions that depend on the integration of numerous temporally dispersed cues.

For all these classes of computational tasks there exist theoretically optimal solutions that can be derived within a

Author Summary

The Markov Chain Monte Carlo (MCMC) approach to probabilistic inference for a distribution P is to draw a sequence of samples from P and to carry out computational operations via simple online computations on such a sequence. But such a sequential computational process takes time, and therefore this simple version of the MCMC approach runs into problems when one needs to carry out probabilistic inference for rapidly varying distributions. This difficulty also affects all currently existing models for emulating MCMC sampling by networks of stochastically firing neurons. We show here that by moving to a space-rate approach where salient probabilities are encoded through the spiking activity of ensembles of neurons, rather than by single neurons, this problem can be solved. In this way even theoretically optimal models for dealing with time varying distributions through sequential Monte Carlo sampling, so called particle filters, can be emulated by networks of spiking neurons. Each spike of a neuron in an ensemble represents in this approach a “particle” (or vote) for a particular value of a time-varying random variable. In other words, neural circuits can speed up computations based on Monte Carlo sampling through their inherent parallelism.

probabilistic inference framework. If one assumes that the hidden random variable $v(t)$ is static (i.e., $v(t)=i$ for some i and all times t), evidence provided by the temporal stream of observations has to be integrated in order to infer the internal belief about the value of the random variable $v(t)$. In such *evidence integration*, an initial prior belief formalized as a probability distribution $P_0(v(t))$ is updated over time in order to infer the time-varying posterior distribution $P_t(v(t)|\mathbf{Y}(t))$, where $\mathbf{Y}(t)$ denotes all evidence up to time $t > t_0$. For example, an observation at time t that is likely for $v(t)=i$ will increase the probability of state i at time t , while the probability of values under which the observation is unlikely will be decreased. *Bayesian filtering* generalizes evidence integration to time-varying random variables. It is often assumed that the dynamics of the random variable is time-independent. Bayesian filtering then infers the posterior $P_t(v(t)|\mathbf{Y}(t))$ by taking the assumed dynamics of the time-varying random variable $v(t)$ into account. For example, if value i is currently likely, and state i is likely to transition to state j , then the probability for state j will gradually increase over time. For many important tasks, the dynamics of the random variable $v(t)$ is not identical at all times but rather depends on context. For example, the change of body position in space (formalized as a hidden random variable $v(t)$) depends on motor actions. We refer to Bayesian filtering with context dependent dynamics as *context-dependent Bayesian*

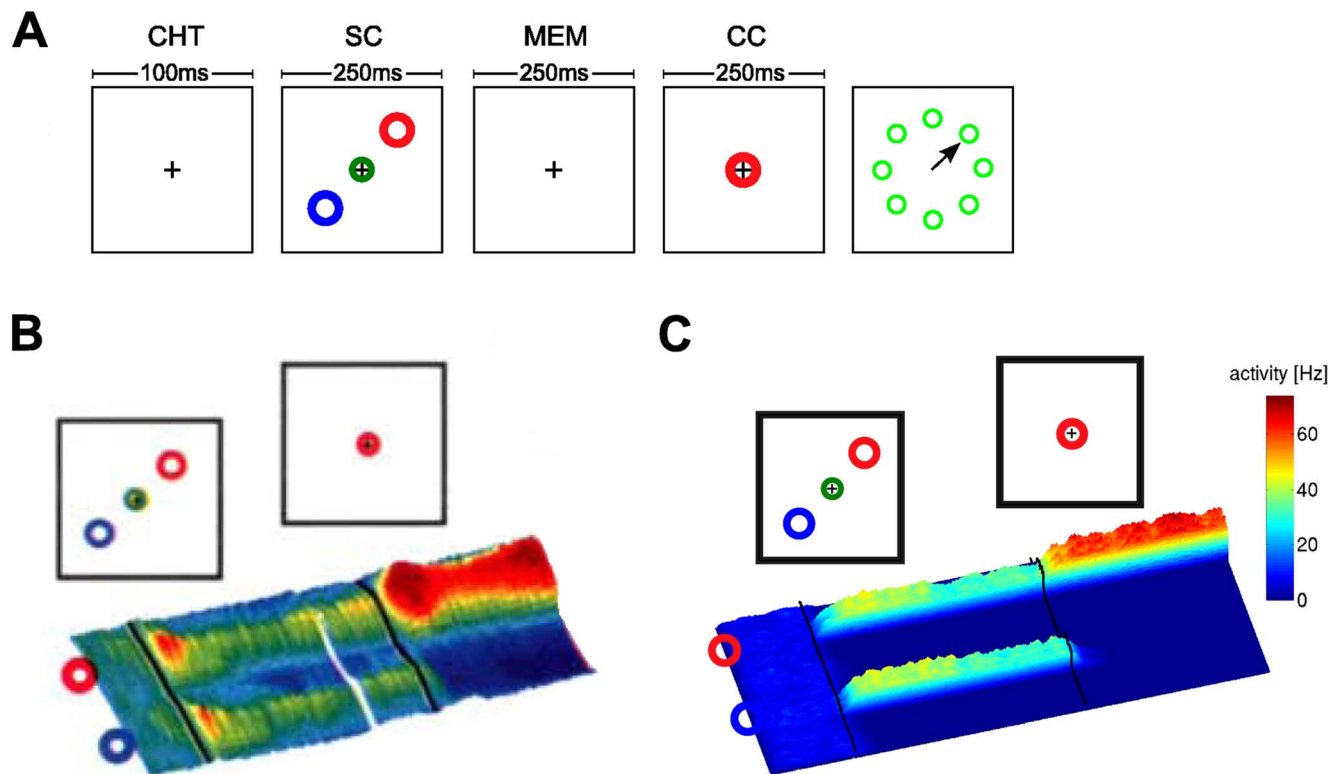


Figure 1. Representation of a belief in dorsal premotor cortex (PMd) in the ambiguous target task. A) Task structure. After an initial fixation (center-hold time; CHT), the spatial cue (SC) is shown in the form of two color markers at one of eight possible locations and displaced from each other by 180 degrees. They mark two potentially rewarded movement directions. After a memory epoch (MEM), the color cue (CC) is shown at the fixation cross. The rewarded movement direction is defined by the direction of matching color in the color cue (time periods of simulation indicated). **B)** Firing activity of neurons in dorsal premotor cortex during the task. Before the spatial cue is shown, neurons are diffusely active. As the spatial cue is shown, neurons with preferred directions consistent with the spatial cue increase their firing rate and others are silenced. This circuit behavior is retained during the memory epoch. As the color cue is presented, neurons with consistent preferred directions increase their firing rates. **C)** Simulation result for a circuit that performs evidence integration in ENS coding (activity smoothed; horizontal axis: time). Neurons are ordered by their preferred direction. Panel B modified with permission from [52]. doi:10.1371/journal.pcbi.1003859.g001

filtering. It infers $P_t(v(t)|\mathbf{Y}(t),\mathbf{C}(t))$, where $\mathbf{C}(t)$ denotes all context information received up to time t .

In previous work, it was shown that networks of spiking neurons can embody a probability distribution through their stochastic spiking activity. This enables a neural system to carry out probabilistic inference through sampling (e.g., estimate of a marginal probability by observing the firing rate of a corresponding neuron) [4]. This model for probabilistic inference in networks of spiking neurons was termed neural sampling. However, neural sampling does not provide a suitable model for the representation and updating of quickly-varying distributions as it is needed for the tasks discussed above, since a good estimate of the current value of $v(t)$ can only be read out after several samples have been observed. Another deficit of the aforementioned simple form of the neural sampling model is, that each salient random variable is represented through the firing activity of just a single neuron. This is unsatisfactory because it does not provide a network computation that is robust against the failures of single neurons. In fact, the representation of random variables through single neurons is not consistent with experimental data, see Figure 1B. In addition it requires unbiologically strong synaptic connections in order to ensure that the random variable that is represented by such single neuron has an impact on other random variables, or on downstream readouts. Furthermore, downstream readout neurons are required to integrate (count) spikes of such neuron over intervals of several hundred ms or larger, in order to get a reasonable estimate of the probability that is represented through the firing rate of the neuron (i.e., in order to estimate a posterior marginal, which is an important form of probabilistic inference).

We examine in this article therefore an extension of the neural sampling model, where random variables (e.g. internal beliefs) are represented through a space-rate code of neuronal ensembles. In other words, we are making stronger use of the inherent parallelism of neural systems. In this *ensemble based neural sampling (ENS) code*, the percentage of neurons in an ensemble E_i that fire within some short (e.g. 20 ms) time interval encodes the internal belief (or estimated probability) that a random variable currently has a specific value i .

This variation of the neural sampling model is nontrivial, since one tends to lose the link to the theory of sampling/probabilistic inference if one simply replaces a single neuron by an ensemble of neurons. We show however that ensemble based neural sampling is nevertheless possible, and is supported by a rigorous theory. In this new framework downstream neurons can read out current internal estimates in the ENS code just through their standard integration of postsynaptic potentials. We prove rigorously that this generates unbiased estimates, and we also show on what parameters the variance of this estimate depends. Furthermore we explore first steps of a theory of neural computation with the ENS code. We show that nonlinear computation steps that are needed for optimal integration of time-varying evidence can be carried out within this spike-based setting through disinhibition of neurons. Hence networks of spiking neurons with noise are in principle able to approximate theoretically optimal filtering operations – such as evidence integration and context-dependent Bayesian filtering – for updating internal estimates for possible causes of external stimuli. We show in particular, that networks of spiking neurons with noise are able to emulate state-of-the-art probabilistic methods that enable robots to estimate their current position on the basis of multiple ambiguous sensory cues and path integration. This provides a first paradigm for the organization of brain computations that are able to solve generic self-localization tasks. The resulting model is especially suited as “computational engine” for an intention-based neural coding framework, as proposed in

[5]. Intention-based neural coding is commonly observed in lateral intraparietal cortex (area LIP) of monkeys, where neurons encode a preference for a particular target of a saccade within the visual field [1].

The remainder of this paper is structured as follows. First we introduce ENS coding. We then discuss basic properties of the ENS code. In *Computational operations through ensemble-based neural sampling*, we show how basic computations on time-varying internal beliefs can be realized by neural circuits in ENS coding. This section is structured along 5 classes of computational tasks of increasing complexity (*Task class A* to *Task class E*). Within these task classes, we present computer simulations where the characteristics of these neural circuits are analyzed and compared to experimental results. A discussion of the main findings of this paper and related work can be found in *Discussion*. Detailed derivations and descriptions of computer simulations are provided in *Methods*.

Results

Ensemble-based neural sampling

In neural sampling [4], each neuron in a network represents a binary random variable. Spike generation is stochastic, with a probability that depends on the current membrane potential. At each time t , the activity of each neuron is mapped to a sample for the value of the corresponding variable by setting the value to 1 if the neuron has spiked in $[t-\tau, t]$ for some small τ (e.g. 20 ms). It was shown that under certain conditions on the membrane potentials of the neurons, the network converges to a stationary distribution that corresponds to the posterior distribution of the represented random variables for the given evidence. Evidence is provided to the circuit by clamping the activities of a subset of neurons during inference. The marginal distribution for a given variable can be read out by observing the firing activity of the corresponding neuron in the stationary distribution.

Neural sampling is an implementation of the Markov Chain Monte Carlo (MCMC) sampling approach (see e.g. [6]) in networks of spiking neurons. By definition, it does not provide a suitable model for the representation of time-varying distributions, since samples are generated in a sequential manner. Convergence to the stationary distribution in MCMC sampling can take substantial time, the readout of marginal distributions demands spike counts of neurons over extended periods of time, and MCMC sampling is only defined for the fictional case of stationary external inputs. But also time-varying distributions can theoretically be handled through Monte Carlo sampling if one has a sufficiently parallelized stochastic system that can generate at each time point t simultaneously several samples $v_1(t), \dots, v_L(t)$ from the time varying distribution $P_t(v(t))$, and by carrying out simple computational operations on this batch of samples in parallel. The resulting computational model is usually referred to as *particle filter*, a special case of sequential Monte Carlo sampling [6,7]. Here each sample $v_i(t)$ from a batch $v_1(t), \dots, v_L(t)$ that is generated at time t is referred to as a *particle*.

To port the idea of neural sampling to the representation of time-varying distributions through neuronal ensembles, we therefore consider N ensembles X_1, \dots, X_N that collectively encode the belief about the value of a random variable $v(t)$ with range $\{1, \dots, N\}$ in terms of a probability distribution $P_t(v(t))$. We refer to the value of a random variable also as the hidden state, or simply the state of the variable. We will in the following omit the subscript t in P_t for notational convenience (formally, we consider a family of variables, indexed by t , that defines a random process, see [8]; $P(v(t))$ is then the distribution over the member $v(t)$ of this

family). Each ensemble X_i consists of M neurons x_i^1, \dots, x_i^M , where we refer to M as the *ensemble size*. We interpret a spike in the circuit as one sample from $P(v(t))$, i.e., one concrete value for the represented variable drawn according to the distribution [4,9]. In particular, if a spike is elicited by some neuron in ensemble X_i , then this value is i . See Figure 2A for an illustration of sample-based representations.

A downstream neuron can evaluate how many spikes it received from each ensemble within its membrane time constant τ through summation of excitatory postsynaptic potentials (EPSPs) caused by spikes from ensemble neurons. We denote the EPSP-filtered spike train of neuron x_i^j by $x_i^j(t)$ (see *Methods* for a precise definition) and adopt rectangular EPSP shapes of length τ in the following, similar to those recorded at the soma of pyramidal neurons for dendrite-targeting synaptic inputs (see Figure 1 in [10]). The sum of all EPSPs from an ensemble X_i , denoted by $X_i(t) \equiv \sum_n x_i^n(t)$, is then the number of samples for hidden state i in the time window from $t-\tau$ to t . The number of samples $X_i(t)$ is thus directly accessible to downstream neurons. We refer to $X_i(t)$ as the (non-normalized) probability mass for hidden state i . The use of plateau-like EPSP shapes is motivated from the need to count spikes in some predefined time interval. An alternative motivation that is based on the idea that the EPSP weights a spike by the

probability that it belongs to the most recent samples is given in *Text S1*.

The represented distribution can be estimated by the relative portion of spikes from the ensembles

$$\tilde{P}(v(t)=i) = \frac{X_i(t)}{\sum_{j=1}^N X_j(t)}, \quad (1)$$

where we assume that at least one sample is available. See Figure 2B for an intuitive illustration of ENS coding. In this representation, probabilities are temporally filtered by the EPSPs. Hence, the ability of the code to capture fast dynamics of distributions depends on the length of EPSPs, where shorter time constants give rise to faster tracking. Due to the stochasticity of the sampling process, $\tilde{P}(v(t)=i)$ is a random variable that assumes different values each time the distribution is represented. We demand in ENS coding that the expected value of $\tilde{P}(v(t)=i)$ is equal to the temporally filtered represented probability $P(v(t)=i)$ for all states i , see *Methods* for details.

We will see in the construction of computational operations in ENS coding that downstream neurons do not have to carry out the division of eq. (1). Instead, for these operations, they can compute

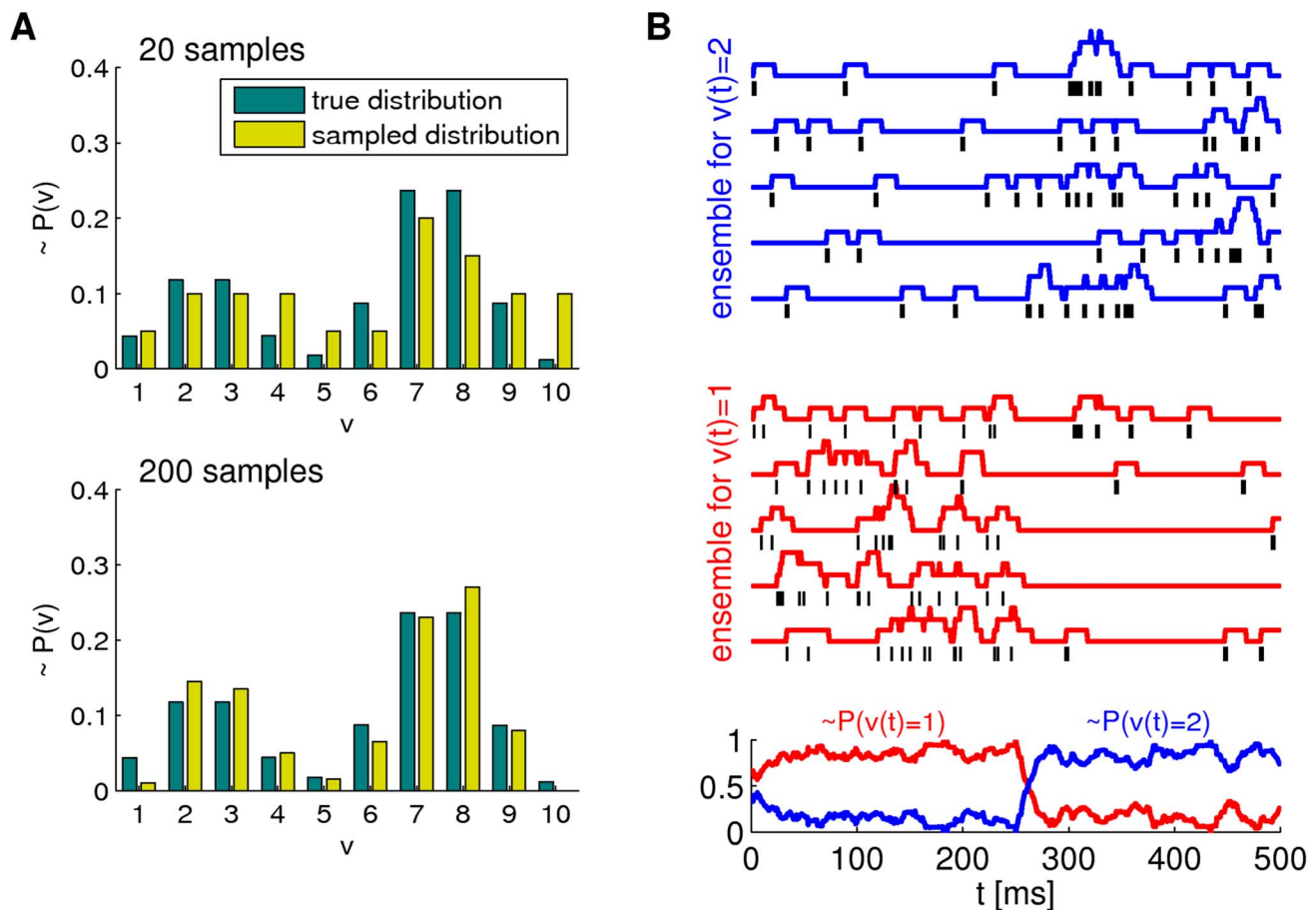


Figure 2. Spikes as samples from probability distributions. **A** Sample-based representations of probability distributions. True distribution of a random variable v (green) and approximated distribution (yellow) based on 20 samples (top) and 200 samples (bottom) **B** Interpretation of the spiking activity of two neuronal ensembles as samples from a probability distribution over a temporally changing random variable $v(t)$. Shown is an example for a random variable $v(t)$ with two possible states. Black lines in the top traces indicate action potentials in two ensembles (5 neurons per state shown). Traces above spikes show EPSP-filtered versions of these spikes (red: state 1; blue: state 2). Bottom plot: Estimated probabilities for state 1 (red) and state 2 (blue) according to eq. (1) based on the spiking activity of 10 neurons per state. doi:10.1371/journal.pcbi.1003859.g002

with the non-normalized probability masses $X_i(t)$ that they obtain through summation of EPSPs from ensemble neurons. The reason is that normalization is not necessary in the representation of a distribution in ENS coding. It is rather the relative portion of spikes for each value of the random variable that defines the represented distribution. Of course, activity needs to be kept in some reasonable range, but this can be done in a rather relaxed manner.

Basic properties of the ENS code

Assume that samples (spikes) for state i are produced by a Poisson process with rate proportional to the represented probability, i.e., the rates $\rho_i^m(t)$ of neurons in ensemble X_i are given by $\rho_i^m(t) = \rho_{\max} P(v(t)=i)$, where ρ_{\max} is a constant that defines the maximal instantaneous rate of each neuron. $X_i(t)$ is then a random variable distributed according to a Poisson distribution with intensity $LP(v(t)=i)$, where the *estimation sample size* $L = M\rho_{\max}\tau$ is the average number of spikes produced by all ensembles within a time span of τ .

We show in *Methods* that in this case, $\bar{P}(v(t)=i)$ is an unbiased estimator of the probability of state i at time t filtered by the EPSPs, hence, an ENS code is established. An important question is how parameters of a circuit influence the fidelity of the encoding. To answer this question, we investigated the variance of the estimator. It is inversely proportional to the ensemble size M , the maximal firing rate ρ_{\max} , and the membrane time constant τ if the number of samples within τ is large, see *Methods*. Hence, the accuracy can be increased by increasing the ensemble size, the firing rate of neurons, and the time constant of neuronal integration. Note however that an increase of the latter will lead to more temporal filtering of the distribution.

We close this discussion by considering the relation between the instantaneous firing rates of ensemble neurons and the mean represented probability mass $X_i(t)$. The probability mass is not an instantaneous function of the ensemble firing rate, since at time t there are still past samples that influence $X_i(t)$ through their EPSPs. A past sample becomes invalid after time τ , when the associated EPSP vanishes. The instantaneous firing rates *change* the mass through the production of novel samples. Consider given continuous firing rates $\rho_i^m(\cdot)$. Under mild assumptions on the firing rates of ensemble neurons (see *Methods*), the change of the expected probability mass $\frac{d}{dt}\langle X_i(t) \rangle$ is then given by

$$\frac{d}{dt}\langle X_i(t) \rangle = \sum_m \rho_i^m(t) - \sum_m \rho_i^m(t-\tau) \quad (2)$$

where the expectation is taken over realizations of spike trains for the given instantaneous rates. Here, the first term is due to the production of novel samples and the second term due to old samples that become invalid. In summary, the membrane potentials of the neurons determine – through the firing rate – the rate of change of the represented probability mass in ENS-coding.

Computational operations through ensemble-based neural sampling

We address now the question how basic computations on time-varying internal beliefs can be realized by neural circuits in ENS coding. Spiking activity of excitatory neurons is modeled according to the stochastic Spike Response model [11,12]. In this model, each neuron x_i^n emits a Poisson spike train with instantaneous firing rate

$$\rho_i^n(t) = f(u_i^n(t)). \quad (3)$$

Here, f denotes a link function that links the somatic membrane potential $u_i^n(t)$ to the instantaneous firing rate. Typically, the link function is either an exponential function or a non-negative linear function. We consider in this article a non-negative linear link function $f(u) = [u]^+$ [12], where $[u]^+$ is u for non-negative u and 0 otherwise.

We discuss five classes of computational tasks.

Tasks class A. In these tasks, the state of a random variable $v^X(t)$ has to be inferred in ENS coding given the belief about a variable $v^Z(t)$ in ENS coding, where the distribution over $v^X(t)$ depends solely on the current state of $v^Z(t)$. The computational operation needed to solve such problems is simpler than the other ones considered here in the sense that the distribution over $v^X(t)$ can be directly inferred from recent samples for $v^Z(t)$. We will use this operation several times to read out the belief about a rewarded motor action (v^X) from the internal belief about some random variable $v^Z(t)$.

Tasks class B. This class consists of tasks that can be solved through evidence integration. In other words, the state of a random variable $v(t)$ has to be estimated based on evidence $\mathbf{Y}(t)$, where $v(t)$ is assumed to be static during each trial. Examples for such tasks are the ambiguous target task, the random-dot motion task, and the probabilistic inference task from [2]. We will exhibit a spiking neural network architecture that approximates optimal solutions for these tasks in ENS coding and compare its behavior to experimental results.

Tasks class C. Also in these tasks, evidence $\mathbf{Y}(t)$ has to be integrated to estimate the state of a random variable $v(t)$. However, the state of $v(t)$ may change over time according to known time-independent stochastic dynamics. Bayesian filtering provides an optimal solution for such tasks. We will extend the circuit architecture from task class B to approximate Bayesian filtering in ENS coding and test its performance in a generic task setup.

Tasks class D. In this class of tasks, the dynamics of the random variable $v(t)$ may change during the task, and changes are indicated by some context-variable $c(t)$. Note that task classes B and C are special cases of this task class. We refer to the optimal solution as context-dependent Bayesian filtering. An approximation based on sequential Monte Carlo sampling is particle filtering. We will extend our circuit architecture to perform full particle filtering in ENS coding. We will show how the important problem of self-localization can be solved by this architecture.

Tasks class E. Finally, we will discuss tasks where the context variable $c(t)$ is not explicitly given but has to be estimated from noisy evidence. Hence, $c(t)$ – which determines the dynamics of $v(t)$ – has also to be considered a random variable. We will treat such tasks by combining two particle filters. One particle filter estimates $c(t)$ and provides context for another particle filter that estimates $v(t)$. As an example, we will reconsider the ambiguous target task. We show that a belief about the current stage within a series of trials can be generated in order to decide whether evidence should be further integrated for the belief about the rewarded action or whether a new trial has started and the belief should be reset to some prior distribution.

Task class A: Simple probabilistic dependencies

We first discuss how the belief for a random variable $v^X(t)$ can be inferred in ENS coding given the belief over a variable $v^Z(t)$.

Consider a random variable $v^X(t)$ for which the distribution depends solely on the current state of a random variable $v^Z(t)$. The task is to infer the distribution over $v^X(t)$ for the current distribution over $v^Z(t)$. This operation is needed for example in typical decision making tasks where inference about a rewarded action has to be performed according to the belief about some hidden variable that is based on sensory information. For example, the rewarded movement direction has to be guessed, based on the belief about the perceived cue combination in the ambiguous target task.

Assume that $P(v^Z(t))$ is constant and represented by neurons z_j^m through ENS coding with estimation sample size L . We are looking for a neural circuit that represents the posterior belief

$$P(v^X(t)=i) = \sum_j P(v^X=i|v^Z=j)P(v^Z(t)=j) \quad (4)$$

in ENS coding. Here, $P(v^X=i|v^Z=j)$ are the known conditional probabilities that determine the dependencies between $v^X(t)$ and $v^Z(t)$.

We consider a layer of ensembles X_i that receive feed-forward synaptic input from ensembles Z_j representing $P(v^Z(t))$, see Figure 3. The membrane potentials of neurons x_i^n are given by

$$u_i^n(t) = \sum_{j,m} w_{ij} z_j^m(t), \quad (5)$$

where w_{ij} denotes the efficacy of the synapse connecting presynaptic neuron z_j^m to postsynaptic neuron x_i^n (we assume for simplicity of notation that all weights between two ensembles are identical). Consider the estimator $P(v^X(t)=i)$ for synaptic efficacies

$$w_{ij} = \alpha \frac{1}{M\tau} P(v^X=i|v^Z=j), \quad (6)$$

where M is the ensemble size of the ensembles X_i , and $\alpha > 0$ is some constant. Due to the stochastic nature of neurons, this estimator is a random variable. Its expected value (with respect to realizations of spikes trains in all ensembles) is equal to the posterior probability and the estimation sample size is αL (see *Methods*). Thus, the layer represents the posterior distribution in ENS coding. The estimate at some specific time t is however variable due to variability in spike counts of both the representation of $P(v^Z(t))$ and the representation of $P(v^X(t))$. An analysis of the variance of the posterior representation is given in *Methods*.

We will use such a layer with feed-forward input several times in our simulations to infer a belief about rewarded actions for a given belief about the state of a random variable and thus refer to it as an *action readout layer*.

We will also use a special case of this operation where the distribution $P(v^Z(t))$ is simply copied, i.e., the state of the random variable $v^X(t)$ is assumed to be identical to the state of $v^Z(t)$. In other words, $P(v^X=i|v^Z=j)$ is 1 for $i=j$ and 0 for $i \neq j$. The copy operation is thus performed for weights $w_{ii} = \frac{1}{M\tau}$ and $w_{ij} = 0$ for $i \neq j$ (where we used $\alpha = 1$).

Task class B: Evidence integration

In pure evidence integration, the value of the random variable $v(t)$ is assumed to be constant and only indirectly observable via stochastic point-event observations. Point-event observations are assumed to arise according to Poisson processes with instantaneous

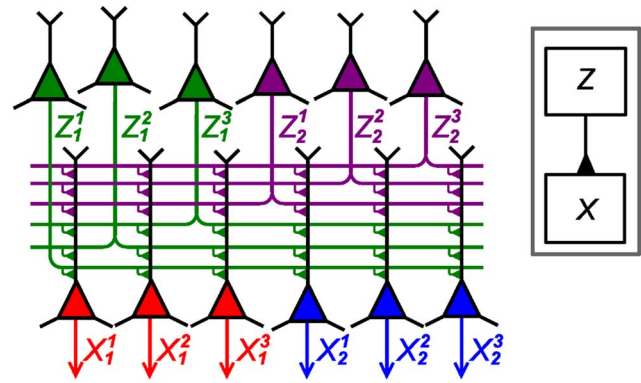


Figure 3. Computations in ENS coding in a feed forward circuit architecture. A binary random variable $v^Z(t)$ is represented in ENS coding through neurons z_j^m . The posterior $P(v^X(t))$ for a binary variable $v^X(t)$ is represented by neurons x_i^n . Each variable is represented by $N=2$ ensembles, one for each possible state (indicated by neuron color), and $M=3$ neurons per ensemble. The two layers are connected in an all-to-all manner. Arrows indicate efferent connections (i.e., outputs in ENS coding). The architecture is summarized in the inset. doi:10.1371/journal.pcbi.1003859.g003

rates that depend on the current hidden state. In the context of neuronal circuits, observations are reported through spikes of afferent neurons $y_1, \dots, y_{M_{in}}$. In particular, afferent neuron y_l is assumed to spike in a Poissonian manner with rate λ_{li} if the hidden variable assumes state i . For a prior distribution over states $P(v(t_0))$, the task is to infer the posterior $P(v(t)|\mathbf{Y}(t))$, that is, the distribution over states at time t , given the spike trains of all afferent neurons up to time t , denoted here by $\mathbf{Y}(t)$. Many laboratory tasks can be formalized as evidence integration task, including the random-dot motion task, the probabilistic inference task considered in [2] and the ambiguous target task discussed in the introduction. We construct in the following a circuit of spiking neurons that approximates optimal evidence integration by performing particle filtering in ENS coding under the assumption that $v(t)$ is constant. We will then evaluate its behavior against experimental data in computer simulations. This circuit will be extended in the subsequent sections to perform particle filtering for tasks classes C and D.

It is well-known that the evidence integration problem can be solved efficiently through a set of coupled differential equations

$$\frac{d}{dt} p_i(t) = p_i(t) \left[\sum_l (\lambda_{li} - 1) \hat{y}_l(t) - \lambda_i \right], \quad (7)$$

where $\hat{y}_l(t)$ is the spike train of afferent neuron l formalized as a sum of Dirac delta pulses at spike times (see *Methods*), and $\lambda_i = \sum_l \lambda_{li}$ [13,14]. The inferred probabilities can be obtained by normalization $P(v(t)=i) = \frac{p_i(t)}{\sum_j p_j(t)}$ (for $i=1, \dots, N$). λ_i induces a constant decrease of p_i such that hidden states that give rise to many observations are punished if no observations are encountered. Is the dynamics (7) compatible with ENS coding, assuming that p_i is estimated from the spiking activity of an ensemble? Four potential difficulties arise. First, the afferent neurons impact eq. (7) via point events and not via EPSPs ($\hat{y}_l(t)$ instead of $y_l(t)$). Second, the deterministic dynamics (7) need to be implemented via particles in the ENS code. Third, the summed evidence needs to be multiplied with the current value of p_i . And finally, to avoid an

exponential blow-up of the p_i 's, their values need to be normalized. We discuss these four issues in the following.

Evidence can be provided through EPSPs. It turns out that the first difficulty can be resolved in a convenient manner. The set of differential equations (7) can be transformed to

$$\frac{d}{dt}p_i(t) = p_i(t) \left(\sum_l w_{il}^{\text{ev}} y_l(t) + \lambda_{\max} - \lambda_i + g(t) \right), \quad (8)$$

with weights $w_{il}^{\text{ev}} = \log \lambda_{li} + \alpha$ for an arbitrary constant α , $\lambda_{\max} = \max_j \{\lambda_j\}$, and an arbitrary function g . Integration of EPSPs weighted by $\log \lambda_{li}$ leads to exactly the same result as integration of eq. (7) after the EPSPs have fully been integrated, even if they are temporally overlapping, see *Methods*. The constant α can be used to shift w_{il}^{ev} to positive values. This constant, λ_{\max} , and the function g are integrated by all p_i 's giving rise to a scaling that cancels in the normalization.

Particle-based implementation of the filtering equations. We now discuss how eq. (8) is approximated in ENS coding. While eq. (8) is a deterministic differential equation, the dynamics of the circuit is stochastic due to sampling noise. We construct a circuit that approximates the desired changes in its expected probability masses. The validity of this approach will later be ascertained through various computer simulations. The belief about the hidden state of the random variable $v(t)$ is in general shaped by two components. First, the assumed internal dynamics of the random variable, and second by novel evidence about the state. Consider a circuit that consists of two layers \mathcal{L}_{dyn} (the dynamics layer) and \mathcal{L}_{ev} (the evidence layer) with neural ensembles $X_{\text{dyn},1}, \dots, X_{\text{dyn},n}$ and $Z_{\text{ev},1}, \dots, Z_{\text{ev},n}$ respectively, see Figure 4A. The dynamics layer \mathcal{L}_{dyn} implements changes of the represented distribution due to the internal dynamics of the random variable. The evidence layer \mathcal{L}_{ev} implements changes due to incoming evidence. Since in task class B, the random variable is assumed to be static, no temporal changes of the random variable are expected, and hence the excitatory weights w_{ij}^{dyn} from \mathcal{L}_{ev} to \mathcal{L}_{dyn} are set such that \mathcal{L}_{dyn} copies the distribution represented by \mathcal{L}_{ev} , as discussed in *Task class A*. For task classes C and D, different weights will be used such that layer \mathcal{L}_{dyn} predicts dynamics changes of the hidden variable.

The evidence layer \mathcal{L}_{ev} receives input from \mathcal{L}_{dyn} and evidence from afferent neurons y_l . Our goal is that its probability masses integrate evidence in the representation of $p_i(t)$ in \mathcal{L}_{dyn} , such that

$$\begin{aligned} \frac{d}{dt} \langle Z_{\text{ev},i}(t) \rangle &\approx \langle X_{\text{dyn},i}(t) \rangle \left(\sum_l w_{il}^{\text{ev}} y_l(t) + \lambda_{\max} - \lambda_i + g(t) \right) \\ &\approx \langle Z_{\text{ev},i}(t) \rangle \left(\sum_l w_{il}^{\text{ev}} y_l(t) + \lambda_{\max} - \lambda_i + g(t) \right), \end{aligned} \quad (9)$$

where the latter approximation applies due to the copy operation of layer \mathcal{L}_{dyn} . This equation resembles eq. (8) where the $Z_{\text{ev},i}(t)$'s represent the $p_i(t)$'s. We show in *Methods* that such changes are obtained if the membrane potentials of the neurons in \mathcal{L}_{ev} are set to

$$u_{\text{ev},i}^m(t) = \frac{1}{M} X_{\text{dyn},i}(t) \left(\sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} + g(t) \right), \quad (10)$$

where $b_i^{\text{ev}} = \frac{1}{\tau} + \lambda_{\max} - \lambda_i$ are positive biases.

Multiplication through gating of activity. We see that neurons need to compute a multiplication between the current probability mass for state i and the summed evidence. In order to implement similar equations in a neuronal-like manner, logarithmic dendritic nonlinearities or multiplicative synaptic interactions have been postulated in a number of studies, see e.g. [14,15]. In ENS coding however, the population response in the evidence layer is the sum of the responses of individual neurons. This linearity allows us to base the membrane potential of an individual neuron $z_{\text{ev},i}^m$ on a small number of particles rather than on the whole set of particles summarized in $X_{\text{dyn},i}(t)$, as long as each particle is used exactly once in the computation. Hence, the same behavior is obtained on the population level if instead of membrane potentials (10), membrane potentials are given by

$$u_{\text{ev},i}^m(t) = x_{\text{dyn},i}^m(t) \left(\sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} + g(t) \right), \quad (11)$$

see *Methods* for a detailed derivation. If spiking of individual neurons $x_{\text{dyn},i}^m$ is sparse, i.e., if the ensemble size is large compared to the estimation sample size, $x_{\text{dyn},i}^m(t)$ nearly always takes on the values 0 or 1. In this case, it suffices that the activity of neuron $z_{\text{ev},i}^m$ is *gated* by neuron $x_{\text{dyn},i}^m$, i.e., the neuron $z_{\text{ev},i}^m$ is able to produce spikes only if $x_{\text{dyn},i}^m$ was recently active. In summary, under the assumption of sparse activity (which can always be accomplished by a suitable choice of parameters), one can replace the multiplication of two analog variables by gating of activity in ENS coding. This multiplication strategy generalizes the one proposed in the context of stochastic computation to ensemble representations [16,17].

Such gating could be accomplished in cortical networks in various ways. One possibility is synaptic gating [18,19] where inputs can be gated by either suppression or facilitation of specific synaptic activity. Another possibility is disinhibition. Disinhibitory circuits provide pyramidal cells with the ability to release other neurons from strong inhibitory currents [20]. We choose in this article disinhibition as the gating mechanism, although no specific mechanism can be favored on the basis of the experimental literature. A small circuit with disinhibition $I^{\text{dis}}(x_1(t), x_2(t))$ is shown in Figure 4B. Here, two pyramidal cells excite an interneuron which inhibits the inhibitory drive to some neuron z_i . Functionally, z_i is released from strong baseline inhibition if one of the neurons x_1, x_2 was recently active (see *Methods* for a formal definition). Using disinhibition, the membrane potential of the neuron can be written as

$$u_{\text{ev},i}^m(t) = \sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} - I^{\text{dis}}(x_{\text{dyn},i}^m(t)) + g(t), \quad (12)$$

where $I^{\text{dis}}(x_{\text{dyn},i}^m(t))$ ensures that the firing rate $\rho_{\text{ev},i}^m(t) = [u_{\text{ev},i}^m(t)]^+$ is nonzero only if neuron $x_{\text{dyn},i}^m$ did spike within the last 20 ms.

Stabilization of firing rates through lateral inhibition. Lateral inhibition is generally assumed to stabilize the activity of excitatory populations [21–23]. A group of pyramidal cells inhibit each other laterally by projecting to a group of inhibitory neurons which in turn inhibit that ensemble, see Figure 4C. The key observation that enables us to use lateral inhibition to stabilize circuit activity is that one has freedom to choose $g(t)$ in eq. (12) as long as it is identical in all ensembles. We model lateral inhibition $I^{\text{lat}}(t)$ that depends on the recent firing activity in \mathcal{L}_{ev} such that inhibitory activity increases if the

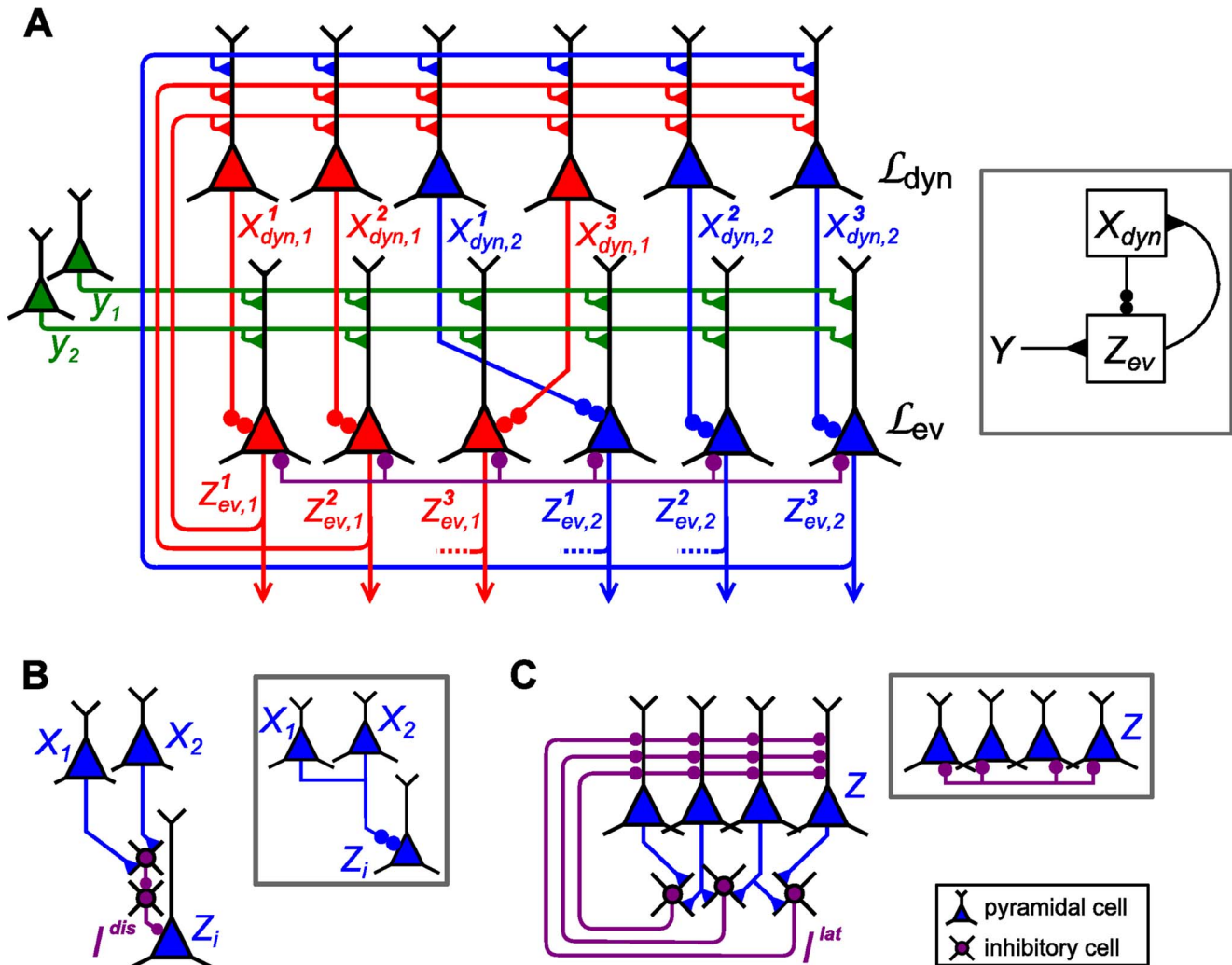


Figure 4. Particle filter circuit architecture for task classes B and C. **A**) Circuit with $N=2$ ensembles (indicated by red and blue neurons respectively) and $M=3$ neurons per ensemble. Neurons in layer \mathcal{L}_{ev} receive synaptic connections from neurons in layer \mathcal{L}_{dyn} and update the represented distribution according to evidence input from afferent neurons (green). Lateral inhibition (magenta; see panel C) stabilizes activity in this layer. Neurons project back to layer \mathcal{L}_{dyn} . For task class B (evidence integration; static random variable $v(t)$), only connections between neurons that code for the same hidden state are necessary and layer \mathcal{L}_{dyn} simply copies the distribution represented by layer \mathcal{L}_{ev} , see *Task class A* and Figure 3 (in contrast to Figure 3, the copying ensembles X_{dyn} are plotted above ensembles Z_{ev} in order to avoid a cluttered diagram). For task class C (Bayesian filtering; random variable $v(t)$ with time-independent dynamics), \mathcal{L}_{dyn} implements changes of the represented distribution due to the dynamics of the random variable and \mathcal{L}_{ev} is potentially fully connected to \mathcal{L}_{dyn} . Neurons in layer \mathcal{L}_{dyn} disinhibit neurons in layer \mathcal{L}_{ev} (double-dot connections; see panel B). Disinhibition and lateral inhibition is indicated by shortcuts as defined in B, C. Arrows indicate efferent connections. A schematic overview of the circuit is shown in the inset. **B**) Disinhibition I^{dis} : neurons x_1, x_2 excite an interneuron (purple) which inhibits the inhibitory drive to some neuron z_i . As a graphical shortcut, we draw such disinhibitory influence as a connection with two circles (inset) **C**) Lateral inhibition: Pyramidal cells (blue) excite a pool of inhibitory neurons (magenta) which feed back common inhibition I^{lat} . The graphical shortcut for lateral inhibition is shown in the inset.

doi:10.1371/journal.pcbi.1003859.g004

estimation sample size is above the desired value L and choose $g(t) = -I^{lat}(t)$, see *Methods* for details and a brief discussion. This concludes the construction of a particle filtering circuit in ENS coding for task class B. The circuit architecture is depicted in Figure 4A. A summary of circuit equations can be found in Table 1.

We tested how well a circuit consisting of 2000 neurons per state and an estimation sample size of 400 can approximate the true posterior in a simple evidence integration setup. The task was to compute the posterior distribution for a random variable $v(t)$ with two hidden states and two observable variables $y_1(t), y_2(t)$, see Figure 5A. The schematic circuit diagram is shown in panel B.

The two evidence neurons y_1, y_2 spiked at times 20 ms and 25 ms respectively, see panel C. Figure 5D depicts the rate dynamics in layer \mathcal{L}_{ev} for one example trial. Novel evidence transiently increases the firing rate in the layer, which is in turn restored by lateral inhibition. The response of ensemble $Z_{ev,1}$ that represents state 1 undergoes a transient increase that is counteracted by inhibition until it stabilizes at an enhanced sustained level. This behavior is reminiscent of the typical response of cortical pyramidal cells to sensory input. Figure 5E shows the temporal evolution of the encoded posterior probability $\hat{P}(v(t)=1|\mathbf{Y}(t))$ in comparison to the true posterior $P(v(t)=1|\mathbf{Y}(t))$. The true posterior is approximated very well after a delay of about 20 ms,

Table 1. Particle filter circuit equations for task classes B and C.

Layer	Ensembles	Neurons	Membrane voltage and parameters
\mathcal{L}_{dyn}	$X_{\text{dyn},i}$	$x_{\text{dyn},i}^n$	$u_{\text{dyn},i}^n(t) = \sum_{j,m} w_{ij}^{\text{dyn}} z_{\text{ev},j}^m(t)$
\mathcal{L}_{dyn}	$X_{\text{dyn},i}$	$x_{\text{dyn},i}^n$	$w_{ij}^{\text{dyn}} = \frac{q_{ij}}{M}$ for $i \neq j$, $w_{ii}^{\text{dyn}} = \frac{1}{\tau M} - \frac{1}{M} \sum_{j \neq i} q_{ij}$ for all i .
\mathcal{L}_{ev}	$Z_{\text{ev},i}$	$z_{\text{ev},i}^n$	$u_{\text{ev},i}^n(t) = \sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} - I^{\text{dis}}(x_{\text{dyn},i}^n(t)) - I^{\text{lat}}(t)$
\mathcal{L}_{ev}	$Z_{\text{ev},i}$	$z_{\text{ev},i}^n$	$w_{il}^{\text{ev}} = \log \lambda_{li} + \alpha$, $b_i^{\text{ev}} = \frac{1}{\tau} + \lambda_{\text{max}} - \lambda_i$

Here we have defined $\lambda_{ij} \equiv \sum_l \lambda_{li}$ and $\lambda_{\text{max}} = \max_j \{\lambda_j\}$. $I^{\text{lat}}(t)$ denotes lateral inhibition and I^{dis} disinhibition. α is an arbitrary constant. In task class B (evidence integration), $q_{ij} = 0$ for $i \neq j$, leading to $w_{ij}^{\text{dyn}} = 0$ for $i \neq j$ and $w_{ii}^{\text{dyn}} = \frac{1}{\tau M}$.
doi:10.1371/journal.pcbi.1003859.t001

which is the time needed to integrate the EPSPs from evidence neurons. We simulated 100 trials where in each trial, prior probabilities for the states and observation likelihoods were drawn randomly such that the posterior $P(v(t) = 1 | \mathbf{Y}(t))$ at time $t = 45$ ms assumed values between 0 and 1 (see *Methods*). The estimate of the circuit at the end of the second EPSP (i.e., at time $t = 45$ ms) is shown in comparison to the true posterior in Figure 5F.

Gating of activity for multiplication can be used for many types of multiplicative operations on probability distributions. In *Text S2*, we discuss its application to cue combination, an operation that has been considered for example in [24].

Comparison to experimental results

We performed computer simulations in order to compare the behavior of the model to various experimental studies on tasks that are examples for task class B.

The ambiguous target task. The ambiguous target task studied in [3] was already discussed above, see also Figure 1A. In our model of the decision making process, the hidden state of a random variable $v(t)$ was estimated through evidence integration. Each of the 16 hidden states corresponded to a tuple (d, c) , where d denotes one of eight possible directions of movement, and c denotes the color of the color cue. In other words, such a state represented the color of the color cue and the movement direction that leads to reward, see Figure 6Aa. Possible observations were the fixation cross, the spatial cues at 8 positions in two colors, and the two color cues. Each of the 19 possible stimuli was coded by 20 afferent neuron that fired at a baseline rate of 0.1 Hz. When a stimulus was present, the corresponding neurons spiked in a Poissonian manner with a rate of 5 Hz.

We simulated a particle filter circuit to compute the belief about the state of the random variable with 1000 neurons per hidden state and an estimation sample size of 400. An action readout layer as described in *Task class A* was added that received connections from \mathcal{L}_{ev} in a feedforward manner, see Figure 6B. This layer computed the current belief over rewarded actions independently from the color of the color cue (i.e., it marginalized over color), see *Methods* for details.

The spiking activity of afferent neurons that provided evidence for one example simulation run is shown in Figure 6Ca. Simulated neural activities from the readout layer are shown in Figure 6Cb, see also Figure 1C. After the spatial cue was presented, the two consistent ensembles increased their activity. Due to competition between these ensembles, neurons fired at a medium rate. After the color cue was shown, only the ensemble consistent with both the spatial and the color cue remained active. These neurons increased their firing rate since the competing action became improbable and the winning ensemble was uncompleted. This

behavior has been observed in PMd [3], see Figure 1B. The action readout layer is not needed to reproduce this behavior, since neurons in the particle filter circuit exhibit similar behavior.

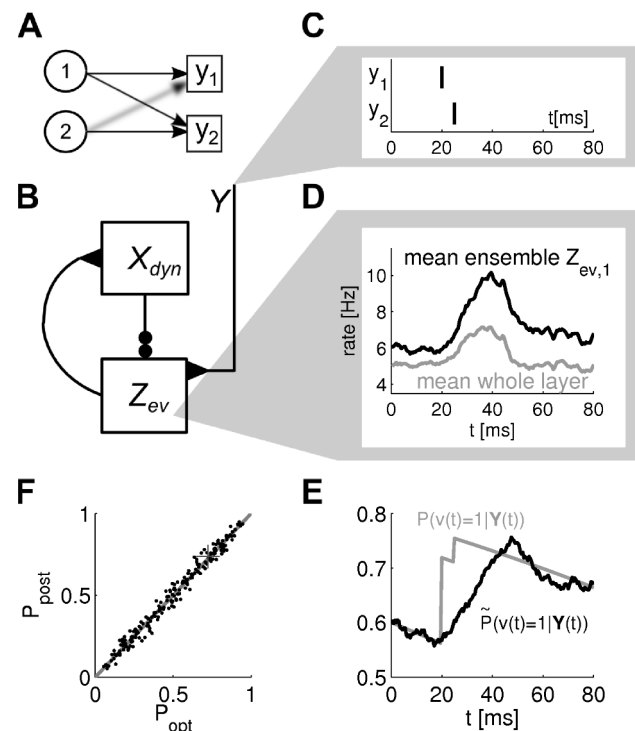


Figure 5. Evidence integration through particle filtering in ENS coding. **A)** The state of a binary random variable $v(t)$ that gives rise to two possible observations $y_1(t), y_2(t)$ is estimated. Both observations occur more frequently in state 1 (indicated by sharpness of arrows). **B)** Estimation is performed by a particle filtering circuit with evidence input Y (X_{dyn} : dynamics layer ensembles; Z_{ev} : evidence layer ensembles). **C)** An evidence spike is observed at times 20 ms and 25 ms in evidence neuron y_1 and y_2 respectively. **D)** Example for the rate dynamics in layer \mathcal{L}_{ev} . Ensemble rate for ensemble $Z_{\text{ev},1}$ (black) and whole layer \mathcal{L}_{ev} (gray). The input leads to a transient increase in the ensemble rate. Inhibition recovers baseline activity. The ensemble rate for state 1 undergoes a transient and a sustained activity increase. **E)** Temporal evolution of estimated posterior probability $\hat{P}(v(t) = 1 | \mathbf{Y}(t))$ for state 1 (black) in comparison to true posterior $P(v(t) = 1 | \mathbf{Y}(t))$ (gray) for this example run. **F)** Posterior probability at $t = 45$ ms (P_{post}) for state 1 of the circuit in comparison to true posterior at this time (P_{opt}). Each dot represents one out of 100 runs with prior probabilities and observation likelihoods drawn independently in each run (see *Methods*). The results of the example run from panels A–E is indicated by a cross.
doi:10.1371/journal.pcbi.1003859.g005

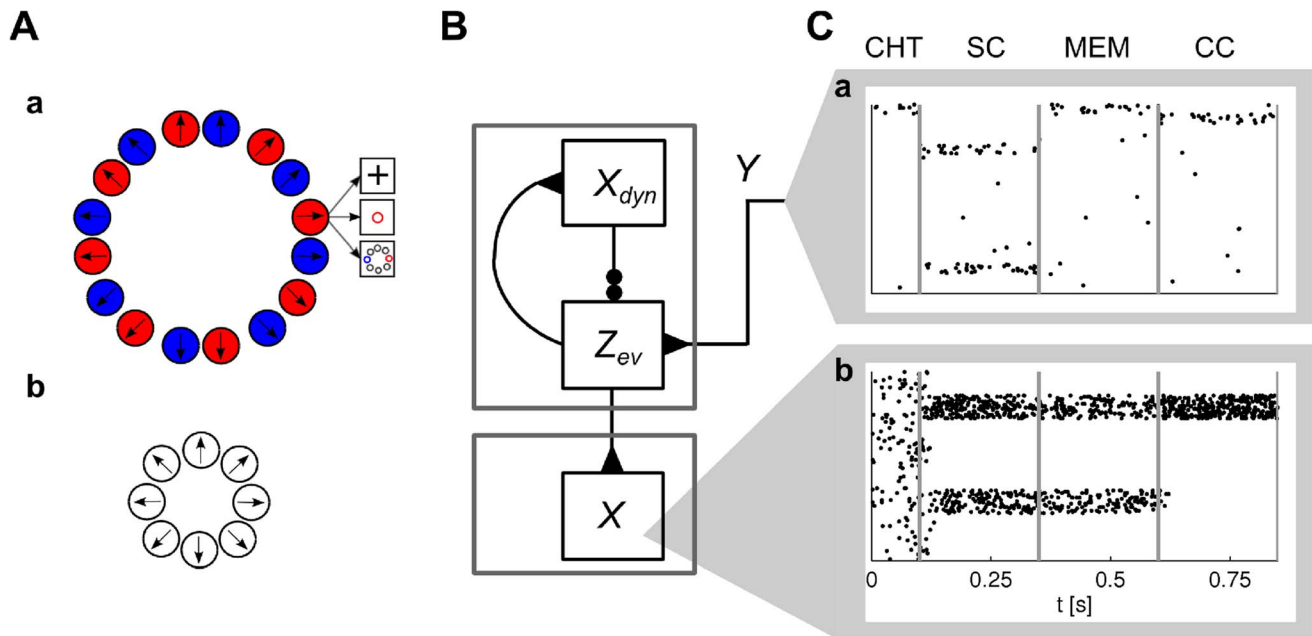


Figure 6. Particle filtering in ENS coding for the ambiguous target task. **A)** Represented random variables. **Aa)** Evidence integration is performed for a random variable with 16 hidden states corresponding to direction-color pairs. Values of the random variable are depicted as circles. Observations accessible to the monkey in one example state are shown as boxes. **Ab)** The action readout layer infers a color-independent random variable by marginalization over color in each direction. **B)** Circuit structure. The circuit on the top approximates evidence integration through particle filtering (top gray box; X_{dyn} : dynamics layer ensembles; Z_{ev} : evidence layer ensembles) on the random variable indicated in panel (Aa). An action readout layer (bottom gray box; ensembles X) receives feed-forward projections from the particle filter circuit. **C)** Spike rasters from simulations for afferent neurons (Ca) and neurons in the action readout layer (Cb). Each line corresponds to the output of one neuron. Afferent neurons are ordered by feature selectivity (e.g., top neurons code the presence of the fixation cross). Action readout neurons are ordered by preferred movement direction. See also Figure 1.

doi:10.1371/journal.pcbi.1003859.g006

However, it was reported that most neurons in PMd were not color selective [3]. In our model, neurons of the particle filter circuit are color selective since states are defined according to direction-color pairs. It is clear that color-related information has to be integrated with movement-related information and memorized during the memory epoch in order to solve the task. The experimental results suggest that this integration is not implemented in PMd but rather in upstream circuits. PMd could then act as a motor readout.

Applications of the model to various other experimental tasks can be found in supporting texts. Action-predictive activity in macaque motor cortex is also modulated by the expected value of the action. This was demonstrated for example in [25]. An application of our model to this scenario is described in *Text S3*. Furthermore, we show in *Text S4* that the model is consistent with features of neuronal activity during random-dot motion tasks [1,26,27]. In *Text S5* it is shown that the model can also explain neuronal activity in area LIP during a probabilistic reasoning task [2].

Task class C: Bayesian filtering

Evidence integration cannot take temporal changes of the hidden variable into account. Knowledge about temporal changes can be exploited by Bayesian filtering, which is an extension of evidence integration. Here, we assume that the dynamics of $v(t)$ are constant during the filtering process. The more general case when the dynamics may change is discussed below in *Task class D*. Formally, the Bayesian filtering problem considered here is to estimate the posterior distribution $P(v(t)|\mathbf{Y}(t))$ over the states of a random variable $v(t)$ that represents the hidden state of a random

process which is only indirectly observable via stochastic point-event observations $\mathbf{Y}(t)$. In particular, state changes are assumed to be Markovian with *transition rates* q_{ij} for each pair of distinct states $i \neq j$. Transition rate q_{ij} defines the rate of transition from state i to state j , i.e., the probability that a transition occurs to state j in some small time interval if the current state is i , thus defining a continuous time Markov chain, see *Methods* for a more formal description.

Note that evidence integration is a special case of Bayesian filtering with the assumption of no state transitions, i.e., $q_{ij} = 0$ for all $i \neq j$. In the following, we show that the particle filtering circuit for task class B constructed above can easily be extended to this generalization. The Bayesian filtering problem can be solved efficiently through a set of coupled differential equations

$$\frac{d}{dt} p_i(t) = p_i(t) \left[\sum_l (\lambda_{li} - 1) \hat{y}_l(t) - \lambda_i \right] + \sum_{j \neq i} q_{ji} p_j(t) - p_i(t) \sum_{j \neq i} q_{ij}, \quad (13)$$

where again the inferred probabilities are obtained by normalization [13,14]. Note that the first term in eq. (13) is identical to eq. (7), since the optimal solution for evidence integration, eq. (7), is the special case of eq. (13) for vanishing transition rates. This term is taken care of in layer \mathcal{L}_{ev} of the particle filtering circuit for task class B. In this circuit, \mathcal{L}_{dyn} simply copies the distribution given by \mathcal{L}_{ev} . We modify the connections from \mathcal{L}_{ev} to \mathcal{L}_{dyn} such that \mathcal{L}_{dyn} instead provides the changes in probability masses needed for the second and third term, i.e., it predicts changes based on the

assumed dynamics of the random variable. \mathcal{L}_{dyn} approximates the desired changes of probability masses if the membrane potentials are given by

$$u_{\text{dyn},i}^m(t) = \sum_{j,n} w_{ij}^{\text{dyn}} z_{\text{ev},j}^n(t), \quad (14)$$

with synaptic efficacies $w_{ij}^{\text{dyn}} = \frac{q_{ji}}{M}$ for $i \neq j$ and $w_{ii}^{\text{dyn}} = \frac{1}{\tau M} - \frac{1}{M} \sum_{j \neq i} q_{ij}$ for all i , see *Methods*. An overview of the circuit equations and parameters is given in Table 1, see also Figure 4A.

We tested the ability of a particle filter circuit consisting of 2000 neurons per state and an estimation sample size of 400 to track the temporal evolution of a binary random variable where state 1 transitions to state 2 with some transition rate q_{12} , see Figure 7A. The dynamics of the circuit and the estimated probability for an example simulation run are shown in panels C,D. We simulated 100 trials, where in each trial the transition rate q_{12} was drawn uniformly in $[0, 30]$ Hz and initial probabilities were drawn uniformly in $[0.1, 0.9]$. The estimate of the posterior at time $t = 50$ ms is shown in comparison to the true posterior in Figure 7E. See *Methods* for details on this simulation.

Particle filtering in a generic setup for task class C. We performed further computer simulations in order to test the

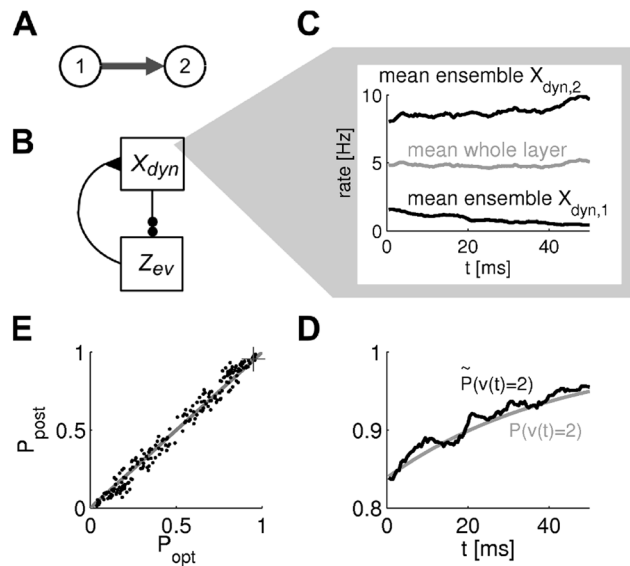


Figure 7. Tracking of dynamics in ENS coding. **A)** The state of a binary random variable $v(t)$ is estimated where state 1 transitions to state 2 with some transition rate q_{12} . **B)** Estimation is performed by a particle filter circuit without evidence input (X_{dyn} : dynamics layer ensembles; Z_{ev} : evidence layer ensembles). **C)** Example for the rate dynamics in layer \mathcal{L}_{dyn} . Ensemble rate for ensembles $X_{\text{dyn},1}, X_{\text{dyn},2}$ (black) and whole layer \mathcal{L}_{dyn} (gray). While rates in ensembles change due to the prediction of a transition, inhibition keeps the overall firing rate in the layer approximately constant. **D)** Temporal evolution of estimated posterior probability $\tilde{P}(v(t)=2)$ for state 2 (black) and true posterior $P(v(t)=2)$ (gray) for this example run. **E)** Circuit estimates of posterior probabilities at time $t = 50$ ms (P_{post}) in comparison to true posteriors at this time (P_{opt}). Shown are 100 runs (dots) with prior probability for state 1 and transition rate drawn from uniform distributions in $[0.1, 0.9]$ and $[0, 30]$ Hz respectively in each run. The result of the example run from panels C-D is indicated by a cross. doi:10.1371/journal.pcbi.1003859.g007

performance of the model in a generic setup. In this setup, a random variable $v(t)$ evolved according to a continuous time Markov chain with five states. In this Markov chain, state 1 transitions to states 2 or 3 which themselves transition to states 4 and 5 respectively. From states 4 and 5, a transition to state 1 is possible, see Figure 8Aa (bottom). Transition rates for all possible transitions were set to 1Hz. Information about the actual state was conveyed by 35 afferent neurons with state-dependent rates defined by Gaussians as shown in Figure 8Aa (top). Note that states 2 and 3 gave rise to very similar observations. Thus, many observations have to be integrated before these states can be distinguished. This makes inference over the current state hard if the full distribution over state probabilities is not communicated over time.

Results from simulations of a particle filter circuit with 2000 neurons per state and an estimation sample size of 400 are shown in Figure 8Ac–Af, see *Methods* for details. Figure 8Af shows a comparison of the model performance (“model”) with the optimal Bayesian filtering (“opt”) and the optimal model that does not take temporal information into account (the Bayes estimate based on the latest observation; “inp”). The performance of the model was very close to optimal and much better than the non-temporal Bayes estimate. We furthermore tested the robustness of the network to variations of synaptic efficacies w_{ij}^{dyn} that determine the assumed transition rates of the random variable. In a control experiment, each individual synaptic weight from ensemble $Z_{\text{ev},j}$ to ensemble $X_{\text{dyn},i}$ was drawn from a log-normal distribution with mean w_{ij}^{dyn} and standard deviation $w_{ij}^{\text{dyn}}/2$. The resulting weights assumed values that were up to 10 times larger than the mean. Network performance with jittered efficacies was indistinguishable from the performance of the homogeneous network (Figure 8Af; “jit”). This robustness stems from two features of ENS-coding. First, as network belief is represented by ensemble rates, it is invariant to firing rate variations of individual neurons as long as the ensemble rate is preserved. Second, network computations are generally based on averages over ensemble activities, see eq. (14). Therefore variations in synaptic efficacies do not influence the result as long as the mean ensemble-to-ensemble weights are preserved. Note that this even holds for nonlinear link-functions, eq. (3).

Task class D: Context-dependent Bayesian filtering

In many important situations, the dynamics of a random variable changes in different contexts. Context cannot simply be formulated as a type of observation since observations just influence the probabilities of states for the given transition rates and not the dynamics themselves. Consider for example the estimation of the current body position in space. Here, an action such as forward movement can be considered as context since it increases the transition rates from any position to positions ahead. Thus, it changes the transition rates and not just the probability of a particular position. For N_C possible contexts, consider a function $c(t)$ with range $\{1, \dots, N_C\}$ that indicates the context at time t . We define a context-dependent Markov chain as a Markov chain with state transition rates $q_{ij}^1, \dots, q_{ij}^{N_C}$ for each pair of distinct states i, j . At each time t , the chain evolves according to $q_{ij}^{c(t)}$. Context-dependent Bayesian filtering determines the distribution over the current state for the given observations and context. Note that Bayesian filtering used in task class C is a special case with just a single context. Also note that we use the term *context* here with the specific meaning of additional information about the dynamics of the random variable.

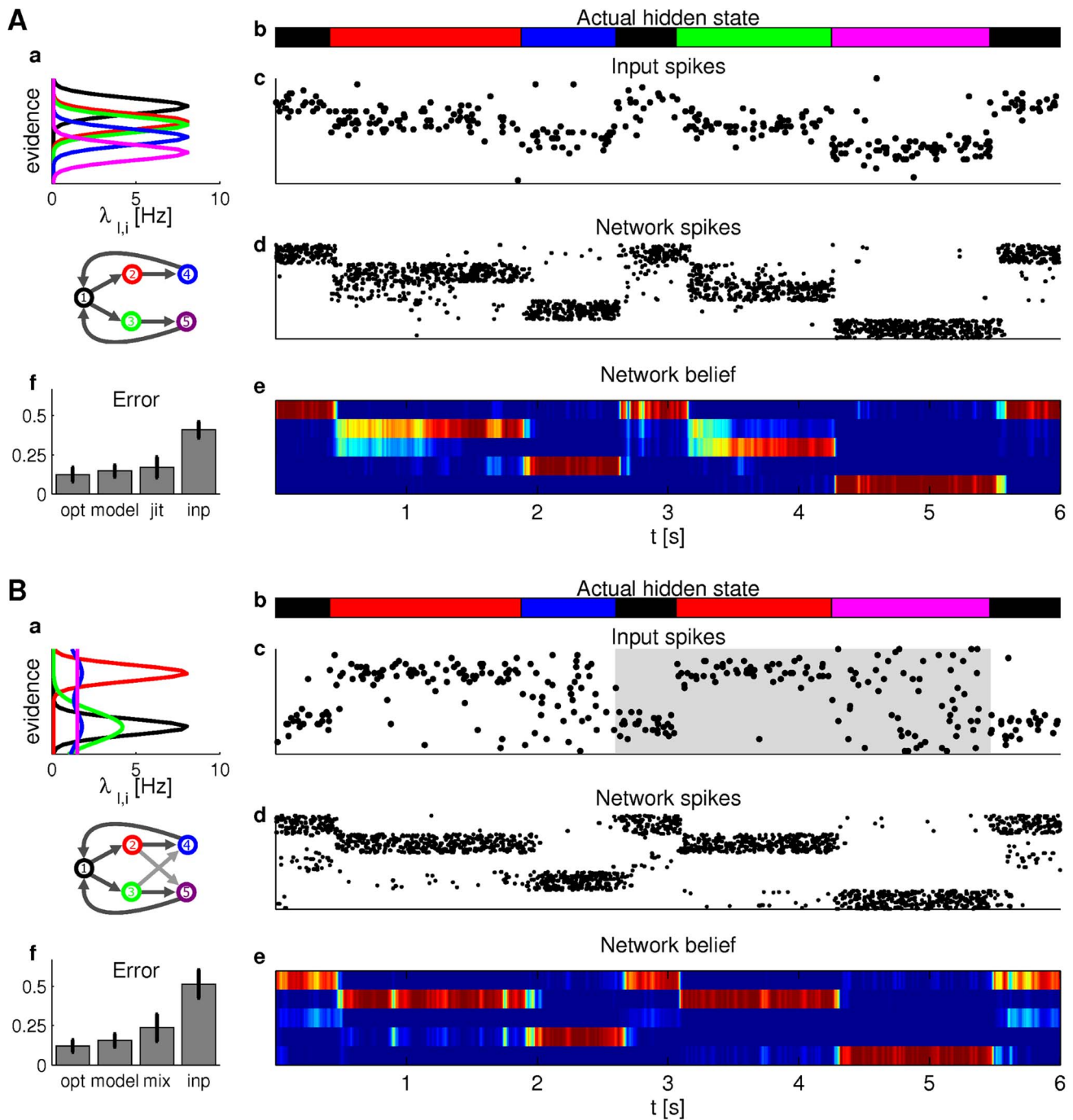


Figure 8. Particle filtering for task class C (A) and task class D (B) in the ENS code. Aa State diagram of the Markov chain for the dynamics of the hidden random variable (bottom) and state-dependent firing rates of afferent neurons (top). Colors indicate the value of the hidden state. **Ab** Actual hidden state over time is indicated by color in correspondence with colors in panel Aa. **Ac** Spike trains of afferent neurons. Each line corresponds to the output of one afferent neuron ordered according to panel Aa. **Ad** Network response to the input in panel Ac. Neurons are ordered according to their preferred state from state 1 (top neurons) to 5 (bottom neurons). **Ae** Network belief (estimated posterior state probability) derived from network activity. Rows ordered by state as neurons in panel Ad. Hot color indicates high probability of the state. Note the uncertainty when state 2 or 3 is entered. **Af** Summary of network performance (“model”; fraction of incorrect state estimates) in comparison with the optimal Bayesian filter (“opt”), a network with jittered synaptic efficacies (“jit”), and the optimal decision based on the most recent observation only (“inp”). Bars are means and errorbars STDs over 20 state and observation sequences (12 seconds each). **B** Particle filtering for task class D. **Ba** As panel Aa but with context. Dark gray arrows in the state diagram indicate transitions in context A. In context B, the transitions from states 2 to 4 and 3 to 5 are interchanged (light gray arrows). **Bb** As panel Ab. Background shading indicates context (context A: white; context B: gray). **Bc–Be** Actual hidden state, input spikes, network spikes, and network belief; see panels Ac–Ae. **Bf** Summary of network performance. “opt” shows performance of the optimal context-dependent Bayesian filter and “mix” a Bayesian filter where the transition rates are the mean rates over contexts A and B. The spiking network performs significantly better than the mixed Bayesian filter (paired t-test, $p < 0.001$). doi:10.1371/journal.pcbi.1003859.g008

We encode the current context by ensembles C_1, \dots, C_{N_C} that provide contextual feedback to the circuit. Each context ensemble C_i consists of M_C context neurons $c_i^1, \dots, c_i^{M_C}$. At time t , only the context ensemble $C_{c(t)}$ is active. The circuit architecture for particle filtering (Figure 4A) is extended as shown in Figure 9 to perform particle filtering for a hidden random variables $v(t)$ with context-dependent dynamics. Layer \mathcal{L}_{dyn} consists of N_C representations of the random variable, one for each context. Hence, for each context k there are N ensembles $X_{1,k}, \dots, X_{N,k}$ in this layer. In addition to excitatory synaptic connections originating from \mathcal{L}_{ev} , these neurons are disinhibited by context neurons from the corresponding context ensembles. The disinhibitory effect of context is consistent with experimental data about neocortical circuits. There, context information from other cortical areas is believed to be provided through feedback connections. Those connections have abundant terminals in neocortical layer 1 where they recruit disinhibitory circuits [28,29]. Formally, the membrane potential of $x_{\text{dyn},i,k}^m$, the m -th neuron in ensemble $X_{i,k}$, is given by

$$u_{i,k}^m(t) = \sum_{j,n} w_{ij}^{\text{dyn},k} z_{\text{ev},j}^n(t) - I^{\text{dis}}(c_k^1(t), \dots, c_k^{M_C}(t)), \quad (15)$$

where $z_{\text{ev},j}^n(t)$ denotes the EPSP-filtered spike train of a neuron in ensemble $Z_{\text{ev},j}$ of \mathcal{L}_{ev} and $w_{ij}^{\text{dyn},k}$ denotes the efficacy of the connecting synapse. These efficacies are set proportionally to the transition rates in the corresponding context, see Table 2. The membrane potential of a neuron in \mathcal{L}_{ev} is similar to the non-contextual case with the difference that each neuron is disinhibited by neurons that code for the same state in various contexts

$$u_{\text{ev},i}^m(t) = \sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} - I^{\text{dis}}(x_{\text{dyn},i,1}^m(t), \dots, x_{\text{dyn},i,N_C}^m(t)) - I^{\text{lat}}(t). \quad (16)$$

Due to disinhibition of layer \mathcal{L}_{dyn} by context ensembles, only those ensembles in the layer which correspond to the current context are active. The active neurons in this layer disinhibit neurons in \mathcal{L}_{ev} in the same way as in the non-contextual case. Thus, in each individual context, a subcircuit is recruited that consists of the whole layer \mathcal{L}_{ev} and the ensembles for the current context in layer \mathcal{L}_{dyn} . Since the weights $w_{ij}^{\text{dyn},k}$ from \mathcal{L}_{ev} to ensembles for context k in \mathcal{L}_{dyn} implement the dynamics of the random variable in that context, the circuit approximates the correct transition dynamics of the random variable in each context. The resulting membrane potential equations are summarized together with parameters in Table 2.

Particle filtering in a generic setup for task class D. In order to test the ability of the model to perform context-dependent Bayesian filtering, we considered a random variable with context-dependent dynamics. Two possible contexts A and B were indicated by 20 context neurons. The underlying Markov chain with context-dependent transition rates is shown in Figure 8Ba (bottom). The dynamics in context A was equivalent to the one considered in the generic test for task class C (dark gray arrows). However in context B, state 2 exclusively transitioned to state 5 and state 3 exclusively to 4 (light gray arrows). Additionally, we considered a more complex observation model in this example (panel Ba, top). State-dependent firing rates were either Gaussians with varying variances, bimodal, or uniform. Note that states 4 and 5 gave rise to quite similar

observations. Hence, it is hard to distinguish these states without context. Panels Bc-Be show that the model makes good use of this context information (context indicated by gray shading in panel Bc). At time $t=2$ s, there was a transition from state 2 to state 4 in context A, and at time $t=4.25$ s, there was a transition from state 2 to state 5 in context B. In both cases, the network estimate followed immediately since the expected transitions were modulated by context information. We also tested the performance of an optimal context-dependent Bayesian filter and a Bayesian filter without context information. This filter was based on a dynamics model with mean transition rates over both contexts, which resulted in suboptimal performance, see Figure 8Bf. Implementation details are given in *Methods*.

Particle filtering in ENS coding for self-localization. Estimation of the body position in space (self-localization) is an essential ingredient of autonomous agents. In robotics, particle filtering is one of the most successful techniques for self-localization [6]. Here, we demonstrate that particle filtering in ENS coding can be used for self-localization in environments with ambiguous evidence.

Every state of the considered hidden random variable corresponded to some position in the environment and transitions were possible between spatially adjacent states. The movement of the agent provided context for the particle filter such that movement in a particular direction enhanced transitions that point to that direction (Figure 10A). Sensory cues provided partial information about the current position. We simulated a two-chamber maze with a small opening that connects these chambers. The southern parts of the chambers gave rise to exactly the same observations, making it impossible to distinguish them without prior information (colored circles in Figure 10A). Observations in more northern parts were different and in the very north, no observations were experienced (corresponding for example to a dark corridor). Figure 10B shows the estimate of the network for a single trajectory. The model was started with a uniform prior distribution over all positions. In the southern terrain, the left and the right chamber cannot be distinguished, and accordingly, ensembles in both areas were active, indicating possible positions ($t=300$ ms). In the more northern parts, observations disambiguate the current position, and activity in the right chamber ensembles ceased ($t=600, 900$ ms). As the upper northern region was reached, no more evidence was provided from the environment ($t=1500, 2400$ ms). Still, the network predicted movement to the right correctly thus utilizing movement information provided via context ensembles. As the right chamber was entered, the posterior was sharpened due to unambiguous sensory input. It remained single peaked even in southern parts of the chamber although these terrains produced ambiguous sensory input ($t=3300, 3800$ ms).

Task class E: Internal beliefs as context

In many tasks, the context variable $c(t)$ is not explicitly available to the animal but rather has to be estimated from noisy evidence $\mathbf{Y}(t)$ as well. For example, in many sequential tasks the current stage within the task can provide valuable context information which can be used to time actions or to decide when beliefs should be reset. In the computational framework considered here, this can be achieved by treating the context variable $c(t)$ as a random variable that is estimated by a particle filter circuit from the given evidence $\mathbf{Y}(t)$. Estimated context is then utilized for the estimation of a random variable $v(t)$ in a particle filter circuit with context.

In the tasks considered in task class D, the context was unambiguously given and all neurons of exactly one context ensemble had a firing rate larger than zero. This is not the case in

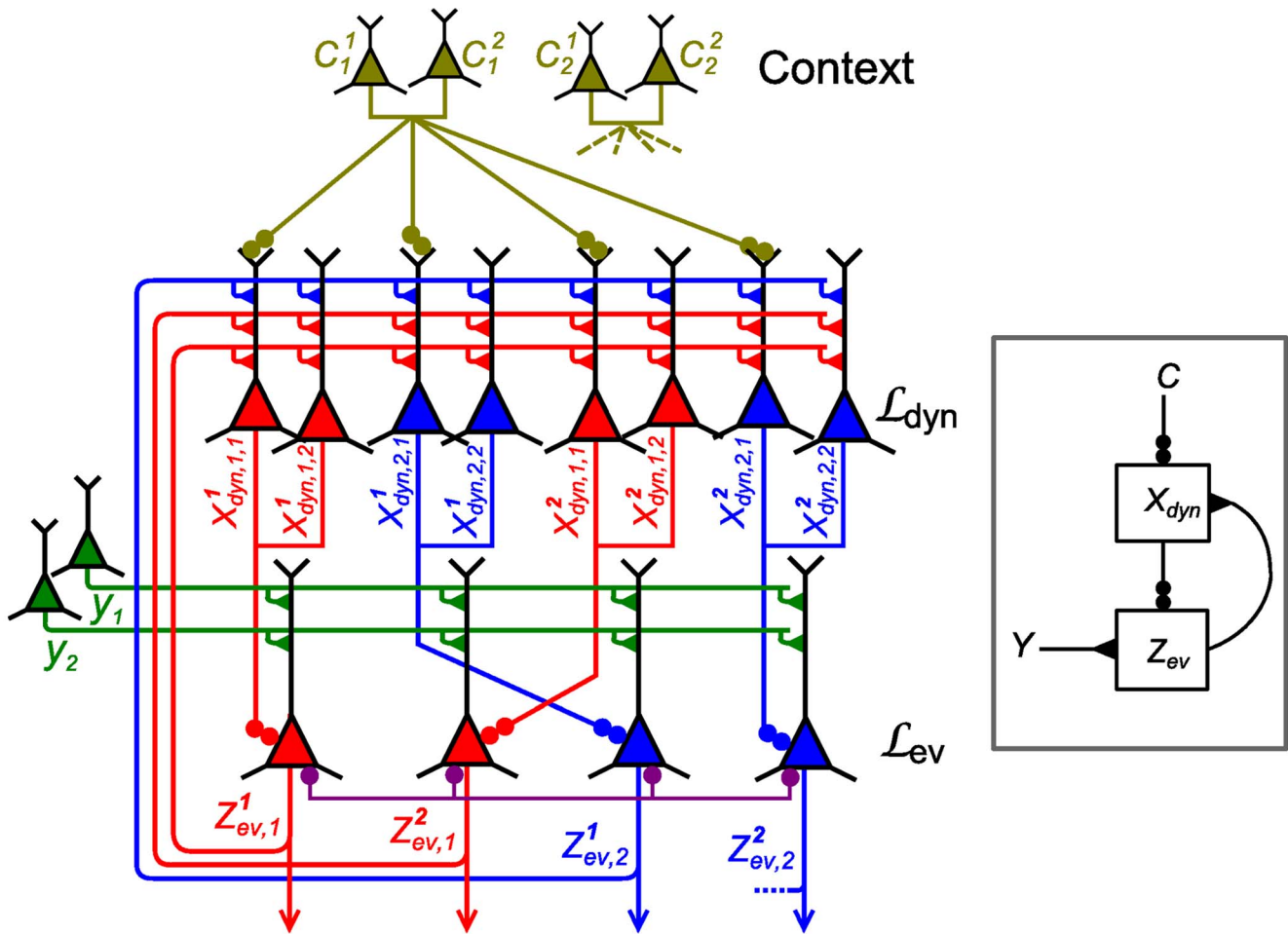


Figure 9. Particle filter circuit architecture for task class D. Extended circuit with $N = 2$ ensembles (indicated by red and blue neurons respectively) and $M = 2$ neurons per ensemble and two possible contexts. Ensembles in layer \mathcal{L}_{dyn} are duplicated for each context. These neurons receive context information via disinhibition from context neurons (yellow; only connections from context 1 shown for clarity). Disinhibition and lateral inhibition indicated by shortcuts as defined in Figure 4B, C. Arrows indicate efferent connections. A schematic overview of the circuit is shown in the inset. doi:10.1371/journal.pcbi.1003859.g009

task class E, since here context neurons encode a belief about the current context. The context-dependent particle filter should therefore deal with context in a graded manner. This is accomplished by a simple modification of how neurons in layer \mathcal{L}_{dyn} integrate context, see *Methods*. In the modified circuit, effective transition rates $q_{ij}(t)$ of the context-dependent filtering circuit are given as a linear mixture of the context-dependent transition rates $q_{ij}^k(t)$, where each contributes approximately

proportionally to the current belief $\tilde{P}(c(t) = k | \mathbf{Y}(t))$ in this context

$$q_{ij}(t) \approx \sum_k \tilde{P}(c(t) = k | \mathbf{Y}(t)) q_{ij}^k. \quad (17)$$

To demonstrate the viability of this approach, we reconsidered the ambiguous target task.

Table 2. Particle filter circuit equations for task class D.

Layer	Neurons	Membrane voltage and parameters
\mathcal{L}_{dyn}	$x_{\text{dyn},i,k}^n$	$u_{\text{dyn},i,k}^n(t) = \sum_{j,m} w_{ij}^{\text{dyn},k} z_{\text{ev},j}^m(t) - I^{\text{dis}}(c_k^1(t), \dots, c_k^{M_C}(t))$
\mathcal{L}_{dyn}	$x_{\text{dyn},i,k}^n$	$w_{ij}^{\text{dyn},k} = \frac{q_{ij}^k}{M}$ for $i \neq j$, $w_{ii}^{\text{dyn},k} = \frac{1}{\tau M} - \frac{1}{M} \sum_{j \neq i} q_{ij}^k$ for all i .
\mathcal{L}_{ev}	$z_{\text{ev},i}^n$	$u_{\text{ev},i}^n(t) = \sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} - I^{\text{dis}}(x_{\text{dyn},i,1}^n(t), \dots, x_{\text{dyn},i,N_C}^n(t)) - I^{\text{lat}}(t)$
\mathcal{L}_{ev}	$z_{\text{ev},i}^n$	$w_{il}^{\text{ev}} = \log \lambda_{li} + \alpha$, $b_i^{\text{ev}} = \frac{1}{\tau} + \lambda_{\text{max}} - \lambda_i$

Here we have defined $\lambda_i \equiv \sum_j \lambda_{ji}$ and $\lambda_{\text{max}} = \max_j \{\lambda_j\}$. $I^{\text{lat}}(t)$ denotes lateral inhibition and I^{dis} disinhibition. α is an arbitrary constant. doi:10.1371/journal.pcbi.1003859.t002

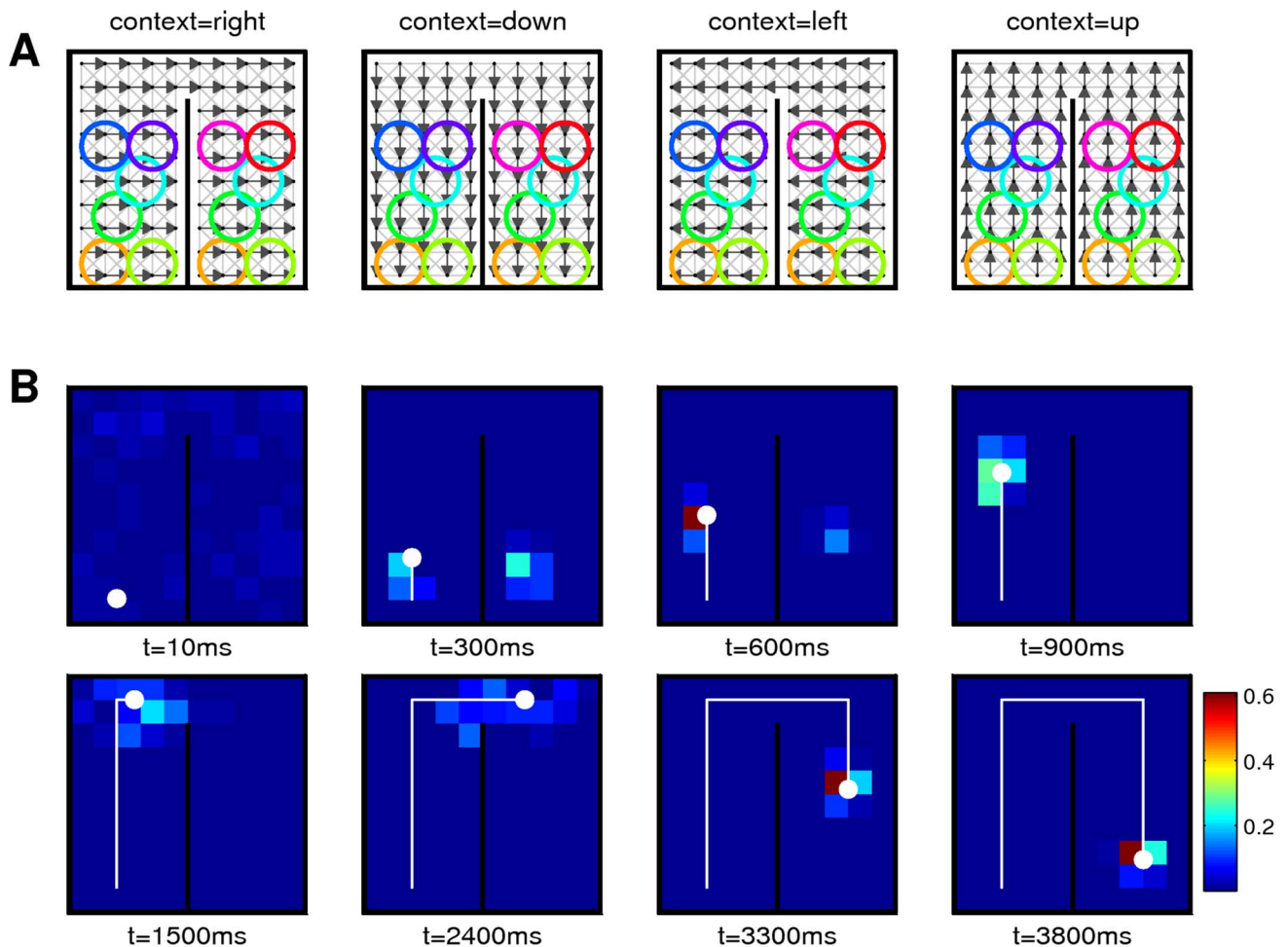


Figure 10. Self-localization through particle filtering in ENS coding. A) Two-chamber maze (black lines) and transitions between states in various contexts. States are arranged on a 10×10 grid in the maze (crossing points of gray lines). Light gray lines indicate bidirectional state transitions with low transition rates (0.1 Hz). Dark gray arrows indicate transitions with high transition rates (3.5 Hz). Context is defined by movement direction (right, down, left, up). Colored circles indicate sensory evidence. Each color stands for one afferent neuron with a Gaussian spatial receptive field. The circle indicates the STD of the Gaussian. Note that the southern chambers give rise to identical observations. Observations are truncated at the height of the opening between the chambers such that no observations are experienced in the most northern parts. **B)** Network estimate of posterior probability (see color bar on the right for color code) for one trajectory through the maze (white trace; dot denotes current position) at different times. Spatial layout as in A. Various phases of the trajectory are shown: Uninformative prior knowledge ($t = 10$ ms), ambiguous estimates ($t = 300$ ms); disambiguation ($t = 600, 900$ ms); states without evidence ($t = 1500, 2400$ ms); unambiguous state estimation based on ambiguous evidence ($t = 3300, 3800$ ms). doi:10.1371/journal.pcbi.1003859.g010

The ambiguous target task revisited. The ambiguous target task has a clear sequential structure (initial fixation, spatial cue, memory epoch, color cue), see Figure 1A. The corresponding hidden variable $c(t)$ – for which the current value is given by the momentary stage of the experimental trial – is shown in Figure 11Aa.

The modeled task is similar to the ambiguous target task considered above with the difference that several trials of the task are performed sequentially, i.e., a sequence of epochs (fixation, spatial cue, memory, color cue) is directly followed by the fixation epoch, indicating the start of a new trial, again followed by the spatial cue and so on. The difficulty of this task is that in the fixation epoch, the internal belief about the random variable $v(t)$ that encodes the current color-direction pair is highly biased by the last trial. This bias is problematic since an optimal prior would assign equal probability to all color-direction pairs. In other words, during the fixation, $v(t)$ should be reset. A reset can be

accomplished by assuming that during fixation the dynamics of $v(t)$ are such that $v(t)$ can quickly change its state from any state to any other state (i.e., transition rates are high between all pairs of states, see Figure 11Ab). In other epochs however, the assumption is that the value of the random variable is fixed and does not change (transition rates are zero between all pairs of distinct states). We arrive at a context-dependent Bayesian filtering problem where the dynamics of $v(t)$ depend on the current estimate of $c(t)$, the epoch in the trial.

We modeled the context-dependent reset of the internal belief by extending the circuit for the ambiguous target task in the following manner. The internal belief about the random variable $v(t)$ that encodes the current color-direction pair (Figure 11Ab) is generated by a particle filter circuit. Context is provided to this circuit by an estimate of $c(t)$, the current epoch in the trial (Figure 11B). This estimate is performed by a particle filter circuit that receives the same sensory evidence but no context (task class

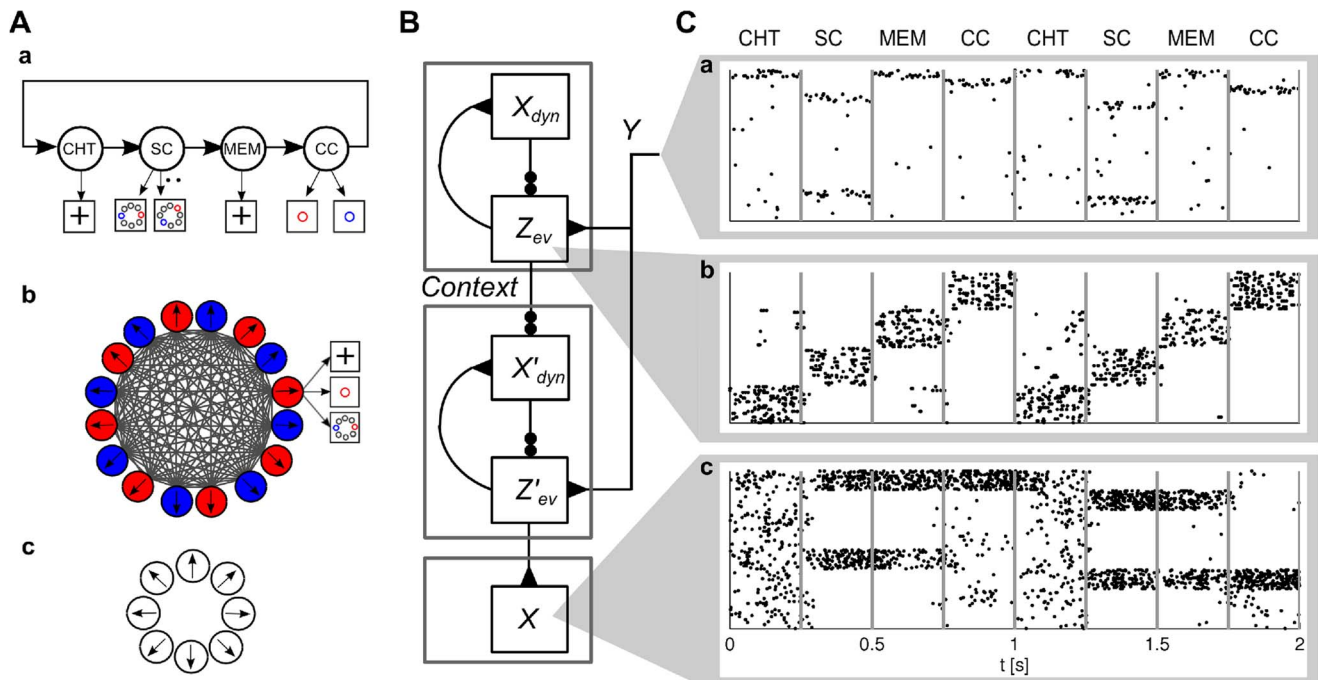


Figure 11. Context-dependent Bayesian filtering in two successive trials of the ambiguous target task. **A**) Represented random variables. **Aa**) Dynamics of a random variable that codes the current phase in a trial of the ambiguous target task (CHT: fixation; SC: spatial cue; MEM: memory cue; CC: color cue). Possible observations in each phase are indicated in boxes. **Ab**) Context-dependent Bayesian filtering is performed for a random variable with 16 hidden states corresponding to direction-color pairs as in Figure 6Aa. Gray lines indicate context-dependent transitions. All-to-all transitions are possible in the fixation phase (CHT). There are no transitions in other phases of a trial. **Ac**) The action readout layer infers a color-independent random variable by marginalization over color in each direction. **B**) Circuit structure. The circuit on the top (ensembles X_{dyn} and Z_{ev}) performs Bayesian filtering on the random variable $c(t)$ indicated in panel (Aa). It provides context for another particle filter circuit (middle gray box; ensembles X'_{dyn} and Z'_{ev}) that generates a belief about the random variable $v(t)$ indicated in Ab. An action readout layer is added (bottom gray box; ensembles X). **C**) Spike rasters from a simulation of two successive trials for afferent neurons (Ca), neurons in the particle filter circuit for the phase in the trial (Cb), and neurons in the action readout layer (Cc). Neurons in Cb are coding for the current phase of the trial (ordered from bottom to top: CHT, SC, MEM, and CC). Neuron ordering in Ca and Cc as in Figure 6. doi:10.1371/journal.pcbi.1003859.g011

C). In the context of the fixation-epoch, $v(t)$ is believed to change its state rapidly (all-to-all transitions in the dynamics of the random variable, gray connections in Figure 11Ab). In the context of other epochs, the random variable is believed to be constant (no transitions between states, i.e., pure evidence integration). Finally, an action readout layer derives the belief over the rewarded action. In Figure 11C, a simulation for two successive trials of the ambiguous target task is shown. The epoch $c(t)$ is correctly inferred by the particle filter circuit that provides the epoch-context (Figure 11Cb). This context influences information processing in the subsequent circuit for $v(t)$ (Figure 11Cc). While information is retained during the memory epochs, the belief about the rewarded movement direction returns to an uninformative prior during the memory epoch at 1 s to 1.25 s. This example demonstrates that internally generated beliefs about random variables can act as valuable context in ENS coding.

Discussion

It has recently been demonstrated that the dynamics of recurrent networks of spiking neurons can perform MCMC sampling on a distribution P [4,9]. The distribution can be approximately recovered by observing the evolution of the network state trajectory for some time. Such a temporal representation is less suitable for distributions that have to be updated rapidly since each estimate of P needs several hundreds of milliseconds. We have therefore proposed and analyzed in this

article ENS coding, where ensembles of neurons code for each state of a random variable. In this coding scheme, adaptations of internal beliefs can be established on the time-scale of EPSPs, see Figures 5E and 8. Similarly, downstream readout neurons can rapidly estimate ENS coded probability values according to eq. (1) from neural ensembles, while such readout operation from neural sampling networks demands the integration of spikes over intervals of several hundred milliseconds. Another deficiency of the neural sampling approach is the need for unbiologically strong synaptic connections. This results from the principle that random variables are encoded by single neurons in neural sampling, which necessitates strong connections in order to ensure sufficient impact on postsynaptic targets. Unbiologically strong synaptic connections are not necessary in ENS coding, since targets can be activated in a cooperative manner by neuronal ensembles. This is for example apparent in the inverse scaling of weights with ensemble size in eq. (6).

We have shown that particle filtering [6,7] can be performed by circuits of spiking neurons in ENS coding. Numerous engineering applications of particle filtering to tasks belonging to task class D exist. In such tasks, particles are evolved according to a dynamics-model that depends on context, such as the movement direction in a self-localization task. This particle filtering with context cannot be emulated by approximate Bayesian filtering as described in task class C. To the best of our knowledge, this article provides the first proof that this powerful operation is in principle accessible to spiking neural circuits. We have demonstrated in computer

simulations that ENS coding enables neuronal circuits to perform these essential operations with high fidelity, thus making them suitable for higher-level decision-related processing. Lee and Mumford [30] proposed that particle filtering could be the basic computational operation in hierarchical cortical processing. We have demonstrated a first step in that direction in a spiking model with ENS coding by showing that the belief about a random variable can provide context information for the temporal processing related to other variables, see *The ambiguous target task revisited*.

Model simplifications and possible extensions

For consistency with the neural sampling approach, we used in this article rectangular shaped EPSPs. To test how deviations from rectangular EPSPs effect circuit performance, we performed simulations where all EPSPs were modeled as exponentially decaying EPSPs with decay time constant $\tau = 20\text{ms}$ such that the integral over the EPSP is unchanged. The particle filter circuit was tested in the generic setup for task class C (see Figure 8A). Despite of its strong deviation from the rectangular shape, exponential EPSPs caused only a slight decrease of circuit performance (the percentage of incorrect state estimates was $17.5 \pm 5.2\%$ as compared to $14.8 \pm 3.8\%$ for rectangular EPSPs).

The model also uses particular forms of instantaneous lateral inhibition and disinhibition. Although both forms of inhibition have been reported to play crucial roles in cortical information processing [20–23,28,29], the exact circuitry and function of those inhibitory circuits is still unknown. The lateral inhibition used in our circuit model is however consistent with a recent study which showed that inhibition has a broader spatial selectivity than excitation in visual cortex of awake mice [31]. Such broader selectivity is expected in our model since lateral inhibition is common to all ensembles for a random variable. Apart from that, the assumption that inhibition follows excitation instantaneously is clearly a model simplification. A recent experimental study [32] revealed that cortical inhibition lags excitation by about 3 ms in anesthetized rats (the lag is possibly smaller in awake animals, see [31]). In order to test whether the circuit is tolerant to delayed inhibition, we performed control simulations where in addition to the use of exponential EPSP shapes, both lateral inhibition and disinhibition was delayed by 3 ms in the generic setup for task class C (see Figure 8A). We found that delayed lateral inhibition can lead to activity peaks as excitation arising from incoming evidence cannot be compensated rapidly. This can lead to unstable circuit activity when lateral inhibition tries to compensate these peaks in the delayed negative feedback loop. Instabilities can be avoided by reduction of the inhibitory drive, i.e., by reducing the lateral inhibition scaling I_0^{lat} in eq. (33). Simulations showed that with reduced inhibition scaling, the network tolerates delayed inhibition with a slight decrease of performance (the percentage of incorrect state estimates was $19.8 \pm 4.5\%$ as compared to $17.5 \pm 5.2\%$ without delay; $I_0^{\text{lat}} = 0.125$).

In ENS coding, each neuron is tuned to one value of a random variable $v(t)$. The encoded random variable may represent a specific feature relevant for some task. Many experiments show that such tunings exist in various cortical areas, such as the tuning of PMd neurons to potentially rewarded movement direction [3]. However, the random variable $v(t)$ does not necessarily correspond to a single task-relevant feature. For example, the random variable encoded in the particle filter circuit for the ambiguous target task represents direction-color pairs (see Figure 6Aa). Therefore, neurons in this circuit are selective for both the spatial cue and the color cue. This mixed tuning helps to integrate the temporally separated cues. Mixed selectivity of neurons has been

found in higher cortical areas such as prefrontal cortex, and its computational benefits have been highlighted in [33]. Hence, ENS coding is consistent with these findings. We note however that in the pure formulation of ENS coding, mixed selectivity to many task aspects is problematic since the number of ensembles necessary to encode all possible states over d task dimensions grows exponentially with d . One possibility to overcome exponential growth for large d is to consider approximations schemes such as neglecting mixed configurations that are highly unlikely.

The important question how the parameters of the network could be attained by learning from experience is outside of the scope of this paper. However, some possible solutions to the learning problem can be sketched. For the particle filtering circuit, two classes of synaptic connections could be adapted through learning processes. First, synaptic efficacies w_{ij}^{ev} from evidence neurons to neurons in layer \mathcal{L}_{ev} . These efficacies encode the log-firing rates of evidence neurons for the given hidden state i . It has been shown that hidden-cause representations that require such synaptic efficacies can be learned in spiking neural networks with lateral inhibition through spike-timing-dependent (STDP)-like synaptic plasticity rules [34,35]. Hence, it seems quite feasible that the efficacies w_{ij}^{ev} can be attained in a self-organized manner through STDP. The second type of connections, w_{ij}^{dyn} from \mathcal{L}_{ev} to \mathcal{L}_{dyn} encode the dynamics of the random variable in terms of the rate of change q_{ji} from state j to state i . In other words, the synapse needs to track how often neuron $x_{\text{dyn},i}^m$ is active after neuron $z_{\text{ev},i}^m$ (the synaptic efficacies needed for task class A are of a similar nature). Again, this tracking can be done by a temporal Hebbian learning rule. In particular, it has recently been shown in [36] that such temporal relationships can be learned through STDP-like learning rules in networks of spiking neurons that implement hidden Markov models. A more sophisticated learning approach that requires additional circuitry was outlined in [37]. Of course, these results do not immediately generalize to the architecture proposed in this article, and further studies are needed to prove the viability of such a learning approach. Finally, we note that the feature of the particle filter circuit (and ENS coding in general) that individual synaptic efficacies do not need to be adjusted to exact values as long as the mean efficacy between ensembles is correct (see *Particle filtering in a generic setup for task class C*) may prove advantageous for learning processes.

Experimentally testable predictions

We investigated the behavior of the model and compared it to experimental results. Our results so far indicate that ENS coding is consistent with a number of experimental studies. It is noteworthy that the lateral inhibition that is needed to stabilize network firing rates leads to the typical transient ensemble rate increases at stimulus-onset (see transient responses in Figure 5D and Figure 1C in *Text S5*).

While many laboratory tasks implement a variant of evidence integration (task class B), there is a lack of studies in the experimental literature for task classes D and E. We hypothesized in *Task class E: Internal beliefs as context* that the current values of important context variables are estimated in higher brain areas. Hence, one prediction of the model is that neural activities in such areas should not only be related to variables of primary interest, but also to context such as the current phase in a task with sequential structure, see Figure 11B,C. Evidence for such representations in monkey dorsolateral prefrontal cortex has been reported [38]. Our model also predicts that activity in neuronal ensembles that represent context should modulate the activity of

decision-related ensembles in a manner that is fundamentally different from the impact of direct evidence. In particular, according to our circuit models in task classes D and E, context gates or modulates activity in such neurons. Task context has been shown to modulate neuronal activity in primate prefrontal cortex [39] as well as in various lower level visual areas including area MT [40,41]. These findings are contrasted by studies that showed that task-context influenced noise-correlations but not firing rates in MT in a variant of the random-dot motion task [42]. The general term context of course subsumes many different types of contextual information in various quite different task settings, which may explain the discrepancies between the different studies. In this work, context is defined specifically as additional information about the dynamics of the random variable. Experimental setups where context information is indicative of the dynamics of task-relevant variables would help to elucidate how such information alters temporal processing in cortical circuits.

We proposed disinhibition as one possibility for context-dependent modulation, consistent with the experimental findings that disinhibitory circuits are recruited by feedback connections in neocortical layer 1 [28]. We note however that such effects could be implemented in cortical circuits by a number of mechanisms [18,19].

We applied particle filtering – one of the most successful techniques for self localization in autonomous robots – in ENS coding to self-localization, a particularly important task for many animals. Whether self-localization in animals is solved in a similar manner is of course still unknown. There have however been studies which show that ambiguous sensory information can be resolved on the neuronal level in rodents [43,44]. This indicates that the algorithm employed by the brain is in fact quite powerful. The particular implementation proposed here also implies that the spatial structure of the environment (i.e., possible transitions between locations in space) should be encoded in the synaptic weight matrix of particular neural circuits (edges and arrows in Figure 10A). Furthermore, our model predicts that information about motor events (movement) is treated by such circuits as context. Hence, these signals impact circuit activity quite differently from sensory evidence, see the discussion above. Evidence for nonlinear interaction of visual information and movement information in self-localization of mice has been reported recently [44].

Related work

Probabilistic population codes (PPC; [24]) have been suggested as one hypothesis how probability distributions could be coded in the spiking activity of neurons. In the PPC concept, one assumes that each neuron is (at least implicitly) linked to a stimulus via a tuning function. A hypothetical decoder would then apply Bayes rule to decode the stimulus distribution, making use of the tuning functions of the neurons. In ENS coding considered here, the neural ensembles produce samples from a distribution and a hypothetical decoder would just count spikes. Accumulation of evidence in LIP in a random-dot motion task has been modeled in the PPC framework [45]. In general, information can be accumulated in PPC simply by adding up activity from afferent neurons, given that this activity follows Poisson-like statistics. The model however assumes that the hidden variable is static. In fact, tracking of dynamic variables, such as those considered in task classes C–E, is hard to implement in the PPC framework [46].

Several spiking neural network models for Bayesian filtering have been proposed in the literature with very similar basic ideas to solve the problem [15,47,48]. Denève [47] proposed a model

where a single integrate-and-fire neuron estimates the hidden state of a binary random variable with temporal dynamics. The model can only deal with binary variables, whereas our proposed model is not restricted in this respect. A similar model was proposed in [48]. There the assumption was that the continuous-valued random variable evolves according to a drift-diffusion process. Our implementation of Bayesian filtering in ENS coding complements this work by considering discrete-valued random variables with the assumption that the dynamics can be approximated by a continuous-time Markov chain.

Rao [15] considered Bayesian filtering in discrete time through a network of spiking neurons. This model was however not based on a rigorous coding scheme with respect to information transfer through spikes. The instantaneous firing rate of individual neurons was regarded as the distribution-encoding quantity and it was implicitly assumed that spikes can communicate this quantity in sufficient quality. In the current article, we base representations of beliefs on the spiking activity in the first place and propose ENS coding as a solution where the fidelity of representation is provided through ensemble activity. Our analysis identified how network properties such as the ensemble size, the maximal firing rate, and the membrane time constant influence the quality of the representation. In any case, the noise introduced by stochastic spiking cannot be neglected in general. We have demonstrated in computer simulations that still, temporal information processing on demanding tasks is possible in ENS coding.

We have argued that particle filtering with context is an important operation that is needed for example for self-localization. The current work shows that this extension of Bayesian filtering, that has not been considered in previous models, can easily be implemented in the ENS code.

Several non-spiking models for Bayesian filtering have been proposed previously [14,49,50]. We have based our circuit model for task classes B and C on well-known filtering equations [13] that also provided the basis for the rate-based model considered in [14]. Several conclusions can be drawn when comparing the model based on ENS coding considered here with the non-spiking model from [14]. First, we confirmed through computer simulations that quite demanding information processing tasks are possible with spiking neurons using ENS coding, despite of substantial noise introduced by stochastic spiking. Second, ensemble coding is clearly necessary for the tasks considered in this article if neuronal responses are stochastic. We used on the order of 1000 neurons per ensemble in the simulations. This number was not optimized, but ensemble sizes below 100 are not sufficient, for example in the ambiguous target task (see *Dataset S1*). Third, besides the complications that stochastic spike codes introduce, we have shown that ENS coding also has some positive effects. We found that multiplicative operations can be replaced by gating of neuronal activity in ENS coding, for example through synaptic gating or through disinhibition. This property of ENS coding provides an attractive alternative to previously proposed solutions for the unavoidable demand of nonlinear processing in Bayesian filtering, such as multiplicative interaction of synaptic inputs [14,50], or the use of precise dendritic nonlinearities [15]. Finally, the use of a spiking model enabled us to directly compare model characteristics to experimental data. We found that the model is consistent with quite diverse experimental results [1–3,25].

Conclusions

Sample-based representations of probability distributions provide an attractive framework for modeling probabilistic inference on static evidence in cortical networks. We have shown that

ensemble-based neural sampling enables cortical networks to perform also powerful context-dependent temporal inference. Hence, our model provides a new and theoretically founded basis for understanding temporal probabilistic computations in various higher-level cortical areas.

Methods

For easy reference, Table 3 summarizes the notational conventions used in this article.

Spike trains, EPSP shapes, and EPSP-filtered spike trains

Here, we define EPSP-filtered spike trains and the shape of EPSPs used throughout the article. We denote the spike-train \hat{x}_i^n of a neuron x_i^n as the sum of Dirac delta functions

$$\hat{x}_i^n(t) = \sum_j \delta(t - t_i^{n,j}),$$

where $t_i^{n,j}$ denotes the j -th spike-time of neuron x_i^n . We define the EPSP-filtered spike train $x_i^n(t)$ as

$$x_i^n(t) = \sum_{j: t_i^{n,j} \leq t} \epsilon(t - t_i^{n,j}). \quad (18)$$

Here ϵ denotes the EPSP shape produced by a spike of a presynaptic neuron. We use in this article rectangular EPSP shapes of the form

$$\epsilon(s) = \Theta(s)\Theta(\tau - s), \quad (19)$$

where Θ denotes the Heaviside step function and τ is the length of the EPSP. For some control simulations, we use exponentially decaying EPSP shapes

$$\epsilon(s) = \Theta(s)\exp(-s/\tau). \quad (20)$$

ENS code and filtered probability distributions

Assuming rectangular EPSP shapes of length τ , the filtered probability denoted as $\bar{P}(v(t)=i)$ is the average probability for

state i in time $[t-\tau, t]$, that is

$$\bar{P}(v(t)=i) = \frac{1}{\tau} \int_0^\tau P(v(t-s)=i) ds. \quad (21)$$

We denote the ensemble average of a random variable by $\langle \cdot \rangle$. Consider a given fixed temporal evolution of the probability $P(v(t)=i)$. In ENS coding, we demand that neural activities are such that the mean of the estimator $\tilde{P}(v(t)=i)$ is equal to the temporally filtered probability of that state at time t , that is

$$\langle \tilde{P}(v(t)=i) \rangle = \bar{P}(v(t)=i) \text{ for } i=1, \dots, N. \quad (22)$$

Mean and variance of the estimator

Here we show that eq. (1) is an unbiased estimator of the probability of state i at time t filtered by the EPSP, given that there is at least one spike in the integration window. Additionally we compute the variance of the estimator.

Consider a given fixed temporal evolution of the probability $P(v(t))$. We first consider the mean of $\tilde{P}(v(t)=i)$ for a given number of spikes $\sum_j X_j(t)=k$, written as $\langle \tilde{P}(v(t)=i) | k \rangle$:

$$\langle \tilde{P}(v(t)=i) | k \rangle = \frac{\langle X_i(t) | k \rangle}{k}.$$

Given a spike at time t , the probability that it was elicited in ensemble X_i is $\frac{\rho_i(t)}{\rho(t)}$, where $\rho(t) = \sum_j \rho_j(t)$ denotes the total network rate. Since the total network rate $\rho(t) = \rho_{\max} M$ is constant and each spike is drawn independently, to count the spikes in different ensembles in a time window τ , we can replace each inhomogeneous Poisson process by a homogeneous process in that time window with rate $\bar{\rho}_i(t) = \frac{1}{\tau} \int_0^\tau \rho_i(t-s) ds = \rho_{\max} M \bar{P}(v(t)=i)$, where $\bar{P}(v(t)=i)$ denotes the temporally filtered probability of the state, see eq. (21). Each individual spike in the time window originates from ensemble X_i

Table 3. Notation.

Variable name	Description
v	Random variable with range $\{1, \dots, N\}$.
N	Number of states of v and number of ensembles that represent v .
X_i	Ensemble i represents the belief that a random variable is in state i .
M	Number of neurons per ensemble.
x_i^m	m -th neuron in ensemble X_i .
$\hat{x}_i^m(t)$	Spike train of neuron x_i^m at time t .
$x_i^m(t)$	EPSP-filtered spike train of neuron x_i^m at time t .
$X_i(t)$	Summed activity (probability mass) for state i at time t : $X_i(t) = \sum_m x_i^m(t)$.
$u_i^m(t)$	Membrane potential of neuron x_i^m at time t .
$\rho_i^m(t)$	Instantaneous firing rate of neuron x_i^m at time t .

Description of frequently used variables for easy reference. In general, capital letters refer to ensembles and lower case letters to neurons in these ensembles. doi:10.1371/journal.pcbi.1003859.t003

with probability $\frac{\bar{\rho}_i(t)}{\rho_{\max}M} = \bar{P}(v(t)=i)$. Hence, for a given k , $X_i(t)$ is drawn from a binomial distribution $\mathcal{B}(k, \bar{P}(v(t)=i))$. It follows that $\langle X_i(t)|k \rangle = k\bar{P}(v(t)=i)$ and $\text{Var}(X_i(t)|k) = k\bar{P}(v(t)=i)(1 - \bar{P}(v(t)=i))$. Hence, $\langle \tilde{P}(v(t)=i) \rangle = \bar{P}(v(t)=i)$, and the estimator is unbiased.

We now turn to the variance of $\tilde{P}(v(t)=i)$. We define $\tilde{P}_i \equiv \tilde{P}(v(t)=i)$ and $\bar{P}_i \equiv \bar{P}(v(t)=i)$ for notational convenience. From $\text{Var}(X_i(t)|k) = k\bar{P}_i(1 - \bar{P}_i)$, we obtain $\text{Var}(\tilde{P}_i|k) = \bar{P}_i(1 - \bar{P}_i)/k$. By the law of total variance, we have

$$\text{Var}(\tilde{P}_i) = \langle \text{Var}(\tilde{P}_i|k) \rangle + \text{Var}(\langle \tilde{P}_i|k \rangle).$$

The second summand is 0 since $\langle \tilde{P}_i|k \rangle = \bar{P}_i$ is independent of k . We thus obtain

$$\text{Var}(\tilde{P}_i) = \bar{P}_i(1 - \bar{P}_i) \sum_{k=1}^{\infty} P(k) \frac{1}{k}.$$

Here, $P(k)$ is Poissonian with intensity $L = \rho_{\max}M\tau$. Inserting the Poisson density, we obtain

$$\begin{aligned} \text{Var}(\tilde{P}_i) &= \bar{P}_i(1 - \bar{P}_i) \sum_{k=1}^{\infty} \frac{L^k}{k!} e^{-L} \frac{1}{k} \\ &\approx \bar{P}_i(1 - \bar{P}_i) \sum_{k=1}^{\infty} \frac{L^k}{k!} e^{-L} \frac{1}{k+1} \\ &= \bar{P}_i(1 - \bar{P}_i) \frac{1}{L} \sum_{k=1}^{\infty} \frac{L^{k+1}}{(k+1)!} e^{-L} \\ &= \bar{P}_i(1 - \bar{P}_i) \frac{1}{L} (1 - P(0) - P(1)) \approx \bar{P}_i(1 - \bar{P}_i) \frac{1}{L} \\ &= \bar{P}_i(1 - \bar{P}_i) \frac{1}{\rho_{\max}M\tau}. \end{aligned}$$

The approximation is excellent if L is large, i.e., a small number of spikes over all ensembles within τ is very unlikely.

Influence of the firing rate on the represented distribution

We provide here the proof for eq. (2). We assume that an antiderivative R_i^m exists for all rates ρ_i^m (which is satisfied for example if the rates are continuous or sums of Heaviside step functions). The probability mass of a population X_i is given by

$$X_i(t) = \sum_m x_i^m(t) = \sum_m \int_{t-\tau}^t \hat{x}_i^m(s) ds, \quad (23)$$

where \hat{x}_i^m is the spike train of neuron x_i^m as defined above. The neurons spike in a Poissonian manner with continuous rates ρ_i^m . Hence we obtain for the mean over realizations of spike trains

$$\langle X_i(t) \rangle = \sum_m \int_{t-\tau}^t \rho_i^m(s) ds = \sum_m R_i^m(t) - \sum_m R_i^m(t-\tau). \quad (24)$$

It follows

$$\begin{aligned} \frac{d}{dt} \langle X_i(t) \rangle &= \sum_m \frac{d}{dt} R_i^m(t) - \sum_m \frac{d}{dt} R_i^m(t-\tau) \\ &= \sum_m \rho_i^m(t) - \sum_m \rho_i^m(t-\tau). \end{aligned} \quad (25)$$

Task class A: Simple probabilistic dependencies

We first show that the expected value of the estimator $\tilde{P}(v(t))$ is equal to the posterior distribution (4) for membrane potentials (5) and weights (6). We will then derive the variance of an alternative estimator $\hat{P}(v(t))$ of the posterior distribution.

Consider two distributions $P(v^X(t))$ and $P(v^Z(t))$ over random variables $v^X(t)$ and $v^Z(t)$ respectively such that the desired posterior is given by eq. (4). $P(v^Z(t))$ is coded by neurons z_j^m through ENS coding. We denote the estimation sample size of the ensembles for $v^Z(t)$ by L^Z for clarity. Consider a circuit with neurons x_i^m and ensemble size M that should represent the posterior. The membrane potentials are given by $u_i^m(t) = \sum_{j,m} w_{ij} z_j^m(t)$ for $w_{ij} = \frac{\alpha}{M\tau} P(v^X = i|v^Z = j)$. For given EPSP-filtered spike trains $z_j^m(\cdot)$, this leads to the firing rate for neuron x_i^m

$$\rho_i^m(t) = \alpha \frac{1}{M\tau} \sum_j P(v^X = i|v^Z = j) Z_j(t). \quad (26)$$

Averaging over realizations of spike trains in the posterior population, we obtain

$$\begin{aligned} \langle X_i(t) | Z_1, \dots, Z_N \rangle &= \sum_n \int_{t-\tau}^t \rho_i^m(s) ds \\ &= \frac{\alpha}{\tau} \sum_j P(v^X = i|v^Z = j) \int_{t-\tau}^t Z_j(s) ds, \end{aligned}$$

where $\langle \cdot | Z_1, \dots, Z_N \rangle$ denotes the average for given activities in ensembles Z_1, \dots, Z_N . Taking also the average over realizations of spike trains in ensembles Z_j , this evaluates to

$$\begin{aligned} \langle X_i(t) \rangle &= \frac{\alpha}{\tau} \sum_j P(v^X = i|v^Z = j) \langle \int_{t-\tau}^t Z_j(s) ds \rangle \\ &= \frac{\alpha}{\tau} \sum_j P(v^X = i|v^Z = j) \int_{t-\tau}^t \langle Z_j(s) \rangle ds. \end{aligned}$$

We assume for simplicity that $P(v^Z(t))$ is constant over time, and obtain

$$\langle X_i(t) \rangle = \alpha L^Z \sum_j P(v^X = i|v^Z = j) P(v^Z(t) = j) = L^X P(v^X(t) = i),$$

where we defined $L^X = \alpha L^Z$. This shows that the represented probability of the circuit is the posterior probability in the mean with an estimation sample size of $L^X = \alpha L^Z$.

Variance of the posterior representation. The estimate of the posterior distribution at some specific time t is however variable due to variability in spike counts of both the representation of $P(v^Z(t))$ and the representation of $P(v^X(t))$. Due to this doubly stochastic nature, the variance of the estimator $\hat{P}(v^X(t))$ is hard to evaluate. We derive in here the variance of an alternative estimator $\hat{P}(v^X(t)=i) = \frac{X_i(t)}{L}$ which is also unbiased but has higher variance. We show that the variance of this estimator for conditional probabilities $P(v^X|v^Z)$ that maximize the variance of the firing rates in the posterior circuit is at most twice the variance of the estimator $\hat{P}(v^Z(t))$. The firing rate of a neuron in the posterior representation has a variance of

$$\begin{aligned} \text{Var}(\rho_i^n(t)) &= \left(\frac{L^X}{L^Z M \tau} \right)^2 \sum_j P(v^X=i|v^Z=j)^2 \text{Var}(Z_j(t)) \\ &= \frac{(L^X)^2}{L^Z M^2 \tau^2} \sum_j P(v^X=i|v^Z=j)^2 P(v^Z(t)=j) \\ &\leq \frac{L^X}{L^Z M^2 \tau^2} L^X P(v^X=i), \end{aligned}$$

where the maximum is achieved when one $P(v^X=i|v^Z=j)=1$ for some j . We derive the variance of the spike count in a window of size τ in the posterior circuit in this case. This variance is not straight forward to compute since it is the variance in spike count of a Poisson process with a rate that is itself a random variable. For the case of maximum variance of the firing rate, the firing rate is given by

$$\rho_i^n(t) = \frac{1}{\tau} \frac{L^X}{L^Z M} Z_j(t)$$

for some j . For the spike count in a window of duration τ over the whole ensemble for state i , that is $X_i(t)$, we consider thus a Poisson-distributed random variable with intensity $\frac{L^X}{L^Z} Z_j(t)$ which is itself a random variable. Since $Z_j(t)$ is Poisson distributed, $X_i(t)$ can be expressed as a compound Poisson distribution in the following way. $X_i(t)$ is the sum of random variables z_k which are i.i.d. Poisson with intensity 1 and scaled by $\frac{L^X}{L^Z}$. The number of random variables that are summed is Poisson distributed with intensity $\langle Z_j(t) \rangle = L^Z P(v^Z(t)=j)$. Hence, the random variable $X_i(t)$ is given by

$$X_i(t) = \sum_{k=0}^{Z_j(t)} z_k.$$

It is easy to see that this leads to the correct distribution over intensities for $X_i(t)$. It is known from the theory of compound Poisson processes [51] that the variance of $X_i(t)$ is given by $\langle Z_j(t) \rangle \text{Var}(z_i) + \langle z_i \rangle^2 \text{Var}(Z_j(t))$ which is in our case $2L^X \left(\frac{L^X}{L^Z} \right) P(v^Z(t)=j) = 2L^X \left(\frac{L^X}{L^Z} \right) P(v^X(t)=i)$. Since the estimated probability for state i is $\hat{P}(v^X(t)=i) = X_i(t)/L^X$, we obtain $\text{Var}(\hat{P}(v^X(t)=i)) = \frac{2}{L^Z} P(v^X(t)=i)$. In comparison with the variance of $\hat{P}(v^Z(t)=j)$, which is $\text{Var}(\hat{P}(v^Z(t)=j)) = \frac{1}{L^Z} P(v^X(t)=i)$, the variance doubles.

Task class B: Evidence integration

Evidence can be provided through EPSPs. We first prove that in a rate-model, integration of evidence signaled via point-

events can be performed with finite-length EPSPs. We denote the spike-train \hat{y}_l of afferent neuron y_l as the sum of Dirac delta functions

$$\hat{y}_l(t) = \sum_j \delta(t-t_j^l),$$

where t_j^l denotes the j -th spike-time of neuron l . Consider a set of non-normalized and non-negative functions $p_i(\cdot)$ from which the probabilities $P(v(t)=i)$ follow after normalization. The change of p_i is given by [13,14]

$$\frac{d}{dt} p_i(t) = \left[\sum_l (\lambda_{li} - 1) \hat{y}_l(t) - \lambda_i \right] p_i(t),$$

where λ_{li} is the firing rate of afferent neuron l if the current state is i . Starting at some time t_0 with $p_i(t_0)$ and integrating up to time $t > t_0$, we obtain

$$p_i(t) = p_i(t_0) \exp(-(t-t_0)\lambda_i) \prod_l \lambda_{li}^{N_l(t_0,t)}, \quad (27)$$

where $N_l(t_0,t)$ denotes the number of spikes of afferent neuron l in $[t_0,t]$.

Now assume finite-size EPSPs of length τ and integral $\bar{\epsilon}$, giving rise to the EPSP-filtered spike-trains y_l . Furthermore, assume that the rate-based network evolves due to eq. (8) with weights given by $w_{il}^{\text{ev}} = \frac{1}{\bar{\epsilon}} \log \lambda_{li} + \alpha$ for some constant α . Integrating up to time t and assuming no evidence spike in $[t-\tau, t]$, we obtain

$$\begin{aligned} p_i(t) &= p_i(t_0) \exp \left(\int_{t_0}^t \left(\sum_l w_{il}^{\text{ev}} y_l(s) + \lambda_{\text{max}} - \lambda_i + g(s) \right) ds \right) \\ &= p_i(t_0) \exp \left(\int_{t_0}^t \lambda_{\text{max}} + g(s) ds \right) \exp \left(\int_{t_0}^t \left(\sum_l w_{il}^{\text{ev}} y_l(s) - \lambda_i \right) ds \right) \\ &= c p_i(t_0) \exp(-(t-t_0)\lambda_i) \exp \left(\sum_l w_{il}^{\text{ev}} \int_{t_0}^t \sum_{t_j^l \in [t_0,t]} \epsilon(s-t_j^l) ds \right) \\ &= c p_i(t_0) \exp(-(t-t_0)\lambda_i) \prod_l \exp \left(w_{il}^{\text{ev}} \sum_{t_j^l \in [t_0,t]} \int_{t_0}^t \epsilon(s-t_j^l) ds \right) \\ &= c p_i(t_0) \exp(-(t-t_0)\lambda_i) \prod_l \exp(w_{il}^{\text{ev}} N_l(t_0,t) \bar{\epsilon}) \\ &= c p_i(t_0) \exp(-(t-t_0)\lambda_i) \prod_l \exp \left(\log \lambda_{li}^{N_l(t_0,t)} + \alpha \bar{\epsilon} N_l(t_0,t) \right) \\ &= c' p_i(t_0) \exp(-(t-t_0)\lambda_i) \prod_l \lambda_{li}^{N_l(t_0,t)} \end{aligned}$$

for a constant c' that only scales the p_i 's. We thus obtained the desired result, compare to eq. (27). Note that α can be used to shift weights to positive values for low firing rates. Since the EPSPs are scaled by the weight, we can assume that $\bar{\epsilon} = 1$. Then the optimal weights are $w_{il}^{\text{ev}} = \log \lambda_{li} + \alpha$, as considered in the main text.

Particle-based implementation of the filtering equations. We now analyze the changes of expected probability masses for membrane voltages given by eq. (10) with $b_i^{\text{ev}} = \frac{1}{\tau} + \lambda_{\text{max}} - \lambda_i$ and $w_{il}^{\text{ev}} = \log \lambda_{li} + \alpha$. We show that they are

approximately equal to the changes in the p_i 's in eq. (8). For a given pattern of spikes in layer \mathcal{L}_{dyn} , the firing rates in layer \mathcal{L}_{ev} evaluate to

$$\rho_{\text{ev},i}^m(t) = \left[\frac{1}{M} X_{\text{dyn},i}(t) \left(\frac{1}{\tau} + \lambda_{\text{max}} - \lambda_i + \sum_l w_{il}^{\text{ev}} y_l(t) + g(t) \right) \right]^+ \quad (28)$$

For positive weights, the only term that can make the argument of the $[\cdot]^+$ operator negative is $g(t)$ which will be equated with normalizing inhibition. We will discuss the influence of this effect below. For now, we assume that the membrane potential does not become negative, which enables us to skip the rectification $[\cdot]^+$. We use the shortcut $\rho_{\text{ev},i}(t) = \sum_m \rho_{\text{ev},i}^m(t)$ for the summed firing rate of ensemble $Z_{\text{ev},i}$ to obtain

$$\rho_{\text{ev},i}(t) = X_{\text{dyn},i}(t) \left(\frac{1}{\tau} + \lambda_{\text{max}} - \lambda_i + \sum_l w_{il}^{\text{ev}} y_l(t) + g(t) \right). \quad (29)$$

The change of the expected probability masses in \mathcal{L}_{ev} is given by eq. (2)

$$\begin{aligned} \frac{d}{dt} \langle Z_{\text{ev},i}(t) \rangle &= \langle \rho_{\text{ev},i}(t) \rangle - \langle \rho_{\text{ev},i}(t-\tau) \rangle \\ &= \langle X_{\text{dyn},i}(t) \rangle \left(\sum_l w_{il}^{\text{ev}} y_l(t) + \lambda_{\text{max}} - \lambda_i + g(t) \right) \\ &\quad + \frac{1}{\tau} \langle X_{\text{dyn},i}(t) \rangle - \langle \rho_{\text{ev},i}(t-\tau) \rangle. \end{aligned}$$

Since layer \mathcal{L}_{dyn} copies the distribution represented by \mathcal{L}_{ev} , we have $\langle X_{\text{dyn},i}(t) \rangle \approx \langle Z_{\text{ev},i}(t) \rangle$ which yields

$$\begin{aligned} \frac{d}{dt} \langle Z_{\text{ev},i}(t) \rangle &\approx \langle Z_{\text{ev},i}(t) \rangle \left(\sum_l w_{il}^{\text{ev}} y_l(t) + \lambda_{\text{max}} - \lambda_i + g(t) \right) \\ &\quad + \frac{1}{\tau} \langle Z_{\text{ev},i}(t) \rangle - \langle \rho_{\text{ev},i}(t-\tau) \rangle. \end{aligned}$$

The first term in this equation is equivalent to the change of the p_i 's in eq. (8). The last term is due to EPSPs that end at time t and has to be compensated. It is approximately compensated by the second to last term since $\langle Z_{\text{ev},i}(t) \rangle = \int_{t-\tau}^t \langle \rho_{\text{ev},i}(s) \rangle ds \approx \tau \langle \rho_{\text{ev},i}(t-\tau) \rangle$ under the assumption that the expected firing rate (i.e., the represented probability distribution) changes slowly on the time scale of the EPSP. Hence, we have

$$\frac{d}{dt} \langle Z_{\text{ev},i}(t) \rangle \approx \langle Z_{\text{ev},i}(t) \rangle \left(\sum_l w_{il}^{\text{ev}} y_l(t) + \lambda_{\text{max}} - \lambda_i + g(t) \right).$$

Comparing this result to eq. (8), we can see that the dynamics of the mean probability masses in layer \mathcal{L}_{ev} approximate those needed for evidence integration.

Multiplication through gating of activity. We now show that the multiplication can be approximated by gating of activity.

The membrane potentials of neurons in \mathcal{L}_{ev} given in eq. (10) gives rise to the ensemble firing rates given in eq. (29), which approximate the optimal changes in probability masses as shown above. In a first step, we show that membrane potentials (11) give rise to identical ensemble firing rates. This can easily be seen since for non-negative membrane potentials we have

$$\begin{aligned} \rho_{\text{ev},i}(t) &= \sum_m \rho_{\text{ev},i}^m(t) = \sum_m [u_{\text{ev},i}^m(t)]^+ = \sum_m u_{\text{ev},i}^m(t) \\ &= \sum_m x_{\text{dyn},i}^m(t) \left(\sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} + g(t) \right) \\ &= X_{\text{dyn},i}(t) \left(\sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} + g(t) \right). \end{aligned}$$

The matching of superscripts m in eq. (11) is chosen for notational simplicity. Of course, any permutation of superscripts on the right hand side is valid as well. If the firing rate of neuron $x_{\text{dyn},i}^m$ is low, the probability of two spikes in a time window of size τ is small and we can approximate $x_{\text{dyn},i}^m(t)$ by a binary variable taking on the values 0 or 1. In this case, the multiplication is accomplished by gating of the activity of neuron $z_{\text{ev},i}^m$ by neuron $x_{\text{dyn},i}^m$.

In a second step, we discuss how this gating can be accomplished through disinhibition. Our general model for disinhibition is discussed below. Here we use the special case of a single disinhibiting neuron $I^{\text{dis}}(x_{\text{dyn},i}^m(t)) = I_0$ if $x_{\text{dyn},i}^m(t) = 0$ and 0 otherwise. For membrane potentials given by eq. (12), and binary $x_{\text{dyn},i}^m(t)$, we have

$$\rho_{\text{ev},i}^m(t) = [u_{\text{ev},i}^m(t)]^+ = x_{\text{dyn},i}^m(t) \left(\sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} + g(t) \right) \quad (30)$$

if I_0 is strong, such that the membrane potential becomes negative if neuron $x_{\text{dyn},i}^m$ was not active recently. The ensemble rates

$$\rho_{\text{ev},i}(t) = \sum_m \rho_{\text{ev},i}^m(t) = X_{\text{dyn},i}(t) \left(\sum_l w_{il}^{\text{ev}} y_l(t) + b_i^{\text{ev}} + g(t) \right), \quad (31)$$

are equivalent to the ensemble firing rates of eq. (29) and therefore approximate the optimal changes of probability masses.

Disinhibition

We formally define the influence of disinhibition arising from neurons x_1, \dots, x_l on the membrane potential of a neuron as

$$I^{\text{dis}}(x_1(t), \dots, x_l(t)) = \begin{cases} I_0 & , \text{ if } \sum_{i=1}^l x_i(t) \leq \mathcal{G}^{\text{dis}} \\ 0 & , \text{ otherwise} \end{cases} \quad (32)$$

where I_0 is some baseline inhibition and \mathcal{G}^{dis} is some threshold. In other words, the disinhibited neuron is released from baseline inhibition if the neurons were recently active enough to overcome the threshold \mathcal{G}^{dis} . We used in this article a threshold of $\mathcal{G}^{\text{dis}} = 0.5$. For rectangular EPSPs, this results in disinhibition whenever $\sum_i x_i(t) > 0$. In our circuit model, the baseline inhibition is strong enough to suppress any activity when inhibited.

Lateral inhibition

Lateral inhibition in layer \mathcal{L}_{ev} is given by

$$I^{\text{lat}}(t) = I_0^{\text{lat}} \left[\sum_{i,m} z_{ev,i}^m(t - \Delta_{\text{delay}}) - L \right]^+, \quad (33)$$

It corrects for excessive spiking activity, i.e., when $\sum_i Z_{ev,i}(t - \Delta_{\text{delay}}) > L$. The constant I_0^{lat} determines how strongly this correction influences the membrane potential. We included the $[\cdot]^+$ operator to model a purely inhibitory population. This means that too low firing rates are not compensated by this mechanism. This is possible, since all other contributions to the membrane potential are non-negative if weights are shifted to non-negative values. In particular, the $\frac{1}{\tau}$ term in the biases b_i^{ev} (see Table 1) leads to a quick recovery of the ensemble firing rate.

Strong lateral inhibition at low potentials could be cut off by the $[\cdot]^+$ -operator in equation (28). That would result in unequal inhibition values for different ensembles. This can be avoided by bounding inhibition to a maximum of $\frac{1}{\tau}$ (note that inhibition is bounded due to the finite size of the network). But our simulations indicate that even with larger values, the circuit performs very well.

General details on computer simulations

All simulations were performed in discretized time with a time step of 0.5 ms. The duration of EPSPs was set to $\tau = 20$ ms in all simulations. The delay for lateral inhibition Δ_{delay} was set to 0.5 ms. The initialization of network activity for the particle filter circuits was performed as follows. Let $P(v(0))$ denote the distribution that the network should represent at the beginning of a simulation. Spikes were drawn in both layers and distributed in the time interval $[-\tau, 0]$ such that at time $t=0$, the represented probability $\tilde{P}(v(0))$ represented $P(v(0))$ in the mean. More precisely, a spike of neuron $z_{ev,i}^m$ in $[-\tau, 0]$ was assumed with probability $P(v(0)=i) \frac{M}{L}$. If there was a spike for this neuron, its exact timing was drawn from a uniform distribution in the interval $[-\tau, 0]$. Simulation of the network began at $t=0$ ms where the EPSPs of these spikes were taken into account when computing the membrane potentials of neurons during the initial phase of the simulation. The initially represented distribution $P(v(0))$ was assumed uniform over all states if not otherwise noted.

All simulation code is available online as *Dataset S1*. Details of individual simulations are discussed in the following.

Computer Simulations for task class B: Evidence integration

Figure 5. For panel F, the prior probability for state $P(v(0)=1)$ was chosen in each run randomly from a uniform distribution over $[0.1, 0.9]$. The two afferent neurons y_1 and y_2 spiked at time 20 ms and 25 ms respectively. Each $\lambda_{1,i}$ was drawn from a uniform distribution over $[5, 20]$ Hz. For panel F, 100 simulation runs were performed and $\tilde{P}(v(t_{\text{post}})=1)$ as well as $\tilde{P}(v(t_{\text{post}})=2)$ is shown, with $t_{\text{post}}=45$ ms (P_{post}). The true posterior P_{opt} was computed according to eq. (27). Panels C–E show an example run with $P(v(0)=1)=0.6$ and $\lambda_{1,1}=10$ Hz, $\lambda_{2,1}=15$ Hz, $\lambda_{1,2}=5$ Hz, $\lambda_{2,2}=12$ Hz.

Circuit parameters: Number of neurons per state $M=2000$; target estimation sample size $L=400$; lateral inhibition scaling $I_0^{\text{lat}}=0.5$.

Action readouts from state variables

We provide here details on the readout layer used to model the ambiguous target task and the random-dot motion task. An application of the feedforward circuit discussed in *Simple probabilistic dependencies* is inference over rewarded actions a based on the current belief about the state of a random variable $v(t)$. In the context of reward-based action selection, one wants to estimate the probability that a given action leads to reward (we consider here binary rewards $R \in \{0, 1\}$ for simplicity). If the probability $P(R=1|v, a)$ of reward for action a in state v is known, then the posterior distribution over actions for the current state distribution can be inferred as

$$P(a(t)=i|R=1) \propto \sum_j P(R=1|a=i, v=j) P(v(t)=j), \quad (34)$$

see below. In the ambiguous target task, $P(R=1|v, a)$ is 1 if the action (movement direction) matches the direction of the spatial cue in the color of the color cue, and 0 otherwise. Eq. (34) defines a direct probabilistic relation between the action and the state. Hence, a layer of neurons as described in *Task class A* computes the posterior distribution. We refer to such a layer as an *action readout layer*.

We discuss this now more formally. Consider a random variable v with range $\{1, \dots, N\}$, a random variable a over possible actions $\{1, \dots, N_A\}$, and a binary random variable $R \in \{0, 1\}$ that indicates whether a reward occurs. Assuming that the joint probability $P(a, v, R)$ factorizes to $P(a, v, R) = P(a)P(v)P(R|a, v)$, one can infer future actions a that will lead to a reward for the given distribution over states by:

$$P(a=i|R=1) = \frac{P(a=i, R=1)}{P(R=1)} \propto \sum_j P(R=1|a=i, v=j) P(a=i) P(v=j). \quad (35)$$

We want to infer future actions a that will lead to a reward for the current state $v(t)$. Assuming a uniform prior over actions, this simplifies to

$$P(a=i|R=1) \propto \sum_j P(R|a=i, v=j) P(v(t)=j). \quad (36)$$

Assume that the distribution $P(v(t))$ is given by ensembles Z_j in a sample-based representation with estimation sample size L^Z . This defines an operation as discussed above under *Task class A: Simple probabilistic dependencies*. Hence, a circuit consisting of N_A ensembles and M neurons per ensemble samples from the posterior distribution (36) with estimation sample size $L^X = \alpha L^Z$ if the membrane voltages of the neurons are given by

$$u_i^m(t) = \sum_{j,n} w_{ij}^R z_j^n(t)$$

with weights $w_{ij}^R = \alpha \frac{1}{M\tau} P(R=1|a=i, v=j)$.

The ambiguous target task

Stimuli were coded by 19 ensembles of afferent neurons, where each ensemble consisted of 20 neurons. One ensemble coded for the fixation cross, and two for the two colors of the color cue. Eight

ensembles coded for each position of a red spatial cue and eight for each position of a blue color cue. First, the fixation cross-stimulus was presented for 100 ms. The spatial cue followed for 250 ms. Here, two ensembles of afferent neurons were active, one for each color of the spatial cue. In the 250 ms memory epoch, only the fixation cross was present. After the memory epoch, the color cue appeared, and the corresponding ensemble of afferent neurons (for red or blue color cue) was active. Afferent neurons spiked at a baseline rate of 0.1 Hz and at a rate of 5 Hz when they were active.

For synaptic weights from afferent neurons, we used $\lambda_{ij} = 1.25 \text{ Hz}$ for observations l that are possible in state i . For example, in a state coding for “movement to 90 degrees with red color cue”, a red spatial cue at 90 degrees, a blue spatial cue at 270 degrees, and a fixation cross are possible. All other λ_{ij} were set to the baseline firing rate of 0.1 Hz. The lateral inhibition scaling was $I_0^{\text{lat}} = 1.25$.

The action readout layer consisted of 100 neurons for each of the 8 possible actions (movement to one of eight directions). Considering eq. (4), we identified the random variable $v^Z(t)$ with the random variable estimated in the particle filter circuit. The hidden states of this variable are the direction-color pairs (d, c) for 8 possible directions d and 2 colors c . The random variable $v^X(t)$ has 8 states encoding that movement in direction d leads to a reward. This random variable can be computed through marginalization over color c . Hence, the conditional probabilities were given by $P(v^X = d | v^Z = (d', c)) = 1$ for $d = d'$ and $P(v^X = d | v^Z = (d', c)) = 0$ otherwise. With these conditionals, we obtained the synaptic weights through eq. (6) with $\alpha = 0.2$, leading to an estimation sample size of 80 in the action readout layer.

For Figure 1, spiking activity was averaged over 50 successful trials (87 successful trials out of 100), temporally (10 ms running average) and spatially (mean activity of 10 neighboring neurons for neurons ordered by their preferred direction).

Continuous time Markov chains

A continuous time Markov chain is characterized by *transition rates* q_{ij} for each pair of distinct states, i.e., for $i, j \in \{1, \dots, N\}$ and $i \neq j$. Assume that at some time t' the state is $v(t') = i$. To sample a sequence of states from the Markov chain starting at t' , a Poisson process \mathcal{P}_j with rate q_{ij} is started for each state $j \neq i$. The time of the next state transition is given by the first event in these processes and the next state is given by the index of the process that produced the event. After a transition, the chain starts afresh at the transition time.

Task class C: Bayesian filtering

We show that the membrane potentials given in Table 1 approximate the set of differential equations (13). These equations can be formulated as two sets of coupled equations, one for evidence integration and one for prediction of dynamics:

$$\frac{d}{dt} p_i^{\text{ev}}(t) = p_i^{\text{dyn}}(t) \left[\sum_l (\lambda_{li} - 1) \hat{y}_l(t) - \lambda_i \right] \quad (37)$$

$$\frac{d}{dt} p_i^{\text{dyn}}(t) = \sum_{j \neq i} q_{ji} p_j^{\text{ev}}(t) - p_i^{\text{ev}}(t) \sum_{j \neq i} q_{ij}, \quad (38)$$

where we assume identical initial conditions for p_i^{ev} and p_i^{dyn} . p_i^{ev} is represented by \mathcal{L}_{ev} and the change of the expected probability masses corresponds to the changes in eq. (37) as shown above.

We show here that the membrane potentials given in eq. (14) lead to changes in the expected probability masses of \mathcal{L}_{dyn} that approximate those of eq. (38). Assume that the membrane potentials evolve according to eq. (14) with weights $w_{ij}^{\text{dyn}} = \frac{q_{ji}}{M}$ where we defined $q_{ii} \equiv \frac{1}{\tau} - \sum_{j \neq i} q_{ij}$. For given spikes in layer \mathcal{L}_{ev} , the summed firing rates $\rho_{\text{dyn},i}(t) = \sum_n \rho_{\text{dyn},i}^n(t)$ in layer \mathcal{L}_{dyn} evaluate to

$$\begin{aligned} \rho_{\text{dyn},i}(t) &= M \sum_j w_{ij}^{\text{dyn}} Z_{\text{ev},j}(t) \\ &= \sum_{j \neq i} q_{ji} Z_{\text{ev},j}(t) + \frac{1}{\tau} Z_{\text{ev},i}(t) - Z_{\text{ev},i}(t) \sum_{j \neq i} q_{ij}. \end{aligned}$$

The change of the expected probability masses in \mathcal{L}_{dyn} is

$$\begin{aligned} \frac{d}{dt} \langle X_{\text{dyn},i}(t) \rangle &= \langle \rho_{\text{dyn},i}(t) \rangle - \langle \rho_{\text{dyn},i}(t - \tau) \rangle \\ &= \sum_{j \neq i} q_{ji} \langle Z_{\text{ev},j}(t) \rangle - \langle Z_{\text{ev},i}(t) \rangle \sum_{j \neq i} q_{ij} \\ &\quad + \frac{1}{\tau} \langle Z_{\text{ev},i}(t) \rangle - \langle \rho_{\text{dyn},i}(t - \tau) \rangle. \end{aligned}$$

The last term is due to EPSPs that end at time t which needs to be compensated. It is approximately compensated by the second to last term since $\langle Z_{\text{ev},i}(t) \rangle \approx \langle X_{\text{dyn},i}(t) \rangle = \int_{t-\tau}^t \langle \rho_{\text{dyn},i}(s) \rangle ds \approx \tau \langle \rho_{\text{dyn},i}(t - \tau) \rangle$ under the assumption that the expected firing rate (i.e., the represented probability distribution) changes slowly on the time scale of the EPSP. Altogether we obtain

$$\frac{d}{dt} \langle X_{\text{dyn},i}(t) \rangle \approx \sum_{j \neq i} q_{ji} \langle Z_{\text{ev},j}(t) \rangle - \langle Z_{\text{ev},i}(t) \rangle \sum_{j \neq i} q_{ij}$$

as needed.

Computer Simulations for task class C: Bayesian filtering

Figure 7. Parameters for the example run, panels C,D: $q_{12} = 23.4 \text{ Hz}$; $P(v(0) = 1) = 0.16$. The true posterior was computed analytically as

$$P(v(t) = 1) = P(v(0) = 1) e^{-q_{12} t}.$$

Circuit parameters were chosen identical to those for Figure 5: Number of neurons per state $M = 2000$; target estimation sample size $L = 400$; lateral inhibition scaling $I_0^{\text{lat}} = 0.5$.

Particle filtering in a generic setup for task class C. State sequences were generated according to the HMM described in the main text. Once a new state was drawn, it was maintained for at least 60 ms. Point-event observations were produced with rates

$$\lambda_{ij} = 50 \frac{\lambda'_{ji}}{\sum_j \lambda'_{ij}} + 0.1,$$

where λ'_{ij} were set according to Gaussian tuning functions with means μ_j and standard deviation $\sigma_j = 2.5$ for all j

$$\lambda'_{ij} = \exp\left(-\frac{(i-\mu_j)^2}{2\sigma_j^2}\right). \quad (39)$$

The means were set to $\mu_1 = 10$, $\mu_2 = 15$, $\mu_3 = 16$, $\mu_4 = 20$, $\mu_5 = 25$. Weights from evidence neurons to the evidence layer were set according to these observation rates.

Circuit parameters: Number of neurons per state $M = 2000$; target estimation sample size $L = 400$; lateral inhibition scaling $I_0^{\text{lat}} = 2.5$. The circuit was initialized at $t = 0$ with the prior distribution $P(v(0) = 1) = 0.8$ and $P(v(0) = j) = 0.05$ for $j = 2, \dots, 5$.

Panel Af: All simulations were performed in discrete time with a discretization time step of 0.5 ms. The optimal Bayesian filtering was performed by implementing the filtering equations (37) – (38). The optimal estimate based on the most recent observation was computed in the following way: If there was no observation at time t , the estimate was identical to the estimate at time $t - 1$. If there was an observation $y(t)$ at time t , then the distribution over states was given by the posterior $P(v(t)|y(t))$. The errors were computed as follows: At each discrete time step, the maximum over the posterior distribution was taken as the predicted state $\tilde{v}(t)$. The error was then computed as the fraction of incorrect predicted states $\frac{1}{T} \sum_{n=1}^T |\tilde{v}(n\Delta t) - v(n\Delta t)|$, where T is the number of time steps of the simulation run and Δt is the discretization time step.

Computer Simulations for task class D: Context-dependent Bayesian filtering

Particle filtering in a generic setup for task class D.

State sequences were generated according to the HMM described in the main text, where state transition rates were chosen according to the current context. The HMM started in context A and context was switched whenever state 1 was entered. Point-event observations were produced as in *Particle filtering in a generic setup for task class C*, but with different tuning functions. The tuning functions for states 1 to 3 were Gaussians with λ_{ij} defined as in eq. (39) with means $\mu_1 = 26.25$, $\mu_2 = 8.75$, $\mu_3 = 26.25$ and standard deviations $\sigma_1 = 2.5$, $\sigma_2 = 2.5$, $\sigma_3 = 5$. Note that the tuning curves for states 1 and 3 differed only in their standard deviations, but not in their means. The tuning for state 4 was given by a sum of two Gaussians with $\sigma_4 = 5$ with an additive offset

$$\lambda'_{i4} = \exp\left(-\frac{(i-8.75)^2}{2\sigma_4^2}\right) + \exp\left(-\frac{(i-26.25)^2}{2\sigma_4^2}\right) + 1.$$

The tuning for state 5 was uniform $\lambda'_{i5} = 1$. Weights from evidence neurons to the evidence layer were set according to these observation rates.

Circuit parameters were identical to those in *Bayesian filtering in a generic setup*. 10 context neurons were used per context. They produced Poisson spike trains with rate 50 Hz when active and no spikes otherwise.

Panel Bf: Simulations were performed and errors were computed as described in *Particle filtering in a generic setup for task class C*. For the optimal context-dependent Bayesian filtering (“opt”), the optimal transition rates were used for the current context. For the optimal filtering without context (“mix”), the weights to layer \mathcal{L}_{dyn} were set according to the mixed transition rates $q_{ij}^{\text{mixed}} = 0.5(q_{ij}^{\text{A}} + q_{ij}^{\text{B}})$, where q_{ij}^{A} and q_{ij}^{B} are the transition rates for context A and B respectively.

Self-localization. For the full maze, we used coordinates between 0 and 1 in each dimension. Locations of variables were uniformly spaced on a 10×10 grid in the maze (see Figure 10A). We simulated an agent that navigated in the continuous space. Let $(x(t), y(t))$ denote the location of the agent at time t . All afferent neurons had a baseline firing rate of 0.1 Hz. The firing rate of each afferent neuron was given by a Gaussian with a corresponding center, an STD of 0.1, and a maximum rate of 50 Hz. The lower four afferent neurons in Figure 10A, had identical positional tuning in both chambers. Their firing rate at each time t was determined by the Gaussian in the chamber where the agent was currently situated. For $y(t) > 0.75$, all afferent neurons spiked at baseline.

Circuit parameters: The network size N and estimation sample size L was 250; $I_0^{\text{lat}} = 4$. 10 context neurons were used per context ensemble, each producing a 50 Hz Poisson spike train in its context and no spikes if the context did not match. The weights from evidence neurons to neurons in \mathcal{L}_{ev} were set according to the rates of the afferent neurons when the agent would be exactly at the place of the corresponding state. The weights to neurons in layer \mathcal{L}_{dyn} were set according to transition rates as follows. There were four groups of populations in this layer, one for each context. Transition rates were 3.5 Hz for possible movements in the direction that corresponded to the context of the population (dark gray arrows in Figure 10A). Transition rates between other states with adjacent positions (horizontally, vertically, or diagonal; states separated by the chamber wall were considered non-adjacent) were 0.1 Hz (light gray lines in Figure 10A). Transition rates between remaining states were 0 Hz.

Task class E: Graded integration of uncertain context information

In task class E, it is advantageous to deal with context in a graded manner. This is achieved by a simple modification the context-dependent filtering circuit from task class D. In the modified circuit, the membrane potential of neurons in \mathcal{L}_{dyn} is given by

$$u_{\text{dyn},i,k}^n(t) = \sum_{j,m} w_{ij}^{\text{dyn},k} z_{\text{ev},j}^m(t) - I^{\text{dis}}(c_k^n(t))$$

where $c_k^n(t)$ denotes the EPSP-filtered spike train of the n^{th} neuron in the k^{th} ensemble that represents the distribution over contexts. In comparison to the equation given in Table 2, each neuron in \mathcal{L}_{dyn} is disinhibited by a single context neuron. This leads to an approximate linear mixture of the context-dependent transition rates

$$q_{ij}(t) \approx \sum_k \tilde{P}(c(t) = k | \mathbf{Y}(t)) q_{ij}^k,$$

where $q_{ij}(t)$ is the effective transition rate for the context-dependent filtering circuit at time t and $\tilde{P}(c(t) = k | \mathbf{Y}(t))$ is the estimated posterior for the context variable at time t .

Computer simulations for task class E: The ambiguous target task revisited

State sequences and spike trains of evidence neurons were generated as in the simulation for the ambiguous target task, with the exception that the fixation phase lasted for 250 ms. Circuit parameters for the context-dependent filtering of $v(t)$ and the action readout layer were identical to those in the simulation for the ambiguous target task (1000 neurons per state; estimation sample size 400; $I_0^{\text{lat}} = 1.25$). In the fixation-context, transition rates between all pairs of states were 1 Hz.

A particle filter circuit (without context) was employed to estimate $c(t)$, the current phase of the trial. The basic parameters for this circuit were identical to the parameters of the context-dependent filtering circuit (1000 neurons per state; estimation sample size 400; $I_0^{\text{lat}} = 1.25$). Other parameters for this circuit were as follows. Synaptic weights to \mathcal{L}_{dyn} were set according to transition rates that were assumed to be 5 Hz between states where a transition is possible and 0 Hz between other states. Synaptic weights from evidence neurons to the evidence layer were set according to the following observation rates (see also Figure 11). A fixation cross in the CHT state or in the MEM state: 5 Hz; Any spatial cue in the SC state: 5/16 Hz; Any color cue in the CC state: 2.5 Hz; For other observation rates, a baseline of 0.1 Hz was assumed.

Supporting Information

Dataset S1 Matlab source files for all simulations.
(ZIP)

References

- Gold JJ, Shadlen MN (2007) The neural basis of decision making. *Annu Rev Neurosci* 30: 535–574.
- Yang T, Shadlen MN (2007) Probabilistic reasoning by neurons. *Nature* 447: 1075–1080.
- Cisek P, Kalaska JF (2005) Neural correlates of reaching decisions in dorsal premotor cortex: specification of multiple direction choices and final selection of action. *Neuron* 45: 801–814.
- 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: e1002211.
- Shadlen MN, Kiani R, Hanks TD, Churchland AK (2008) Neurobiology of decision making: An intentional framework. In: *Better than conscious? Decision-making, the human mind, and implications for institutions*, MIT-Press, chapter 4, pp.71–101.
- Murphy KP (2012) *Machine learning: a probabilistic perspective*. MIT press.
- Doucet A, Godsill S, Andrieu C (2000) On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing* 10: 197–208.
- Grimmett G, Stirzaker D (2001) *Probability and random processes*. Oxford Univ Press, 3rd edition.
- Peccevi D, Buesing L, Maass W (2011) Probabilistic inference in general graphical models through sampling in stochastic networks of spiking neurons. *PLoS Comput Biol* 7: e1002294.
- Williams SR, Stuart GJ (2002) Dependence of EPSP efficacy on synapse location in neocortical pyramidal neurons. *Science* 295: 1907–1910.
- Jolivet R, Rauch A, Lüscher HR, Gerstner W (2006) Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *J Comput Neurosci* 21: 35–49.
- Mensi S, Naud R, Gerstner W (2011) From stochastic nonlinear integrate-and-fire to generalized linear models. In: *Adv Neural Inf Process Syst*, volume 24, pp.1377–1385.
- Brémaud P (1981) *Point processes and queues*, volume 30. Springer.
- Bobrowski O, Meir R, Eldar YC (2009) Bayesian filtering in spiking neural networks: Noise, adaptation, and multisensory integration. *Neural Comput* 21: 1277–1320.
- Rao RP (2005) Hierarchical Bayesian inference in networks of spiking neurons. In: *Adv Neural Inf Process Syst*. MIT Press, volume 17, pp.1113–1120.
- Gaines B (1969) Stochastic computing systems. In: *Advances in information systems science*, Springer, pp.37–172.
- Srinivasan MV, Bernard GD (1976) A proposed mechanism for multiplication of neural signals. *Biol Cybern* 21: 227–236.
- Katz PS (2003) Synaptic gating: the potential to open closed doors. *Current Biol* 13: R554–R556.
- Gisiger T, Boukadoom M (2011) Mechanisms gating the flow of information in the cortex: What they might look like and what their uses may be. *Front Comput Neurosci* 5.
- Letzkus JJ, Wolff SB, Meyer EM, Tovote P, Courtin J, et al. (2011) A disinhibitory microcircuit for associative fear learning in the auditory cortex. *Nature* 480: 331–335.
- Douglas RJ, Martin KA (2004) Neuronal circuits of the neocortex. *Annu Rev Neurosci* 27: 419–451.
- Ecker AS, Berens P, Keliris GA, Bethge M, Logothetis NK, et al. (2010) Decorrelated neuronal firing in cortical microcircuits. *Science* 327: 584–587.
- Fino E, Yuste R (2011) Dense inhibitory connectivity in neocortex. *Neuron* 69: 1188–1203.
- Ma WJ, Beck JM, Latham PE, Pouget A (2006) Bayesian inference with probabilistic population codes. *Nat Neurosci* 9: 1432–1438.
- Pastor-Bernier A, Cisek P (2011) Neural correlates of biased competition in premotor cortex. *J Neurosci* 31: 7083–7088.
- Shadlen MN, Gold JJ (2005) The neurophysiology of decision-making as a window on cognition. In: *The cognitive neurosciences*, MIT Press, 3rd edition, pp.1229–1241.
- Churchland AK, Kiani R, Chaudhuri R, Wang XJ, Pouget A, et al. (2011) Variance as a signature of neural computations during decision making. *Neuron* 69: 818–831.
- Jiang X, Wang G, Lee AJ, Stornetta RL, Zhu JJ (2013) The organization of two new cortical interneuronal circuits. *Nat Neurosci* 16: 210–218.
- Pi HJ, Hangya B, Kvitsiani D, Sanders JI, Huang ZJ, et al. (2013) Cortical interneurons that specialize in disinhibitory control. *Nature* 503: 521–524.
- Lee TS, Mumford D (2003) Hierarchical bayesian inference in the visual cortex. *J Opt Soc Am A Opt Image Sci Vis* 20: 1434–1448.
- Haider B, Häusser M, Carandini M (2013) Inhibition dominates sensory responses in the awake cortex. *Nature* 493: 97–100.
- Okun M, Lampl I (2008) Instantaneous correlation of excitation and inhibition during ongoing and sensory-evoked activities. *Nat Neurosci* 11: 535–537.
- Rigotti M, Barak O, Warden MR, Wang XJ, Daw ND, et al. (2013) The importance of mixed selectivity in complex cognitive tasks. *Nature* 497: 585–590.
- Habenschuss S, Puhf H, Maass W (2013) Emergence of optimal decoding of population codes through STDP. *Neural Comput* 25: 1371–1407.
- Nessler B, Pfeiffer M, Buesing L, Maass W (2013) Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity. *PLoS Comp Biol* 9: e1003037.
- Kappel D, Nessler B, Maass W (2014) STDP installs in winner-take-all circuits an online approximation to hidden markov model learning. *PLoS Comp Biol* 10: e1003511.
- Rezende DJ, Gerstner W (2014) Stochastic variational learning in recurrent spiking networks. *Front Comput Neurosci* 8.
- Jin DZ, Fujii N, Graybiel AM (2009) Neural representation of time in cortico-basal ganglia circuits. *Proc Natl Acad Sci USA* 106: 19156–19161.
- Asaad WF, Rainer G, Miller EK (2000) Task-specific neural activity in the primate prefrontal cortex. *J Neurophysiol* 84: 451–459.
- Maunsell JH, Treue S (2006) Feature-based attention in visual cortex. *Trends Neurosci* 29: 317–322.
- Saalmann YB, Pigarev IN, Vidyasagar TR (2007) Neural mechanisms of visual attention: how top-down feedback highlights relevant locations. *Science* 316: 1612–1615.
- Cohen MR, Newsome WT (2008) Context-dependent changes in functional circuitry in visual area MT. *Neuron* 60: 162–173.
- Skaggs WE, McNaughton BL (1998) Spatial firing properties of hippocampal CA1 populations in an environment containing two visually identical regions. *J Neurosci* 18: 8455–8466.
- Chen G, King JA, Burgess N, O’Keefe J (2013) How vision and movement combine in the hippocampal place code. *Proc Natl Acad Sci USA* 110: 378–383.
- Beck JM, Ma WJ, Kiani R, Hanks T, Churchland AK, et al. (2008) Probabilistic population codes for Bayesian decision making. *Neuron* 60: 1142–1152.
- Lochmann T, Denève S (2011) Neural processing as causal inference. *Current opinion in neurobiology* 21: 774–781.
- Denève S (2008) Bayesian spiking neurons I: Inference. *Neural Comput* 20: 91–117.

Author Contributions

Conceived and designed the experiments: RL WM. Performed the experiments: RL. Analyzed the data: RL WM. Wrote the paper: RL WM.

48. Boerlin M, Denève S (2011) Spike-based population coding and working memory. *PLoS Comp Biol* 7: e1001080.
49. Rao RP (2004) Bayesian computation in recurrent neural circuits. *Neural Comput* 16: 1–38.
50. Beck JM, Pouget A (2007) Exact inferences in a neural implementation of a hidden markov model. *Neural Comput* 19: 1344–1361.
51. Bean MA (2001) Probability: the science of uncertainty with applications to investments, insurance, and engineering, volume 6. American Mathematical Soc.
52. Cisek P (2006) Integrated neural processes for defining potential actions and deciding between them: a computational model. *J Neurosci* 26: 9761–9770.