Master Thesis

Approximate Inference in Spiking Neural Networks

Kjartan van Driel

First Supervisor:
Dr. G.J. Stephens
Second Supervisor:
Dr. C. de Mulatier

External Supervisor:
Dr. Viola Priesemann
Daily Supervisors:
Fabian Mikulasch
Lucas Rudelt

VU VRIJE UNIVERSITEIT AMSTERDAM

UNIVERSITEIT VAN AMSTERDAM

MAX PLANCK INSTITUTE
FOR DYNAMICS AND SELF-ORGANIZATION

Contents

Abstract

Investigation of brain functions, especially its capacity to learn and adapt to environmental stimuli, is a central theme in theoretical neuroscience. One prevalent theory, the Bayesian brain hypothesis, proposes that one of the brain's primary functions is to interpret the states of the world through the principles of Bayesian inference. However, there exists a notable gap between the theoretical formulation of such hypotheses and their relation to the complex neurobiology of the brain.

This work attempts to take a small step in closing the gap between theory and biology by extending the Spike by Spike (SbS) framework. The Spike by Spike framework is able to derive deterministic spiking neural networks from the principle of efficiently encoding a stimulus. We show that one can extend the SbS framework and derive stochastic spiking neural networks from the principle of performing approximate Bayesian inference.

Our extended framework thereby provides a novel perspective between the principles of Bayesian inference and its potential biological implementation by spiking neurons. Furthermore, the framework supplies solid scaffolding to test theories surrounding the Bayesian brain hypothesis. Additionally, we demonstrate the ability of small spiking neural networks to perform simple inference tasks. We discuss the relation of our extended framework to other prominent Bayesian-inspired theories of brain function, such as predictive coding. And finally, we discuss its potential relationship to experimental studies, its current limitations as well as possible directions for future research.

Introduction

In the exploration of natural phenomena, science has always exhibited a fondness for principles. It likes to reduce the apparent complexity of the universe into digestible laws and axioms, frameworks with which we can make sense of the chaos. Indeed, we observe this fondness most strikingly in the field of physics, where all the grandeur of the universe is distilled into elegant principles. Principles such as Newton's laws of motion and Einstein's theory of relativity have formed the bedrock upon which our understanding of the physical universe rests.

When we turn our gaze to biology, however, we are confronted with a seemingly different picture. Biology, with its abundant complexity and intricate networks of interactions, often resembles a messy, undecipherable soup rather than an orderly system governed by clear principles. It is a science of exceptions, of diversity, and variation, which at first glance, seems resistant to the kind of simplifying principles we find in physics.

But to hold onto this impression would be overly simplistic. Biology, particularly through the lens of evolution, is not devoid of underlying principles. Darwin's theory of natural selection proposes that species adapt and optimize in response to their environment. Through this lens, the biology of a species can be interpreted as the result of a complex optimization problem, with survival and reproduction within an environment as its main objective. This reasoning suggests that even in the apparent chaos of biology, a species will gradually find a way to survive in the most efficient manner possible.

Neuroscience still has a rich phenomenology that is left unexplained, and we lack satisfying scaffolding to understand these phenomena. It is here that a principled approach could prove insightful. A major success of such an approach can be found in the work of Olshausen and Field [53]. Their research demonstrated how the formation of receptive fields (the stimulus to which a neuron is most responsive) in the primary visual cortex can be derived by combining knowledge about its function with a principle of efficiency. They, therefore, supply us with a clear theoretical lens through which we can understand a piece of the brain.

For other efforts, we can turn to the fascinating topic of plasticity: how do neurons change their connections, and in service of what, if any, objective do they adapt? Neuroscience has a rich history of proposals to explain plasticity. From the famous work of the Hopfield network [33], a model of associative memory, to the more recent theories around predictive coding [57, 51, 26], which aims to show how the brain could form a representation of a noisy world. Plasticity becomes clearly interpretable in these frameworks, as it is clear what the adaptation aims to accomplish.

Another recent, relatively unexplored effort is given by the Spike by Spike (SbS) framework [17]. At its core, the framework derives biologically plausible spiking neural networks from the principle of efficiently representing a stimulus. A unique feature of the framework is that it respects the characteristic ability of a neuron to spike, which is often neglected in theoretical neuroscience. It is respect for this feature that makes

the framework particularly unique and well-suited to contextualize not only plasticity but more general spiking neural network phenomenology, as has been shown in [11, 12, 15, 16, 17, 62, 48, 18, 4]

This work aims to provide an extension to the SbS framework. We show that one can derive stochastic spiking neural networks from a general, and natural objective of performing approximate Bayesian inference. Bayesian Inference is often heralded as one of the ultimate objectives of the brain, so much so that it inspired the popular Bayesian brain hypothesis [38, 39, 20]. We hope that this work will provide further theoretical foundation to better understand the function and form of spiking neural networks. Particularly we hope that our work, as an extension to the SbS framework, might offer a stomping ground to test ideas surrounding the Bayesian brain hypothesis. The work will consist of four main sections.

- The first will provide a background on several general concepts in mathematics and neuroscience. The section will introduce our notation, discuss a theorist's perspective on neurobiology, introduce central principles in theoretical neuroscience and conclude with a small explanation of variational inference. To read the text well, it is expected that the background is familiar to the reader.

- The second section will provide an introduction to the SbS framework. We will start by introducing the conceptual ideas behind the framework, and illustrate them by deriving a single neuron performing the task of tracking a stimulus over time. Then, we will show how these ideas generalize to networks of neurons. Note that these developments are recent and that to our knowledge no conceptual introductory work exists; as such we took much creative liberty in its exposition.

- The third section contains our main theoretical results, deriving an extension of the SbS framework. We show how the objective of performing probabilistic inference with spiking networks can naturally lead to networks with similar architecture to the SbS framework. To this end, we start by building the conceptual and theoretical framework, analogously to our presentation of SbS. We show, with the help of numerical methods, the efficacy of our derived spiking neural networks in performing simple inference tasks. The section concludes by discussing the relation between our framework and that of the prominent theory of predictive coding.

- In the fourth and final section we will discuss our theoretical results. Our extended framework is still in its infancy and we leave many questions unanswered. Here we discuss the relation of our framework to the Bayesian brain hypothesis, as well as its potential relevance to experiments. We continue to discuss its current limitations and sketch an outlook for further research.

1 Background - General

The present work is situated within the highly interdisciplinary domain of neuroscience. It is a conceptual and theoretical piece of work that incorporates various concepts and ideas from mathematics and physics, along with a broad understanding of neuroscience concepts. The target audience for this thesis is physics students, who may not possess extensive knowledge in these areas. Hence, we provide the necessary background information, notational conventions, and general concepts.

1.1 Notation & methods

The current investigation is situated within an emerging and expansive realm of theoretical neuroscience, where standardized notation and methodology are still lacking. As a result, there is a need to develop succinct and intelligible notation. We have made significant efforts to accomplish this task to the best of our ability. Here we briefly note our notational conventions:

Linear algebra. We denote vectors by bold lowercase letters like $\mathbf{x}, \mathbf{u}, \mathbf{s}$, and their components as x^i, where the index is in superscript. Similarly for matrices with bold capital letters $\mathbf{D}, \mathbf{W}$, and components again with superscript D^{ij}. To write basis vectors we use the notation $\mathbf{e}^i$, and write $\mathbf{D}^i = \mathbf{D}\mathbf{e}^i$ for the projection of $\mathbf{D}$ onto the basis vector $\mathbf{e}^i$.

Time denotation. All of our considerations will be made for systems described in discrete time. To facilitate ease of notation we denote individual points in time t by integers so that notation like $\sum_t$ and $\sum_{t'<t}$ makes immediate sense. Even though we work in discrete time with integer notation, our dynamics will still have a timescale. As such we have specified the time elapsed between two time-steps, which we denote by δt.

Scalars and vectors that vary with time are frequently encountered, and we denote them using a subscript to indicate the specific time, such as $\mathbf{x}_t$. Additionally, we come across functions that depend on time, and we represent them as $\kappa(t)$, where the timescale δt is implicit within the function. For instance, we can have a function defined as $\kappa(t) = \exp\left(-t\frac{\delta t}{\tau}\right)$, where τ denotes the real time-scale of the system, and δt indicates the timescale of the discretization.

Probability theory. Our results are generally in a probabilistic setting, and we should clarify our notation precisely. We write probability distributions generally with a letter p, so that $p(\mathbf{x})$ is then the distribution over $\mathbf{x}$. Similarly, we write $\mathbf{x} \sim p(\mathbf{x})$ to denote that $\mathbf{x}$ is drawn from the distribution $p(\mathbf{x})$ and denote $p(\mathbf{x} \mid \mathbf{c})$ to be the distribution of $\mathbf{x}$ 'conditioned on', or 'given' $\mathbf{c}$.

For expectation values we use bracket notation common in physics. To denote the expectation value of $f(\mathbf{x})$ over the distribution $p(\mathbf{x})$ we write $\langle f(\mathbf{x})\rangle_{p(\mathbf{x})}$.

In general we hope the context will make it clear which distributions are being considered, but there are few which we make very explicit. We use $p_\Theta(\mathbf{x}, \mathbf{c})$, with subscript Θ, to denote the model assumed to generate the observations $\mathbf{x}$ with underlying causes $\mathbf{c}$. The model $p_\Theta(\mathbf{x}, \mathbf{c})$ can be called the world model, or generative model. We typically factor the world model $p_\Theta(\mathbf{x}, \mathbf{c})$ as $p_\Theta(\mathbf{x}, \mathbf{c}) = p_\Theta(\mathbf{x} \mid \mathbf{c})p_\Theta(\mathbf{c})$. The factor $p_\Theta(\mathbf{x} \mid \mathbf{c})$ is called the observation model, or as is common in the literature, the likelihood given data. The other factor $p_\Theta(\mathbf{c})$ is called the prior over causes. We use $p_\Psi(\mathbf{s} \mid \mathbf{x})$, with subscript Ψ to denote the distribution of the brain. The distribution of the brain describes the activity $\mathbf{s}$ given observations $\mathbf{x}$.

It is common for us to write distributions using proportionality notation $\propto$, as it enables us to succinctly ignore the normalization constant of probability distributions. For instance, in the context of physics we can write a Boltzmann distribution with energy $E(\mathbf{x})$ and (inverse) temperature β as

$$p(\mathbf{x}) \propto \exp(-\beta E(\mathbf{x})),$$

which would be equivalent to

$$p(\mathbf{x}) = \frac{\exp(-\beta E(\mathbf{x}))}{\sum_{\mathbf{x}'} \exp(-\beta E(\mathbf{x}'))}.$$

While this does not work in general, with either distributions over a non-finite sample space, we ignore these subtleties in this work.

Bayesian inference. Bayesian inference describes how one should integrate new observations with prior beliefs. Mathematically, Bayesian inference requires a few key elements: a prior belief over the causes $\mathbf{c}$, denoted as $p(\mathbf{c})$, and a model of observations $\mathbf{x}$ given causes $\mathbf{c}$, given as $p(\mathbf{x} \mid \mathbf{c})$. With these two elements, we can compute a posterior distribution of the causes post observation, expressed as $p(\mathbf{c} \mid \mathbf{x})$. Bayes theorem expresses the relation between the prior $p(\mathbf{c})$, observation model $p(\mathbf{x} \mid \mathbf{c})$ and posterior $p(\mathbf{c} \mid \mathbf{x})$ as

$$\begin{aligned} p(\mathbf{c} \mid \mathbf{x}) &\propto p(\mathbf{x}, \mathbf{c}) && (1) \\ &\propto p(\mathbf{x} \mid \mathbf{c})p(\mathbf{c}). && (2) \end{aligned}$$

Functionals. Some common objects in our work will be functionals (functions of functions) over probability distributions. We denote the arguments of a functional by square brackets, so that the functional $F[f(\mathbf{x})]$, or equivalently $F[f]$ takes as argument the function $f(\mathbf{x})$.

Derivatives. We write derivatives in the Leibniz notation, $\frac{d}{dx}$ for full derivatives and $\frac{\partial}{\partial x}$ for partial derivatives.

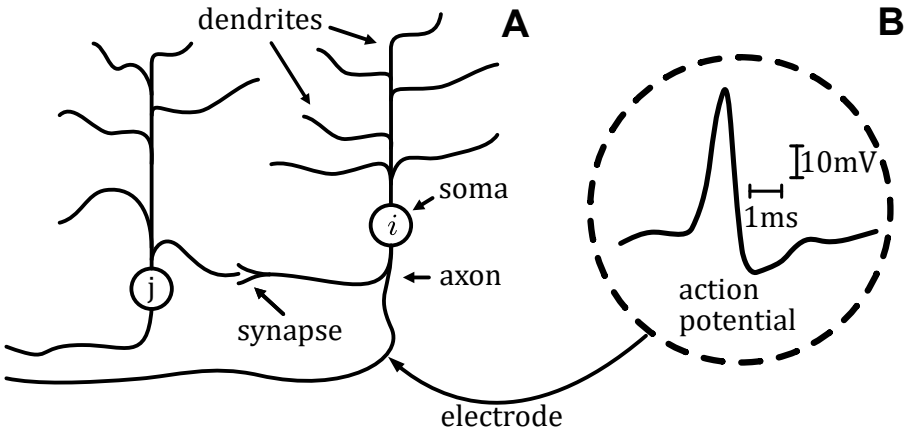

Figure 1.1: Illustration of a neuron and an action potential. **A**: The neuron features a central soma (cell body) with numerous dendrites (signal receivers) at the top and an axon (signal transmitter) at the bottom, and their intersection called synapse. **B**: An illustration of the typical shape and scales of an action potential (neural response or 'spike'), which is generated in the soma, travels along the axon, reaches a synapse and subsequently travels to the soma of another neuron, depicted as a process from neuron i to neuron j.

1.2 Elements of Computational Neuroscience

This section serves as an introduction to the basic concepts relevant to computational neuroscience. Particularly we will introduce the basic biology and phenomenology of the neuron, the basic building block of the brain. We will build on these basic notions to motivate a simple model of neural dynamics, which will serve as a reference anchor to neurobiology for the rest of our work. This chapter builds heavily on the introductory chapters of the work by Gerstner [31], which contains an excellent introduction to models used in computational neuroscience.

1.2.1 Neural phenomenology

The ideal neuron. The brain is a massively complex and heterogeneous system, consisting of numerous cell types and countless interactions between them. The past century has supplied us with enormous knowledge about the intricate details of these different cell-types and interactions. The bulk of the cells present in the brain are of a particular type called neurons. Neurons are the computational building blocks of the brain. While neurons come in countless shapes and sizes, their basic features remain relatively constant across the brain.

A typical neuron has three different components that generally perform distinct functions. These components are called the dendrites, the soma, and the axon, as shown in figure 1.1**A**. Roughly, the dendrites serve as the input to the neuron, they transfer signals from other neurons to its own main cell body, called the soma. The soma is seen as the main computational unit of the neuron, it combines the incoming signals from

the dendrites and performs a (non-linear) computation. The soma sums the incoming stimuli, and when the sum exceeds a threshold, an output signal is generated called an action potential. The output signal is then transferred along the axon onto other neurons.

The intersection point where the axon of one cell meets the dendrite of another, is called a synapse. For this reason the sending cell is called the pre-synaptic cell, and the receiving cell the post-synaptic cell.

Spikes and neural dynamics. The signals generated by neurons are short electrical pulses. These characteristic short pulses go by the names of neural responses, action potentials or spikes for short. Spikes can be observed by placing an electrode along the axon of a neuron as shown in figure 1.1**B**. Remarkably, the exact shape of a spike will remain conserved over a wide range of conditions surrounding the neuron. Therefore, as the shape of the pulse is assumed to be absent of information itself, the information is implied to be contained in the exact timing and number of spikes.

The effect of a single spike on a receiving (post-synaptic) neuron can be measured by the effect it has on the potential difference u_t between the inside of a neuron and its surroundings. This potential difference u_t is called the membrane potential, which when undisturbed will sit at a resting level of u_{rest}. The resting potential is typically around -65 mV, and as such, the resting neuron is said to be polarized.

We can classify the effect of a spike in two categories. If the spike has the effect of raising the membrane potential, we call it excitatory. Similarly, if the spike has the effect of lowering the membrane potential, we call it inhibitory.

To be more precise, we can measure the precise effect of a spike at time t_0 from pre-synaptic neuron j to post-synaptic neuron i at rest by considering

$$u_t^i - u_{\text{rest}} := \varepsilon^{ij}(t - t_0).$$

The kernel ε^{ij} on right hand side defines the post-synaptic potential (PSP) of neuron i after the incidence of a spike from neuron j. Following our classification of spikes, if $\varepsilon^{ij} > 0$ we say that the neuron induces a excitatory post-synaptic potential (EPSP) and if $\varepsilon^{ij} < 0$ we say that the neuron induces a inhibitory postsynaptic potential (IPSP).

To first approximation, as long as there are not too many spikes, the total membrane potential can be described by a linear sum of the PSPs. We can express this in discrete time as

$$u_t^i = \sum_j \sum_{t'<t} s_{t'}^j \varepsilon^{ij}(t - t') + u_{\text{rest}},$$

where the sum over j ranges over the various neurons connecting to neuron i, t' ranges over all time less than t, and $s_{t'}^j \in \{0, 1\}$ indicates whether neuron j spiked at time t'.

The linearity, however, will break down significantly as the membrane potential u_t^i reaches a critical threshold ϑ^i. At this point the membrane potential of the neuron will exhibit a short pulse-like excursion. This pulse is exactly the spike which will travel along the axon to other neurons in the network. The pulse will quickly depolarize (raise)

the membrane potential, and then decrease to an extended period of hyper-polarization (lowering) of the membrane potential.

The refractory period of hyper-polarization briefly makes further elicitation of spikes nearly impossible. The refractory effect can be included if we add an initially strongly-negative refractory kernel η^i to the membrane potential

$$u_t^i = \sum_{t'<t} s_{t'}^i \eta^i(t-t') + \sum_j \sum_{t'<t} s_{t'}^j \varepsilon^{ij}(t-t') + u_{\text{rest}}.$$

which concludes a basic treatment of the inter-neuron interactions.

1.2.2 The spike response model (SRM): a benchmark

The considerations of the previous section can be used to construct a phenomenological model of neural dynamics called the spike response model (SRM).[30, 31] The SRM is a widely studied model of neural dynamics and this work will use it as a benchmark for neuro-plausibility. We continue by concluding our phenomenological considerations and defining the SRM.

We construct the SRM by noting that in addition to neuron to neuron interactions, there are various other sources affecting the membrane potential. We can abstract these other sources by introducing a term that integrates a generic current I_t^i. We obtain the definition of the SRM membrane potential u_t^i

$$u_t^i = \underbrace{\sum_{t'<t} s_{t'}^i \eta^i(t-t')}_{\text{self}} + \underbrace{\sum_{t'<t} \sum_j s_{t'}^j \varepsilon^{ij}(t-t')}_{\text{inter-neuron}} + \underbrace{\sum_{t'<t} \kappa^i(t-t') I_{t'}^i}_{\text{external}} + u_{\text{rest}}, \tag{3}$$

with threshold condition for the deterministic SRM

$$s_t^i = \begin{cases} 1, & \text{if } u_t^i > \vartheta^i \\ 0, & \text{otherwise} \end{cases}, \tag{4}$$

or for the stochastic SRM, also called the the spike response model with escape noise, we have

$$p_\Psi(s_t^i = 1) = \delta t \ \rho(u_t^i - \vartheta^i) \tag{5}$$

$$p_\Psi(s_t^i = 0) = 1 - \delta t \ \rho(u_t^i - \vartheta^i), \tag{6}$$

where instead of a sharp threshold, the spiking probability of the network is determined by a firing rate $\rho(u_t^i - \vartheta^i)$, with discretization timescale δt. The firing rate $\rho(u_t^i - \vartheta^i)$ is also called the instantaneous spiking intensity, or the escape rate and can be experimentally measured.

It has been shown that the SRM with exponential escape rate $\rho(u_t^i - \vartheta^i) = \exp(u_t^i - \vartheta^i)$ is in excellent agreement with experimental data as shown by Jovilet et al. [35] Furthermore, it has been shown that many well studied spiking models can be mapped

with great accuracy to the SRM with exponential escape rate. [47, 31] We therefore deem the SRM, particularly the SRM with exponential escape rate, a good benchmark for bio-plausible dynamics.

The only caveat which we have yet to mention is Dale's law. Experimentally we observe that most neurons illicit only one type of post-synaptic potential, either excitatory on inhibitory, and this observation is known as Dale's law. While our benchmark does not require this, we note that the literature shows that one can adapt networks to respect Dale's law [17].

1.3 Central ideas of theoretical neuroscience

This section is dedicated to briefly explain some key conceptual ideas in contemporary neuroscience. These concepts are the backbone of our work, underpinning the methods, and interpretations that we will delve into in subsequent sections.

1.3.1 Representing uncertainty

In our complex and noisy world, uncertainty surrounds our beliefs and actions. Remarkably, both humans and animals have demonstrated the capacity to acknowledge and integrate this uncertainty [60]. The evidence that humans and animals represent uncertainty about observations during perceptual processes raises questions about its possible neural representation [56].

From a mathematical perspective, uncertainty is naturally embedded into probability distributions, thereby providing a formal perspective to representing uncertainty. Building upon this formal perspective, the neuroscience literature has proposed two main schemes to represent uncertainty in the brain [25]. The first scheme suggests that neural activity represents the parameters of a distribution [44]. The second scheme, on the other hand, suggests that neural activity represents the stochastic quantities themselves [24].

Both the schemes offer unique perspectives and could potentially co-exist or be used in different circumstances. Although much remains to be discovered about the brain's preferred representation of uncertainty, current research into the benefits and drawbacks of these two schemes is an active and exciting area of research [40].

1.3.2 The Bayesian brain hypothesis

We previously discussed the notion that the brain deals with uncertainty in observations and therefore necessarily constructs a representation of this uncertainty. This representation is particularly important for the brain to make sense of its experiences and to anticipate future ones. Indeed, it must somehow utilize its representation to model how prior observations relate to present and future ones.

A particularly appealing proposal of how the brain integrates past and present observations is given by the Bayesian brain hypothesis. [38, 39, 20] The Bayesian brain

hypothesis posits that the brain integrates past and present observations in a particular manner, according to the principles of Bayesian inference.

To understand the practical potency of the Bayesian brain hypothesis, we turn to a seemingly odd example: the Apollo moon landing. The Apollo mission required an efficient method of estimating and predicting the spacecraft's position and velocity based on incoming data. The solution was the Kalman filter, an application of Bayesian inference. [46] Despite being one of the simplest applications of Bayesian inference, the Kalman filter proved more than sufficient. This approach ensured a successful and safe mission, proving the value of Bayesian inference in high-stakes, real-world applications.

Though compelling, the Bayesian brain hypothesis poses substantial challenges for empirical validation. While there is a consensus that the brain forms a representation of uncertainty, it remains less clear whether this representation represents some kind of Bayesian computation. The flexibility inherent in Bayesian inference allows one to potentially select a model in line with the experimental data and post-hoc justify the brain's implementation of Bayesian inference [29].

In summary, the Bayesian brain hypothesis presents an exciting, if complex, approach to understanding how our brains interpret and predict the world around us. Given the challenges associated with empirically validating this hypothesis, future research must focus on developing rigorous methods to discern what constitutes a Bayesian computation.

1.3.3 The Kalman filter

To better illustrate the previous two sections we present an example that showcases the discussed uncertainty representations and the principle of Bayesian inference.

Our example will be that of a Kalman filter [36], one of the simplest examples of Bayesian inference. Remarkably, it was a variation of this simple example that was used in the Apollo mission [46].

The Kalman filter is the result of Bayesian inference with a simple model that relates two quantities. For each point in time t we have an observation x_t, and some latent structure c_t that is assumed to give rise to that observation. The model assumes a relation $p(x_t \mid c_t)$ between c_t and x_t called the observation model, stating that x_t is normally distributed around c_t with variance σ_x^2. Similarly the model assumes a relation $p(c_t \mid c_{t-1})$, called the dynamical model, stating that c_t is normally distributed around its past value c_{t-1} with variance σ_c^2. And finally the model assumes a prior $p(c_{t-1})$ on the previous value of the latent cause c_{t-1}.

First we use the Kalman filter to illustrate a representation in terms of parameters. We assume that the prior $p(c_{t-1})$ is given as a normal distribution with mean μ_{t-1} and variance σ_{t-1}^2. The normal assumptions of the model and prior allow us to explicitly

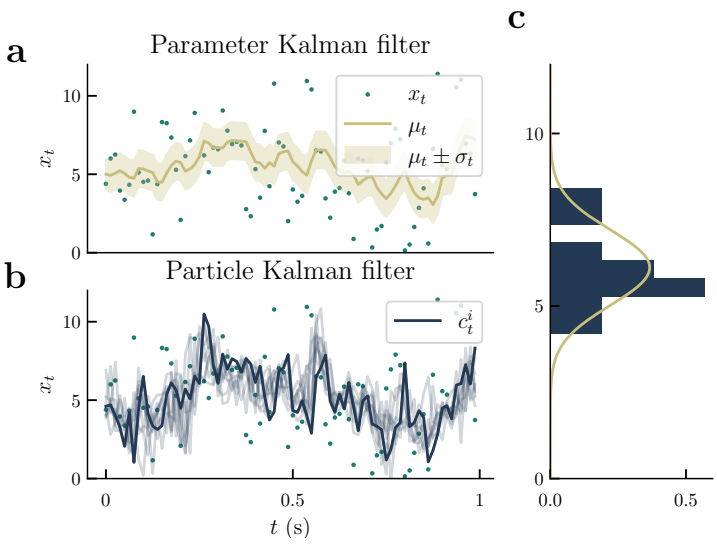

Figure 1.2: An illustration of inference based on a parameter based representation of uncertainty, as opposed to a sampling based representation of uncertainty. **a**: a trajectory of μ_t and σ_t of a parameter based Kalman filter, inferring the noisy signal x_t. **b**: 10 sample trajectories c_t^i of a particle Kalman filter inferring the noisy signal x_t. **c**: histogram and parameterized normal distribution at time $t = 1$. The curve depicts a normal distribution parameterized by μ_t and σ_t at time $t = 1$. The histogram depicts the distribution of the 10 particles at time $t = 1$ showing a reasonable agreement between the two methods.

calculate the posterior $p(c_t \mid x_t)$

$$p(c_t \mid x_t) \propto p(c_t \mid x_t) \propto p(x_t \mid c_t)p(c_t \mid c_{t-1})p(c_{t-1}) \tag{7}$$

$$\propto \exp\left(-\frac{1}{2\sigma_t^2}\left(c_t - \mu_t\right)^2\right), \tag{8}$$

with the complicated but easily computable expressions

$$\mu_t = \frac{\sigma_{t-1}^2 + \sigma_c^2}{\sigma_{t-1}^2 + \sigma_x^2 + \sigma_c^2}\mu_{t-1} + \frac{\sigma_x^2}{\sigma_{t-1}^2 + \sigma_x^2 + \sigma_c^2}x_t \quad \text{and} \quad \sigma_t^2 = \frac{\sigma_x^2(\sigma_{t-1}^2 + \sigma_c^2)}{.} \tag{9}$$

The posterior reflects our updated belief about the cause and will in turn be used as a prior in the next time-step. The result of such a recursive procedure is shown in figure 1.2**a**

Second, we illustrate the representation using a sample-based representation called a particle filter [58]. The key idea is that a distribution $p(c_{t-1})$ can be approximated by a set of samples c_{t-1}^i called particles as $p(c_{t-1}) \approx \frac{1}{n}\sum_i^n \delta(c_{t-1} - c_{t-1}^i)$, which collectively form a histogram. Here, i ranges from 1 to the total number of particles n. Moreover, it is possible to calculate the approximate posterior for a histogram by sampling from the the posterior for each individual particle. The resulting histogram then serves as an approximation of the posterior distribution.

We continue to calculate the posterior $p(c_t^i)$ for a single particle c_t^i and assume that the prior $p(c_{t-1}^i)$ is given by the sample as $p(c_{t-1}) = \delta(c_{t-1} - c_{t-1}^i)$, where δ is the Dirac delta. Next we explicitly calculate the posterior for a single particle

$$p(c_t^i \mid x_t) \propto p(x_t \mid c_t^i) p(c_t^i \mid c_{t-1}) \delta(c_{t-1} - c_{t-1}^i) \tag{10}$$

$$\propto \exp\left(-\frac{(\sigma_x^2 + \sigma_c^2)}{2\sigma_x^2 \sigma_c^2} \left(c_t - \frac{\sigma_c^2}{\sigma_x^2 + \sigma_c^2} x_t - \frac{\sigma_x^2}{\sigma_x^2 + \sigma_c^2} c_{t-1}^i \right)^2 \right). \tag{11}$$

Repeating this procedure for each particle grants us a histogram approximation of the posterior. The result of recursively applying the particle filter at each time step is shown in figure 1.2**b**.

The connection of both the regular and the particle Kalman filter to Bayesian inference and representations of uncertainty is so tight, that researchers have used both as analogies to the functioning of the brain. [51, 42]

1.4 Variational Inference

The following section will contain an introduction to variational inference [9, 32].

In the general setting we have a model description $p_\Theta(\mathbf{x}, \mathbf{c})$, given as a distribution over observations $\mathbf{x}$ and latent causes $\mathbf{c}$. One would like to use observations to either improve the model, or to infer the latent structure behind the observations, i.e. one desires to know the posterior $p_\Theta(\mathbf{c} \mid \mathbf{x})$. These problems only admit tractable solutions in rare cases and we therefore require approximation methods.

Variational inference describes a broad array of methods to solve exactly these problems. Here one usually starts with a family of distributions $p_\Phi(\mathbf{c} \mid \mathbf{x})$ parameterized by Φ. The methods of variational inference prescribe how one can adjust the parameters Φ so that $p_\Phi(\mathbf{c} \mid \mathbf{x})$ better resembles $p_\Theta(\mathbf{c} \mid \mathbf{x})$. And one can in turn use this approximation to improve the general model p_Θ to explain observations.

This method has special appeal for theoretical neuroscientists as many hold that the brain performs some kind approximative Bayesian inference. From this perspective, part of our brain is dedicated to implement an approximative distribution, attempting to perform inference of some kind of model of the world.

This view has become especially prominent when it was shown that the predictive coding framework of Rao and Ballard [57] can be derived from the assumption that the brain implements a particular kind of variational inference [26, 27, 28]. These efforts are hugely prominent as a top-down approach to brain function [52].

Principles of variational inference. Suppose we have model of the world $p_\Theta(\mathbf{x}, \mathbf{c})$ that we believe is sufficient to explain observations $\mathbf{x}$. Furthermore, suppose that the model factors into an observation model given causes $p_\Theta(\mathbf{x} \mid \mathbf{c})$ and a prior on the latent causes $p_\Theta(\mathbf{c})$. We can interpret the observation model as projecting the causes onto observations, and the prior as our belief about the causes.

We desire to have a **good** model of the world. Remember that $p_\Theta(\mathbf{x}, \mathbf{c})$ defines how we think the observations could be generated, but the description or parameters might be off. What happens if the general form of $p_\Theta(\mathbf{x}, \mathbf{c})$ is sufficient but the parameters are not? In that case we want to use our observations to improve our model. To see how we can accomplish this, note that by definition the observations are generated by the world, i.e. $\mathbf{x} \sim p_\gamma(\mathbf{x})$. Where p_γ denotes the actual, unknown generative distribution. The log-likelihood $\ln p_\Theta(\mathbf{x})$ satisfies the following relation with the world $p_\gamma(\mathbf{x})$:

$$\langle \ln p_\Theta(\mathbf{x}) \rangle_{p_\gamma} = -H[p_\gamma] - D_{KL}[p_\gamma \,||\, p_\Theta] \tag{12}$$

.

This is the first principle. To elaborate: as the entropy $H(p_\gamma)$ is independent of the model p_Θ, the maximisation of the (expected) log-likelihood of our model is equivalent to the minimisation of the KL-divergence between the model p_Θ and the world p_γ. Therefore the optimization of the expected model log-likelihood will improve the ability of the model to explain the observations provided by the environment.

As such, a major assumption of variational inference is that one actually has access to the expected log-likelihood. We continue under this premise and assume that we can approximate the expectation value well by an average over observations, as is common in any method aiming to do maximum-likelihood estimation.

Above we considered the log-likelihood of the data $\mathbf{x}$, however, our model also considers the latent causes $\mathbf{c}$. To improve the model we require access to the log-likelihood $\ln p_\Theta(\mathbf{x})$ and therefore need to integrate out $\mathbf{c}$, or equivalently, we require access to the posterior. So either

$$\ln p_\Theta(\mathbf{x}) = \ln \sum_{\mathbf{c}} p_\Theta(\mathbf{x}, \mathbf{c}), \tag{13}$$

or by Bayes rule,

$$\ln p_\Theta(\mathbf{x}) = \ln \frac{p_\Theta(\mathbf{x}, \mathbf{c})}{p_\Theta(\mathbf{c} \mid \mathbf{x})}. \tag{14}$$

We proceed our derivation from the latter (eq 14), which we rewrite

$$\ln p_\Theta(\mathbf{x}) = \ln \frac{p_\Theta(\mathbf{x}, \mathbf{c})}{p_\Theta(\mathbf{c} \mid \mathbf{x})} = \ln \frac{p_\Theta(\mathbf{x}, \mathbf{c})}{p_\Phi(\mathbf{c} \mid \mathbf{x})} + \ln \frac{p_\Phi(\mathbf{c} \mid \mathbf{x})}{p_\Theta(\mathbf{c} \mid \mathbf{x})}. \tag{15}$$

By taking the expectation value with respect to $p_\Phi(\mathbf{c} \mid \mathbf{x})$ we obtain the central relation of variational inference

$$\begin{aligned} \ln p_\Theta(\mathbf{x}) &= \left\langle \ln \frac{p_\Phi(\mathbf{c} \mid \mathbf{x})}{p_\Theta(\mathbf{c} \mid \mathbf{x})} \right\rangle_{p_\Phi(\mathbf{c}|\mathbf{x})} + \left\langle \ln \frac{p_\Theta(\mathbf{x}, \mathbf{c})}{p_\Phi(\mathbf{c} \mid \mathbf{x})} \right\rangle_{p_\Phi(\mathbf{c}|\mathbf{x})} \\ &= D_{\mathrm{KL}} \left[p_\Phi(\mathbf{c} \mid \mathbf{x}) \parallel p_\Theta(\mathbf{c} \mid \mathbf{x}) \right] - \mathcal{F}[p_\Phi, p_\Theta], \end{aligned} \tag{16}$$

with

$$\mathcal{F}[p_\Phi, p_\Theta] := \langle -\ln p_\Theta(\mathbf{x}, \mathbf{c}) \rangle_{p_\Phi(\mathbf{c}|\mathbf{x})} - \langle -\ln p_\Phi(\mathbf{c} \mid \mathbf{x}) \rangle_{p_\Phi(\mathbf{c}|\mathbf{x})}. \tag{17}$$

We have decomposed the model log-likelihood $\ln p_\Theta(\mathbf{x})$ into two objects: The first is a KL-divergence between the exact posterior $p_\Phi(\mathbf{c} \mid \mathbf{x})$ and our approximation $p_\Theta(\mathbf{c} \mid \mathbf{x})$. We note again that the KL-divergence is positive and a measure of dissimilarity between distributions. We do not have direct access to the exact posterior and as such we cannot compute the KL-divergence directly.

The second quantity $\mathcal{F}[p_\Phi, p_\Theta]$ is known as the (negative) evidence lower bound (ELBO) or variational free energy (VFE). Let us explain its name and utility. The name variational free energy comes from its functional form. $\mathcal{F}$ is the difference between an energy term $\langle -\ln p_\Theta(\mathbf{x}, \mathbf{c}) \rangle$ and an entropy term $\langle -\ln p_\Phi(\mathbf{c} \mid \mathbf{x}) \rangle$, which in physics is commonly known as the variational free-energy. We will not elaborate on these connections but they are plentiful, and we highly recommend [45] for further reading.

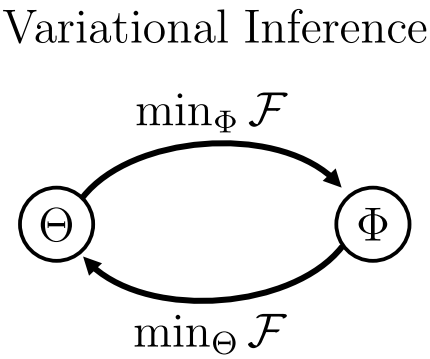

Figure 1.3: Cartoon depicting the tandem optimization underlying variational inference. First we optimize the free energy $\mathcal{F}[p_\Phi, p_\Theta]$ with respect to Φ to improve our approximative distribution. Then we optimize the free energy $\mathcal{F}[p_\Phi, p_\Theta]$ with respect to Θ to improve our model's ability to explain observations.

To continue, the positivity of the KL-divergence term ensures the inequality

$$-\mathcal{F}[p_\Phi, p_\Theta] \leq \ln p_\Theta(\mathbf{x}), \tag{18}$$

which motivates the term evidence lower bound, as in the context of Bayesian inference, the log-likelihood is also known as the model evidence [34]. As mentioned before, maximisation of the log-likelihood with respect to Θ ensures an improvement of the generative model $p_\Theta(\mathbf{x}, \mathbf{c})$ in explaining the data. The inequality states that decreasing $\mathcal{F}$ increases a lower bound of the likelihood. While the increase of a lower bound does not strictly imply an increase of the log-likelihood, we can improve the tightness by minimizing the KL-divergence.

Furthermore, as the log-likelihood does not depend on our approximation, we obtain by (16) that minimization of $\mathcal{F}$ is equivalent to minimization of the KL-divergence. This implies that minimization of $\mathcal{F}$ with respect to the approximation parameters Φ is equivalent to reducing the KL-divergence. To illustrate this, we consider the gradient of $\mathcal{F}$ with respect to Φ and obtain

$$\nabla_\Phi \mathcal{F}[p_\Phi, p_\Theta] = \nabla_\Phi D_{\mathrm{KL}} \left[p_\Phi(\mathbf{c} \mid \mathbf{x}) \parallel p_\Theta(\mathbf{c} \mid \mathbf{x}) \right].$$

Lastly and of great importance: $\mathcal{F}$ consists of objects that, as a modeller, we can explicitly access, namely the total model $p_\Theta(\mathbf{x}, \mathbf{c})$ and the approximation distribution $p_\Phi(\mathbf{c} \mid \mathbf{x})$. Furthermore, one can proof that by taking samples $\mathbf{c} \sim p_\Phi(\mathbf{c} \mid \mathbf{x})$, we actually gain an unbiased estimate of $\mathcal{F}$. So in short: we can calculate $\mathcal{F}$ effectively.

The repeated tandem optimisation, depicted in figure 1.3, of $\mathcal{F}$ with respect to Θ and Φ ensures both a better model and a better posterior approximation. It is this tandem which generally defines variational inference.

Relation to MAP estimation. One of the main purposes of variational inference is to circumvent the issue of tractability in calculating the posterior distribution.

Another, much simpler approach is that of maximum a posteriori (MAP) estimation [22]. Here we briefly explain MAP estimation and illustrate how it can be seen as a special case of variational inference.

MAP estimation is a simple technique of approximating the posterior $p_\Theta(\mathbf{c} \mid \mathbf{x})$ by Dirac delta distribution $\delta(\mathbf{c} - \mathbf{c}_{\text{MAP}})$ of its most likely value

$$\mathbf{c}_{\text{MAP}} = \max_{\mathbf{c}} p_\Theta(\mathbf{x}, \mathbf{c}). \tag{19}$$

In the continuous case, one attempts to find $\mathbf{c}_{\text{MAP}}$ by following the gradient $\frac{\partial}{\partial \mathbf{c}} p_\Theta(\mathbf{x}, \mathbf{c})$ of the probability distribution.

We can draw the parallel to variational inference by choosing the approximation distribution to be a Dirac delta

$$p_\Phi(\mathbf{c} \mid \mathbf{x}) = \delta(\mathbf{c} - \boldsymbol{\phi}), \tag{20}$$

where $\boldsymbol{\phi} \in \Phi$ parametrizes the center of the Dirac delta.

In this case the free energy simplifies as the entropy of a Dirac delta is assumed to be zero[1], and the expectation value simplifies as

$$\mathcal{F}(p_\Phi, p_\Theta) = -p_\Theta(\mathbf{x}, \boldsymbol{\phi}). \tag{21}$$

We conclude the minimization of the free energy with respect to $\boldsymbol{\phi}$ to be equivalent to MAP estimation. Moreover, we can interpret the gradient of the free energy with respect to this distribution as a method to perform approximative MAP estimation.

[1]This is actually quite complicated, but it can be neglected for our purposes, as the entropy does not vary on a change of $\boldsymbol{\phi}$

2 Background - Spike by Spike (SbS) framework

In this section we introduce the Spike by Spike (SbS) framework, the conceptual precursor to our research. The SbS framework conceptualizes a method to derive spiking neural networks from a global objective. The framework does not have a particular name in the literature, and so we took the liberty to name it after the recent publication "Learning to Represent Signals Spike by Spike".[17]

We start by illustrating the framework for just a single neuron. Furthermore, we assume that the optimisation is done greedily, implying that we only consider the current time. After illustrating the key idea we illustrate various avenues of possibility for extending the method, such as to a network of neurons.

It is important to mention that although this is an introductory piece, it is not merely a replication of existing literature. Mathematically, all of the results in this section are present in existing literature, however, the perspective and conceptual emphasis is novel.

2.1 A one-dimensional deterministic world

The SbS framework is a top-down theory of neural function. That is, it derives a neural network from an appropriate objective, which we desire our neurons to fulfill. Here we consider the simple of objective of forming a representation $\hat{x}_t$ of a one-dimensional stimulus x_t. We encode this objective by the square error

$$L = (x_t - \hat{x}_t)^2. \tag{22}$$

Currently, the loss is agnostic to the activity s_t of our neuron, and to relate the loss to the activity, we have to specify exactly how the representation is formed. We follow the literature [17] and suppose the representation $\hat{x}_t$ at time t is given as some leaky readout current which in discrete time can be written as

$$\hat{x}_t = \alpha\ \hat{x}_{t-1} + D\ s_t, \tag{23}$$

where $\alpha = \exp\left(-\frac{\delta t}{\tau}\right) \in (0,1)$ is the decay constant of the leaky current with δt the timescale of our discretization, τ the time-scale of the leaky current, D a constant determining the scaling of the representation, and $s_t \in \{0,1\}$ the activity of our neuron at time t.

Equivalently, by collapsing the recursion we can write the representation as:

$$\hat{x}_t = D \sum_{t' \leq t} s_{t'} \kappa(t - t') = D\left(r_t + s_t\right), \tag{24}$$

with κ a 'response' kernel given by $\kappa(t) = \alpha^t = \exp\left(-\frac{\delta t}{\tau} t\right)$, and $r_t := \sum_{t'<t} s_{t'} \kappa(t - t')$ is the sum of past responses. The shape of the response kernel κ and an example linear readout $\hat{x}_t$ are shown in figure (2.1). We prefer the description of our representation in terms of kernels, mirroring its use in the SRM.

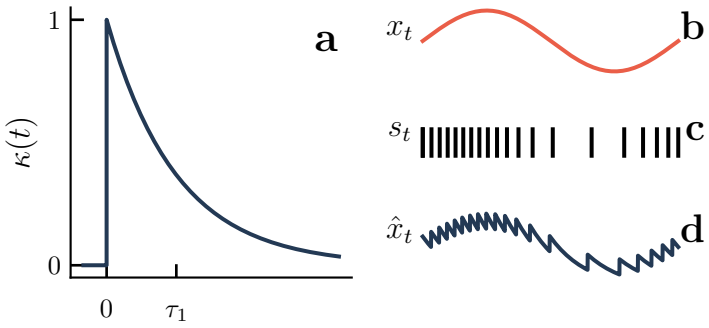

Figure 2.1: An example representation of a stimulus. **a**: the exponential neural response kernel κ. **b**: the stimulus x_t the neuron aims to encode. **c**: spiking activity s_t of the neuron representing x_t. **d**: the representation $\hat{x}_t$ resulting from the activity s_t and kernel κ

We note that the neural code is made explicit before we even begin to consider dynamics. This is the top-down perspective: we have fully defined an objective in terms of neural activity, but have not defined the dynamics of our neuron.

To continue we will derive dynamics from our loss function L and representation. We derive the dynamics by assuming that the neuron will behave in a manner that best accomplishes the objective, as measured by L. We suppose a spiking condition which states the neuron should spike if that would reduce the current loss L at time t

$$\text{spike at time } t \iff L(x_t, s_t = 1) < L(x_t, s_t = 0). \tag{25}$$

The spiking condition (25) is a form of greedy optimization, as we do not consider the impact of our spike on the future loss.

It is worth noting that the spiking condition (25) is analogous to deriving dynamics by following a gradient, a method that is more common in top-down theories of neural function, such as the predictive coding framework [57, 27].

We can continue and write out the spiking condition for the square loss defined before (22) and obtain the following:

$$\text{spike at time } t \iff \underbrace{x_t - D\, r_t}_{\text{error}} > \frac{D}{2}, \tag{26}$$

which simply states that we should spike when the current reconstruction error, without considering the spike, exceeds half the effect of a single spike.

As the representation $\hat{x}$ can be written in terms of a sum over past neural responses, we obtain the elegant form

$$\underbrace{x_t - D \sum_{t'<t} s_{t'} \kappa(t - t')}_{u_t} > \underbrace{\frac{D}{2}}_{\vartheta}. \tag{27}$$

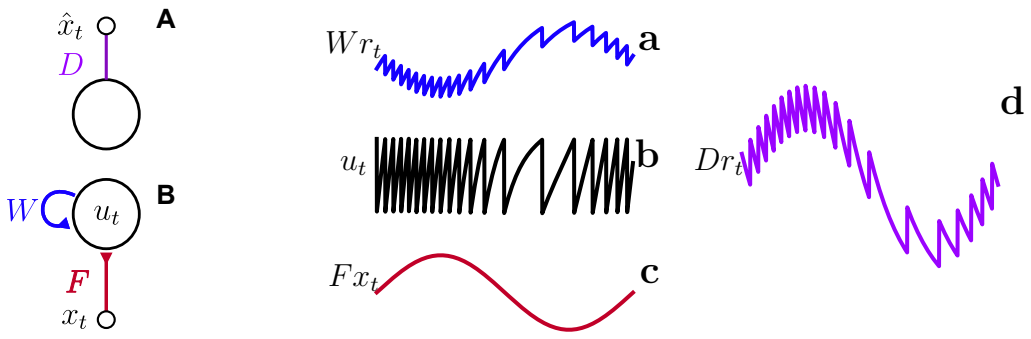

Figure 2.2: *Left*: resulting network architecture from the functional objective of forming a representation $\hat{x}_t$ of a stimulus x_t. **A**: Illustration of the objective, which is agnostic to the possible dynamics of the neuron. **B**: Illustration of the resulting SRM derived by imposing a spiking condition. The feed-forward connection $F = 1$ and lateral connection $W = -D$ are derived from a spiking condition imposing local optimization. *Right*: Network dynamics of a single neuron tracking a simple one-dimensional stimulus. **a-d** shows the inhibitory current Wr_t of the neuron balancing the excitation Fx_t on the membrane potential u_t ensuring a good representation $\hat{x}_t = Dr_t$. Deterministic spiking model following a noisy stimulus. We notice that it traces the envelope of the stimulus as much as the spikes permit.

The left hand side can be interpreted as a membrane potential and the right hand side is then simply a threshold.

Indeed, the derived dynamics can be neatly mapped to the SRM (1.2.2), our benchmark for bio-plausibility. We illustrate the derived network architecture and an example of its dynamics in in figure 2.2.

The membrane potential of our neuron consists of an excitatory current x_t given by the input stimulus, and a self-inhibitory current Dr_t. The balance between the stimulus x_t and the current representation Dr_t explicitly encodes the error between the stimulus and the representation, and therefore the degree to which the neuron performs the objective is coded onto its membrane potential. This balance between excitation and inhibition is also experimentally observed and known as Excitation-Inhibition (E-I) balance [23]. As such, the SbS framework supplies us with an intuition of why this phenomenon may occur.

Metabolic costs. We have shown how we can derive a bio-plausible SRM by defining a functional objective, a representation, and subsequently imposing a spiking condition based on optimally. However, the brain has to balance performing a function with the metabolic cost of performing that function. Indeed, the utility of any ability needs to be weighed against the cost of performing it.

We show that one can easily include metabolic costs in the SbS framework. To illustrate this we will say that there is a cost $C(s_t)$ associated to spiking at time t, here we simply assume a fixed cost ν associated to a single spike. Next we redefine the loss

function L as a sum of two objectives

$$L = (x_t - \hat{x}_t)^2 + \nu s_t = \underbrace{F}_{\text{function}} + \underbrace{C}_{\text{metabolic cost}}, \tag{28}$$

where F quantifies function, and C quantifies the metabolic cost.

With the new definition of the loss L we can interpret the spiking condition (25) as imposing a balance between fulfilling the functional objective and the metabolic costs associated with performing that function. Indeed for this loss we observe that our threshold of spiking will simply increase $\vartheta \to \vartheta + \frac{\nu}{D}$ indicating that the neuron should not follow the signal perfectly, as to not expend too many resources.

To summarize, we have seen that we can derive bio-plausible spiking neuron in a principled manner. We achieved this by first defining a loss function L in terms of a functional objective F, a representation $\hat{x}$ and metabolic cost C on that representation. Subsequently, we derived dynamics requiring that a neuron should spike if that decreases the loss at any point in time. Conceptually, it is this procedure which, to us, defines the SbS framework.

2.2 A multi-dimensional deterministic world

We continue by showing how to extend the previous section to describe a network of neurons. Here we consider how a population of neurons can represent a stimulus of arbitrary dimension. In analogy to the previous section, we define the functional part of our loss to be the square distance between the stimulus and our representation.

$$F = \frac{1}{2}\|\mathbf{x}_t - \hat{\mathbf{x}}_t\|^2, \tag{29}$$

where $\mathbf{x}_t$ now represents the vector valued stimulus, $\hat{\mathbf{x}}_t$ the vector valued representation.

We proceed analogously as before by defining the representation as a linear combination of neural responses of a population of neurons.

$$\hat{\mathbf{x}}_t = \alpha\ \hat{\mathbf{x}}_{t-1} + \mathbf{D}\ \mathbf{s}_t, \tag{30}$$

with α as before the decay constant of all the leaky currents and $\mathbf{D}$ now the decoder matrix, as it describes how one should decode the neural responses to retrieve the representation. Note that for simplicity we choose α to be the same for each part of the representation, though this constraint is easily loosened.

We again collapse the recursion and obtain for $\hat{\mathbf{x}}_t$

$$\hat{\mathbf{x}}_t = \mathbf{D}\sum_{t' \leq t} \mathbf{s}_{t'}\kappa(t - t') = \mathbf{D}\left(\mathbf{r}_t + \mathbf{s}_t\right), \tag{31}$$

with κ again being defined as $\kappa(t) = \alpha^t$, and we define $\mathbf{r} = \sum_{t'<t} \mathbf{s}_{t'}\kappa(t - t')$. The vector $\mathbf{r}$ is conveniently defined as such because it signifies the effect that the past neural responses have on the representation, and thus later, the spiking condition.

Lastly, the metabolic cost is again defined such that it penalizes the spiking of a neuron with weight $\boldsymbol{\nu}$.

$$C(\mathbf{s}_t) := \boldsymbol{\nu} \cdot \mathbf{s}_t. \tag{32}$$

So that finally our objective loss for the network becomes

$$L = \frac{1}{2}\|\mathbf{x}_t - \hat{\mathbf{x}}_t\|^2 + \boldsymbol{\nu} \cdot \mathbf{s}_t. \tag{33}$$

Extending the spiking condition. In the single neuron case, the spiking condition defining our neural dynamics was the result of optimization with respect to a single variable s_t. However, as we now consider a network of neurons, we are faced with a choice. Should we optimize each neuron individually, or should we consider the entire network as a whole? Individual optimization would suggest the same spiking condition as before, under the assumption that the rest of the network stays silent. Global optimization on the other hand, would prescribe us to choose, at any point in time, the activity which minimizes the loss.

global optimisation	**local optimisation**	
$\mathbf{s}_t = \arg\min_{\tilde{\mathbf{s}}_t} L(\tilde{\mathbf{s}}_t, \mathbf{x}_t)$	$L(\mathbf{s}_t = \mathbf{e}^i, \mathbf{x}_t) < L(\mathbf{s}_t = \mathbf{0}, \mathbf{x}_t)$	(34)

where $\mathbf{e}^i$ denotes the i-th basis vector.

The difference between global and local optimization in equation (34) is analogous to the difference between a gradient, and a derivative. The gradient considers the direction of steepest ascend for the system as whole, whereas the derivative only considers an individual component.

We have not encountered this consideration explicitly in the literature on the SbS framework, which all suppose the local spiking condition. Presumably because local optimization remains sufficient, and global optimization simply becomes unmanageable.

A network of neurons. To continue we define a spiking condition on the basis of local optimization and apply this to our objective (33) with representation (30). We obtain that neuron i spikes whenever the following condition holds:

$$L(\mathbf{s}_t = \mathbf{e}^i, \mathbf{x}_t) < L(\mathbf{s}_t = \mathbf{0}, \mathbf{x}_t) \tag{35}$$

$$\iff \left\|\mathbf{x}_t - \mathbf{D}\mathbf{r}_t - \mathbf{D}\mathbf{e}^i\right\|^2 + \nu^i < \left\|\mathbf{x}_t - \mathbf{D}\mathbf{r}_t\right\|^2$$

$$\iff (\mathbf{D}^i)^T\Big(\underbrace{\mathbf{x}_t - \mathbf{D}\mathbf{r}_t}_{\text{current error}}\Big) > \frac{\|\mathbf{D}^i\|^2}{2} + \nu^i, \tag{36}$$

where again, $\mathbf{D}^i$ is defined as $\mathbf{D}\,\mathbf{e}^i$, and the superscript T denotes the matrix-transpose.

Let us take (36) apart and note its properties. The spiking condition states that if the current representational error $\mathbf{x}_t - \mathbf{D}\mathbf{r}_t$, projected onto $\mathbf{D}^i$, exceeds a certain threshold, neuron i should spike. In our network, $\mathbf{D}^i$ is exactly the contribution that a spike of neuron i has on the representation. The projection is therefore a measure of how much neuron i could assist in reducing the representational error.

We shift our attention to a biological interpretation. As before in the one-dimensional case we can map (36) as a membrane potential and the right side as a threshold.

$$\mathbf{u}_t := \mathbf{D}^T\Big(\mathbf{x}_t - \mathbf{D}\mathbf{r}_t\Big) \qquad\qquad \boldsymbol{\vartheta} := \frac{\mathbf{diag}(\mathbf{D}^T\mathbf{D})}{2} + \boldsymbol{\nu}, \tag{37}$$

where **diag** is the function that returns the diagonal elements of a square matrix. The spiking condition for neuron i can then be rephrased as in the deterministic SRM: neuron i should spike if its membrane potential u_t^i exceeds its threshold ϑ^i,

$$u_t^i > \vartheta^i.$$

Furthermore, we observe that the membrane potential has two components: A component $\mathbf{D}^T\mathbf{x}_t$ where the bottom-up stimulus $\mathbf{x}_t$ is mapped onto the membrane potential by $\mathbf{D}^T$, which we interpret as feed-forward synaptic connections, and denote by $\mathbf{F}$. And the neural responses of neurons get mapped onto other membrane potentials of other neurons by $\mathbf{D}^T\mathbf{D}$, which we in turn interpret as the lateral synaptic connections, and denote by $\mathbf{W}$.

$$\mathbf{F} := \mathbf{D}^T, \qquad\qquad \mathbf{W} := -\mathbf{D}^T\mathbf{D}, \tag{38}$$

$$\mathbf{u}_t := \mathbf{F}\mathbf{x}_t + \mathbf{W}\mathbf{r}_t, \qquad\qquad \boldsymbol{\vartheta} := \frac{\mathbf{diag}(\mathbf{W})}{2} + \boldsymbol{\nu} \tag{39}$$

We illustrate the network topology in figure 2.3.

We conclude that the SbS framework is able to derive spiking neural networks that conform to our bio-plausiblity benchmark given by the spike response model. Furthermore, our network have the explicit functional objective of encoding a stimulus over time.

2.3 Learning as representation optimization

In this section we show how we can understand learning in the SbS framework. We will begin with a concise conceptual overview, followed by an exploration of technical considerations using the decoder $\mathbf{D}$ as an example, and conclude by addressing the disparity between top-down learning in the SbS framework and biological learning.

Our derivations started with a functional objective we wanted to perform, embodied by equation (33). In principle one could be agnostic to the precise structure of the representation $\hat{\mathbf{x}}_t$, and simply assume that it depends on certain (possibly time-dependent) parameters. This agnostic view provides a clear intuition about what the dynamics of any network parameter should accomplish: the parameters should continuously strive

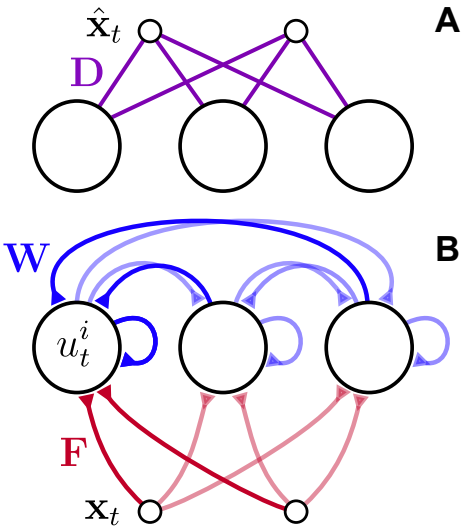

Figure 2.3: Resulting network architecture from the functional objective of forming a representation $\hat{\mathbf{x}}_t$ of a stimulus $\mathbf{x}_t$. **A**: Illustration of the objective of a network of neurons constructing a representation $\hat{\mathbf{x}}_t$. The functional objective is agnostic to the dynamics of the neurons themselves, it merely relates activity $\mathbf{s}_t$ to the representation $\hat{\mathbf{x}}_t$. **B**: Illustration of the resulting SRM derived by imposing a spiking condition. The feed-forward connection $\mathbf{F} = \mathbf{D}^T$ and lateral connection $\mathbf{W} = -\mathbf{D}^T\mathbf{D}^T$ are derived from a spiking condition imposing local optimization.

to optimize our representation to minimize loss L at any moment. Indeed, our spiking dynamics were derived with this view in mind.

We will illustrate this perspective further by introducing dynamics on the other component of the representation, namely the decoder matrix $\mathbf{D}$. To derive the dynamics we proceed similarly as with the spiking dynamics and require that our decoder components D^{ij} optimize the loss, i.e.

$$\frac{\partial}{\partial D^{ij}} L = 0$$

Here we have to proceed with care. If we optimize the (continuous) decoder components in a similar greedy fashion as before with the neural responses $\mathbf{s}_t$, we could set the decoder weights such that any non-zero spike trace $\mathbf{r}_t$ would be sufficient to perfectly represent the signal. This goes against the intuition that our decoder should evolve slowly, over longer timescales. Indeed, rapid alterations in a neural code may present difficulties for an organism as it would require constant adaptation for the code to be useful, which itself would be metabolically expensive. Furthermore, changing the neural code on a short time-scale would limit the ability of the neural code to generalize to longer timescales. These two perspectives on slow-adaptation are often seen to go hand in hand [8].

One can solve this issue heuristically by letting the parameters follow the gradient of the loss instead of explicitly finding the optimal solution. Here we show that our framework allows us to neatly circumvent the heuristics and derive slow adaptation by making our appeal to metabolic costs more precise.

We say it is costly for the organism to change it's neural code too fast. In this case the neural code is given by D_t^{ij}, which we will briefly treat as a function of time. Therefore we penalize the rate of change in the decoder by introducing an additional cost C'

$$C' := \frac{\gamma}{2\delta t} \sum_{i,j} \left\| \Delta D_t^{ij} \right\|^2, \tag{40}$$

where γ is a constant quantifying the penalty on the change and $\Delta D_t^{ij} := D_t^{ij} - D_{t-1}^{ij}$. We obtain a new loss L'

$$L' := L + C',$$

the optimization of which leads to an intuitive, regularized and slow adaption:

$$\begin{aligned} &\frac{\partial}{\partial D^{ij}} L' = 0 \\ \iff &\Delta D_t^{ij} = -\delta t \ \gamma^{-1} \frac{d}{dD^{ij}} L. \end{aligned} \tag{41}$$

Indeed, the dynamics for D^{ij} does not minimize the loss L directly, but instead follows the gradient on a timescale given by γ^{-1}. Learning in general systems is commonly understood as the gradual adaptation of a network to perform a task, as this is precisely what (41) embodies, we therefore call these dynamics *learning rules*.

The learning rule for $\mathbf{D}$ is then derived by taking the derivative of the deterministic loss L with respect to each D^{ij}, and we obtain

$$\Delta D_t^{ij} = -\delta t\ \gamma^{-1} \frac{d}{dD^{ij}} L = \delta t\ \gamma^{-1} \big(\mathbf{x}_t - \hat{\mathbf{x}}_t\big)^i (\mathbf{r}_t + \mathbf{s}_t)^j. \tag{42}$$

While we believe it is necessary to contextualize slow-adaptation in our framework, we will not explicitly add (40) to our losses. From here on we simply assume such a term is present for continuous variables such that we can always cast the adaptation of our representation in the form of equation (41). Moreover, we avoid the mention of time-dependence in our slow-variables and assume that the neural responses evolve on a much faster time-scale, for both ease of notation and conceptualization.

We conclude our discussion by evaluating the biological plausibility of our learning rule. We have set a standard for biological plausibility in spiking dynamics using the Spike Response Model. However, this benchmark does not take into account other aspects of representation, such as the dynamics of the decoder, represented as $\mathbf{D}$. As a result, we must explore other approaches to address this aspect.

The main issue lies in the fact that $\mathbf{D}$, unlike $\mathbf{s}_t$, lacks a direct biological interpretation. By presuming a fixed decoding matrix $\mathbf{D}$, we succeeded in deriving a SRM where $\mathbf{D}$ implied the form of the feed-forward $\mathbf{F}$ and lateral $\mathbf{W}$ connections, as well as determining the value of the threshold $\boldsymbol{\vartheta}$. Yet, as $\mathbf{D}$ is not an explicit component of the network, it is simply a part of the objective we aim to achieve with our network.

Therefore, when considering biological learning, our focus should be on the adaptation of actual synaptic connections, $\mathbf{F}$ and $\mathbf{W}$, within the network. In section 2.4 we present a specific solution to this challenge, although we note that this is still an active field of research.

2.3.1 Example: effect of metabolic costs on an optimized network

In this section we showcase consequences of adding a particular metabolic cost. First we will introduce a general metabolic cost, penalizing individual spikes and repeated activity. Next we show the derived network architecture corresponding to the metabolic cost. Finally we learn an optimal decoder $\mathbf{D}$ using the theory of the last section and showcase the effect of metabolic cost on an optimized network.

We assume metabolic cost of the form

$$C := \mu \|\mathbf{q}_t\|^2 + \nu \sum_i s_t^i + \frac{\rho}{2} \|\mathbf{D}\|^2, \tag{43}$$

with constants μ, ν, ρ, and similarly to r_t we define q_t as

$$\mathbf{q}_t := \sum_{t' \leq t} \mathbf{s}_{t'} \xi(t - t'), \tag{44}$$

for a kernel $\xi(t)$ (similar to the kernel κ) indicating how past neural activity affects the current cost of spiking. That is, we have a term $\mu \|\mathbf{q}_t\|^2$ penalizing large individual

activity in neurons, and a term $\nu \sum_i s_t^i$ penalizing individual spikes. Furthermore, we have a term $\rho\|\mathbf{D}\|^2$ penalizing the magnitude of the decoder. We motivate this as follows, from the perspective of an organism we have to consider the difficulty of extracting our neural code from our network of neurons. The organism quite literally has to construct a decoder. The decoder will consist of synaptic connections, and strong connections carry a significant metabolic cost. Hence we penalize the magnitude of the decoder $\mathbf{D}$ necessary to carry out the function.

From this cost together with function defined as (29) we can derive the following SRM network

$$\mathbf{F} := \mathbf{D}^T, \qquad \mathbf{W} := -\mathbf{D}^T\mathbf{D}, \tag{45}$$

$$\mathbf{u}_t := \mathbf{F}\mathbf{x}_t + \mathbf{W}\mathbf{r}_t - \mu\mathbf{I}\mathbf{q}_t, \qquad \boldsymbol{\vartheta} := \frac{\mathbf{diag}(\mathbf{W})}{2} + \nu + \frac{\mu}{2}\xi(0), \tag{46}$$

which is identical to our previously derived network in section 2.2, with additional self-inhibition $-\mu\mathbf{I}\mathbf{q}_t$. We note that the penalty on the decoder has no effect on the overall structure of the SRM.

Importantly, the term $-\mu\mathbf{I}\mathbf{q}_t$ can be seen as an adaptive current, penalizing repeated activity. The phenomenon of neuron adaptation refers to the observation that it becomes progressively more challenging to elicit repeated spiking within a short time frame. Indeed this formulation has close ties to general descriptions of adaptation [7]. Furthermore, this description has an exact correspondence to many commonly used models to study adaptive neurons [31, 43] as well as those used to test the computational capabilities of adaptive neurons [5, 6].

The role of the cost terms containing μ and ν are roughly orthogonal, one incentives a sparse encoding, while the other incentives a distributed encoding. In a network with equal ν for all neurons, a single neuron will be best equipped to encode a stimulus. This neuron will always reach the spiking threshold before the rest of the population resulting in a sparse neural code. In a network with non-zero μ, the spiking of a neuron subsequently raises its metabolic cost, disincentivizing further activity and therefore promoting a distributed code.

We illustrate these effects on a network of 9 neurons with the task of encoding a one-dimensional stimulus. The network is derived from the cost above with the simplification that that we set the penalizing kernel as $\xi = \kappa$ resulting in the following SRM

$$\mathbf{F} := \mathbf{D}^T, \qquad \mathbf{W} := -\mathbf{D}^T\mathbf{D} - \mathbf{I}\mu, \tag{47}$$

$$\mathbf{u}_t := \mathbf{F}\mathbf{x}_t + \mathbf{W}\mathbf{r}_t, \qquad \boldsymbol{\vartheta} := \frac{\mathbf{diag}(\mathbf{W})}{2} + \nu. \tag{48}$$

Lastly, we derive a learning rule to optimize the decoder $\mathbf{D}$ using equation (41)

$$\Delta D^{ij} = \frac{\delta t}{\gamma}\left(\left(\mathbf{x}_t - \hat{\mathbf{x}}_t\right)^i(\mathbf{r}_t + \mathbf{s}_t)^j - \alpha\mathbf{D}^{ij}\right). \tag{49}$$

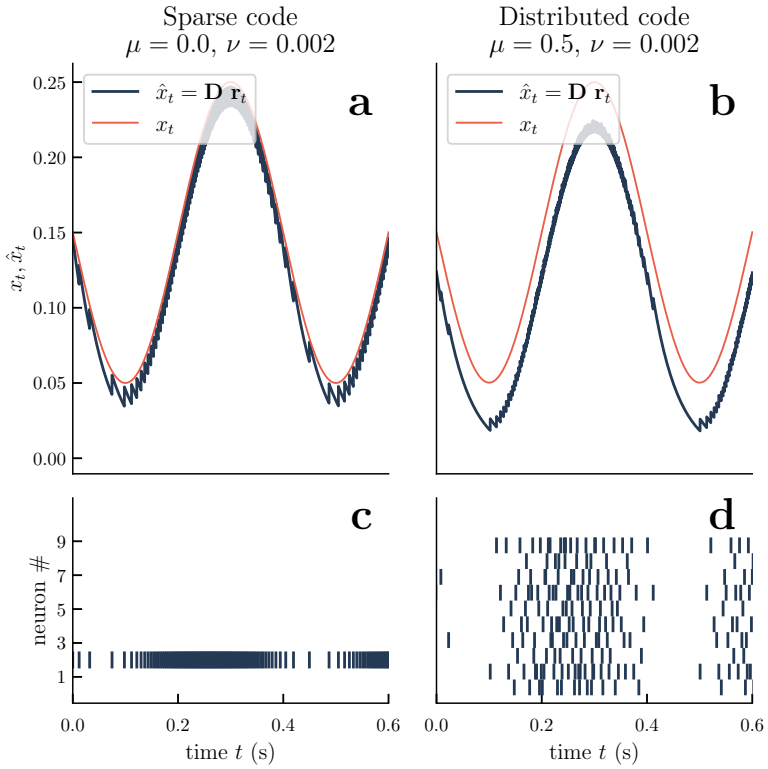

Figure 2.4: Two networks of 10 neurons encoding a one-dimensional stimulus. Both networks were parameterized with a different metabolic cost resulting in a sparse encoding on the left side, and distributed encoding on the right side. **a**,**b**: encoding of the stimulus for the two different networks. The metabolic cost on the right is strictly larger, therefore also resulting in a worse encoding, as is evident by the gap between the representation $\hat{x}_t$ and x_t. **c**,**d**: raster plots showcasing the activity of both networks. (Simulation parameters in

We illustrate these effects for two different values of μ showing two optimized networks attempting to encode a one-dimensional stimulus. The networks were optimized by using the learning rule for the decoder $\mathbf{D}$ shown above. The results are shown in figure 2.4.

With only ν non-zero we obtain a sparse encoding, as there will always be a single neuron best suited to encode the stimulus. However, with a non-zero μ the repeated spiking of a single neuron will be penalized to ensure that multiple neurons participate in representing the signal. Here we increased μ sufficiently to ensure the participation of the entire network. Note that the presence of a non-zero ρ ensures that the weights will remain relatively small, explaining the offset between $\hat{x}_t$ and x_t.

2.4 Application: bio-plausible learning

This section briefly showcases one of the principal applications of the SbS framework: bio-plausible learning. A significant portion of the literature [16, 62, 15, 17, 48] devoted to the SbS framework concerns this very subject. In particular we illustrate how the SbS can provide a backbone of intuition for bio-plausible learning in spiking neural networks.

Previously, we have derived dynamics for the various components of our representation. However, we recognized that the dynamics derived for the decoder were not biologically plausible. The main reason lies in the fact that the decoder in the representation is separate from the network's actual architecture, particularly the connection matrices $\mathbf{F}, \mathbf{W}$. Consequently, we cannot directly translate the optimization of our representation into the plasticity of these connection matrices.

Thus, we can phrase the problem of bio-plausible learning more precisely as follows. Given that we have a SRM with membrane potential $\mathbf{u}_t$, defined as:

$$\mathbf{u}_t := \mathbf{F}\mathbf{x}_t + \mathbf{W}\mathbf{r}_t, \tag{50}$$

with connection matrices $\mathbf{F}, \mathbf{W}$ and threshold $\boldsymbol{\vartheta}$, how can we optimize this network to align with our derived optimal network for the loss function:

$$L = \frac{1}{2}\|\mathbf{x}_t - \hat{\mathbf{x}}_t\|^2 + \boldsymbol{\nu} \cdot \mathbf{s}_t + \frac{\rho}{2}\|\mathbf{D}\|^2? \tag{51}$$

This question does not have a comprehensive, principled answer. However, existing literature [16] offers some insightful intuition. The crux of the idea is the following: since we derived the dynamics of our membrane potentials from an objective loss function L, it is plausible to assume that the membrane potential encodes enough information to compute gradients.

We can illustrate this idea with the recurrent weights of the optimal network. Indeed, the optimal decoder $\mathbf{D}^*$ satisfies

$$0 = -\left\langle \left(\mathbf{x}_t - \mathbf{D}^*\mathbf{r}_t\right)\mathbf{r}_t^T \right\rangle + \rho\mathbf{D}^*, \tag{52}$$

where $\langle\cdot\rangle$ without the subscript over distributions denotes an average over time. Likewise we know that the optimal configuration of the membrane potential is given by (38) and reads

$$\mathbf{u}_t^* := \mathbf{D}^{*T}\left(\mathbf{x}_t - \mathbf{D}^*\mathbf{r}_t\right). \tag{53}$$

Multiplying (52) by $\mathbf{D}^{*T}$ we obtain that for the optimal network configuration the recurrent weights $\mathbf{W}^*$ have the form

$$\mathbf{W}^* = -\mathbf{D}^{*T}\mathbf{D}^* = -\frac{1}{\rho}\left\langle \mathbf{u}_t^*\mathbf{r}_t^T \right\rangle \tag{54}$$

Next we show how we can adapt an arbitrary $\mathbf{W}$ to $\mathbf{W}^*$. To achieve this we turn to another central idea: we can separate the different timescales over which our dynamics evolve[2] as introduced by Bourdokan et al. [16] and elaborated on by Mikulasch et al. [48] We suppose that the timescale is ordered from fast to slow as $\mathbf{r}_t, \mathbf{x}_t, \mathbf{W}, \mathbf{F}$. From the perspective of the lateral connections $\mathbf{W}$, the dynamics of $\mathbf{r}_t$ and $\mathbf{x}_t$ are experienced by their statistical properties, while the connection $\mathbf{F}$ is viewed as constant. As the feed-forward weights $\mathbf{F}$ are viewed as constant, the lateral connections $\mathbf{W}$ can only hope to approximate the optimal solution $-\mathbf{D^*}^T\mathbf{D^*}$ and evolve to the form $-\mathbf{FD^*}$, learning the optimal decoder inside the feed-forward weights. Indeed, if we assume $\mathbf{W}$ to be of the form $\mathbf{W} = -\mathbf{FD}$, we can evolve $\mathbf{W}$ as

$$\begin{aligned}\Delta\mathbf{W} &\propto -\mathbf{F}\frac{\partial L}{\partial \mathbf{D}} && (55)\\ &= -\mathbf{F}\big(\mathbf{x}_t - \hat{\mathbf{x}}_t\big)\mathbf{r}_t^T + \rho\mathbf{FD} && (56)\\ &= -\mathbf{u}_t\mathbf{r}_t^T - \rho\mathbf{W}, && (57)\end{aligned}$$

we obtain a learning rule with the fixed point

$$\mathbf{W} = -\frac{1}{\rho}\langle \mathbf{u}_t\mathbf{r}_t^T\rangle. \tag{58}$$

In addition to having the correct fixed point, the learning rule is local. We can see this as the change in each individual lateral connection is given by

$$\Delta W^{ij} \propto -u_t^i r_t^j - \rho W^{ij}, \tag{59}$$

requiring each connection only to have information about the pre-synaptic potential from neuron j and the membrane potential of neuron i, implying a local bio-plausible learning rule. The recurrent connections W^{ij} adapt themselves to make sure that the PSP of neuron j balances the membrane potential of neuron i. From the interpretation of the membrane potential as the error, we see that that the learning rule implies that the PSP of neuron j is decorrelated from the error encoded by neuron i. We see that the SbS framework has therefore allowed us to derive a bio-plausible learning rule that is greatly interpretable.

2.5 Summary and re-conceptualization

In the preceding sections, we explored the SbS framework and demonstrated how the framework can derive spiking neural networks that balance function with metabolic constraints. We also illustrated how the framework could foster intuition around bio-plausible optimization.

To extend the approach, it is essential to zoom out and summarize the explicit choices made throughout the derivations.

[2]We can make the timescale argument formal by appealing to γ^{-1} in section 2.3

-Greedy optimization We defined a loss function L quantifying the performance of our network. The loss was defined for each time-step t separately, and our spiking condition did not take into account any possible future loss. While it has been shown in the literature [17] that this does not limit performance much with an exponential kernel κ, this prevents us from considering representations without a strong initial transient response, as the spiking condition cannot separate the case of $s_t = 1$ and $s_t = 0$.

-Deterministic spiking Our loss function was defined over deterministic binary variables s_t^i. It is, however, possible to optimize with respect to a distribution over s_t^i.

-Functional objective We made an explicit choice for the functional component F of our loss function. We proposed the square difference as a measure of how well our representation $\hat{x}_t$ fit the stimulus x_t. While the square loss is convenient and easily interpretable, the SbS is in principle not limited to the square loss.

-Neural representation All of the work on the SbS framework explicitly sets the representation to be a linear sum of neural responses. However, in principle, we have much freedom here. For instance, in rate networks it is common to have representations be a non-linear function of neural activity [57].

-Metabolic costs The balance of function with metabolic costs depends on how valuable the function is to the organism. Here we focused on simple loss functions presuming a spike, or repeated activity, to be costly. But again, the metabolic cost should be related to the utility of a specific representation. We therefore expect more complex representations to naturally lead to more considerations around the cost function

The various choices offer future paths of generalization.

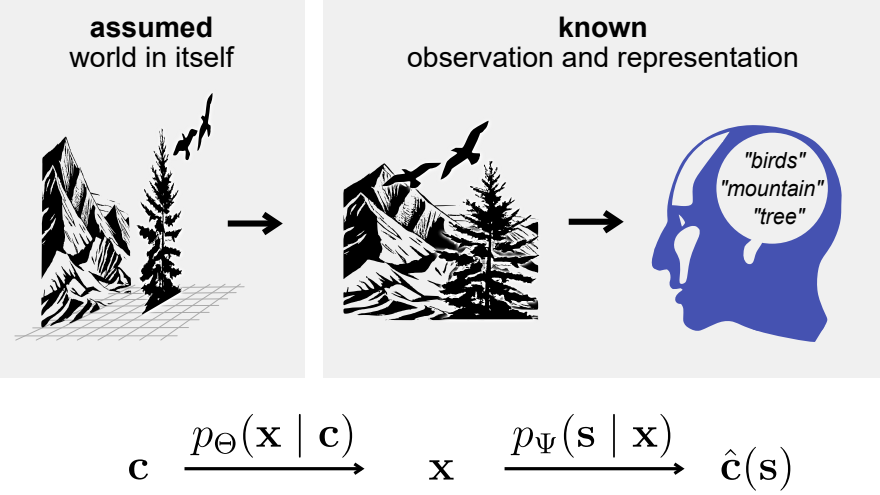

Figure 3.1: Cartoon depiction of our functional objective of performing approximate Bayesian inference. We assume that there is a model $p_\Theta(\mathbf{x}, \mathbf{c})$ adequately describing the observations $\mathbf{x}$ of the world itself. These observations are assumed to be caused by some underlying latent structure $\mathbf{c}$. The objective of the Bayesian brain is to have the brain $p_\Psi(\mathbf{s} \mid \mathbf{x})$ implement (approximate) inference with respect to the world model, where the causes $\mathbf{c}$ will be encoded in the activity of the brain by a neural representation $\hat{\mathbf{c}}(\mathbf{s})$. (Figure used with permission of Fabian Mikulasch)

3 Results - A noisy world

In this section we provide our main theoretical results. We start by introducing a new functional objective. Instead of encoding a stimulus, we aim to perform approximate Bayesian inference. Then, we will build upon the conceptual foundations of the SbS framework and extend its approach for this new objective. We show that again, by considering the trio of Function, Neural representation and Metabolic cost, we will be able to derive bio-plausible neural networks. We continue by illustrating how to interpret learning from a top-down, inference centered perspective, and we show the capacity of our networks by simple experiments. We finish the section by briefly discussing the relation between our network to that of predictive coding, an ambiguous, but widely used term in theoretical neuroscience.

3.1 Inference as an objective

A criticism of the SbS framework is that the function of encoding a stimulus alone does not reflect real-life utility of the brain. Neither the sensory stimuli nor the dynamics of the brain are deterministic [35]. Furthermore, many parts of the sensory system serve as an incredible information bottleneck [21], implying that the brain extracts relevant features from the stimulus for future computation.

A much more useful objective is given by Bayesian inference. Our starting assump-

tion will be in line with the Bayesian brain hypothesis. The brain somehow implements a model of the world and performs approximate Bayesian inference, extracting the different causes that give rise to observations. The utility and elegance of this proposal is what inspired the Bayesian brain hypothesis. We follow this line of thought and show how we can make a network of spiking neurons perform approximate Bayesian inference.

We consider how we can make the brain $p_\Psi(\mathbf{s} \mid \mathbf{x})$ — now defined as a probability distribution over activity $\mathbf{s}$, given a stimulus $\mathbf{x}$ — best represent a model of the world $p_\Theta(\mathbf{x}, \mathbf{c})$. Here we define Ψ as the brain parameters such as synaptic strength and Θ as the world model parameters.

This problem is currently ill-defined. We note that the model of the world $p_\Theta(\mathbf{x}, \mathbf{c})$ is defined in terms of latent causes $\mathbf{c}$, not in terms of neural activity. If we want the brain to represent $\mathbf{c}$, we need to make explicit how those causes are represented. And for this we require exactly the second part of the trio: a neural representation $\hat{\mathbf{c}}(\mathbf{s})$ of the latent causes $\mathbf{c}$. We define the representation $\hat{\mathbf{c}}$ as a function of neural activity completely analogous to defining the representation $\hat{\mathbf{x}}$ in terms of neural activity, as we did before. Given a neural representation, we can consider our model of the world in terms of activity $\mathbf{s}$ instead of causes $\mathbf{c}$[3].

In the background on variational inference (1.4), we showed that the expected free energy $\mathcal{F}$ quantifies both the degree to which $p_\Psi(\mathbf{s} \mid \mathbf{x})$ and $p_\Theta(\hat{\mathbf{c}}(\mathbf{s}) \mid \mathbf{x})$ match as distributions, and the degree to which $p_\Theta(\hat{\mathbf{c}}(\mathbf{s}), \mathbf{x})$ explains the observations $\mathbf{x}$. These properties suggest that $\mathcal{F}$ is an ideal candidate as a functional objective quantifying the degree to which we perform inference.

As such we define the functional part of our objective as the free-energy $\mathcal{F}$ between the brain and the model given a representation, as defined in section (17).

$$\mathcal{F} := \mathcal{F}[p_\Psi(\mathbf{s} \mid \mathbf{x}), p_\Theta(\hat{\mathbf{c}}(\mathbf{s}), \mathbf{x})] = \langle \ln p_\Psi(\mathbf{s} \mid \mathbf{x}) - \ln p_\Theta(\mathbf{x}, \hat{\mathbf{c}}(\mathbf{s})) \rangle_{p_\Psi(\mathbf{s}|\mathbf{x})}. \tag{60}$$

To complete our framework need to define metabolic costs with which we want to balance function, proceeding in the same fashion as the SbS framework. The free energy is an expected value over the brains activity, in that spirit we define the metabolic cost as the expected value of a deterministic cost C, possibly dependent on the activity $\mathbf{s}$ and other parameters of the representation. Now that we have completed our trio we can write down the probabilistic loss function $\mathcal{L}$ as the sum of the free energy and the expected cost defining our overall objective:

$$\mathcal{L} := \underbrace{\mathcal{F}}_{\text{Function}} + \underbrace{\langle C \rangle_{p_\Psi(\mathbf{s}|\mathbf{x})}}_{\text{Metabolic costs}}. \tag{61}$$

Before we proceed we have to note the main components on which the loss function depends, and note what the optimization of these components accomplishes. The three components are the brain given by $p_\Psi(\mathbf{s} \mid \mathbf{x})$, the world model parameters given by Θ, and the parameters defining the representation $\hat{\mathbf{c}}$ (such as the decoder matrix before).

[3]We note that this implicitly does change the distribution. We return to this issue in section 4.3

Optimization of $\mathcal{L}$ with respect to the dynamics $p_\Psi(\mathbf{s} \mid \mathbf{x})$ balances model inference of $p_\Theta(\mathbf{x}, \hat{\mathbf{c}}(\mathbf{s}))$ with the cost of implementing such a model C.

Optimization of $\mathcal{L}$ with respect to Θ is the optimization of $\mathcal{F}$ with respect to Θ and therefore aims to make the world model a better description of the observations $\mathbf{x}$, and has precisely the usual interpretation in variational inference.

Optimization of $\mathcal{L}$ with respect to the parameters defining the representation is somewhat more subtle. Formally these parameters become part of the world model when we replace $p_\Theta(\mathbf{c}, \mathbf{x}) \to p_\Theta(\hat{\mathbf{c}}(\mathbf{s}), \mathbf{x})$, and therefore we want to optimize them so that our model best describes the observations. However, the representational cost must also be considered. Thus, we balance between improving the model's fit to observations and the incurred cost of model implementation.

In the next section we will illustrate the optimization of the loss with respect to $p_\Psi(\mathbf{s} \mid \mathbf{x})$ for a one-dimensional neuron, implementing a one-dimensional model of the world.

3.2 A one-dimensional noisy world

The next two subsections will continue analogously to the Background on the SbS framework, where instead of a single neuron encoding a one-dimensional stimulus, we consider how we can make a single neuron implement a one-dimensional model of the world. The first subsection will build the theoretical scaffolding, and the second will feature a practical implementation which closely mimics the dynamics found in the SbS framework.

3.2.1 Theory

We will start by developing the theory for a single neuron performing inference with spikes. We suppose we have a world model $p_\Theta(x_t, c_t)$ which at each point in time t explains a one-dimensional observation x_t based on some underlying one-dimensional latent cause c_t. Furthermore we have a representation $\hat{c}_t(s_t)$ as function of neural activity, and a metabolic cost C as a function of neural activity and other parameters defining the representation. We reasoned that the objective of performing approximate inference is encoded well by the probabilistic loss $\mathcal{L}$ as a sum of the free-energy and expected metabolic cost.

$$\mathcal{L} = \mathcal{F}[p_\Psi(s_t \mid x_t), p_\Theta(\hat{c}(s_t), x_t)] + \langle C \rangle_{p_\Psi(s_t|x_t)}, \tag{62}$$

for clarity, we will solve the problem without a metabolic cost first, so

$$\mathcal{L} = \mathcal{F}[p_\Psi(s_t \mid x_t), p_\Theta(\hat{c}(s_t), x_t)] \tag{63}$$

We continue with the optimization with respect to $p_\Psi(s_t \mid x_t)$. The loss is expressed as an expectation value over the distribution $p_\Psi(s_t \mid x_t)$, indicating that optimization is going to result in a distribution of activity, contrasting the deterministic dynamics

derived in the SbS framework. As the loss is just composed of the the free-energy, we can use the relation (16) to rewrite the objective loss as

$$\mathcal{L} = D_{KL}[p_\Psi(s_t \mid x_t) \,||\, p_\Theta(\hat{c}_t(s_t) \mid x_t)] - \ln p_\Theta(x_t), \tag{64}$$

where we see that the only term dependent on the brain p_Ψ is the KL-divergence between the brain and the model given a representation. By the properties of the KL-divergence, the distribution $p_\Psi(s_t \mid x_t)$ minimizing our loss $\mathcal{L}$ is simply the posterior $p_\Theta(\hat{c}_t(s_t) \mid x_t)$.

We have to note our abuse of notation, the posterior is now a distribution over s_t, instead of c_t, as we performed a very non-trivial change of variables from c_t to s_t. We assume that the distribution written above is again properly normalized. We will elaborate on this in the discussion.

Distributions over spikes I. To continue, one might be concerned that calculating the posterior $p_\Theta(\hat{c}_t(s_t) \mid x_t)$ is going to be difficult, but in our simple setting this will not be the case. To show why, we will have to divert our discussion to show particular properties of distributions over binary variables. These properties will prove crucial to understand and utilize our framework.

To avoid confusion we have to state the following results are, to our knowledge, original, and we have not previously encountered them in the literature. The seed of this work was an unexplained connection between the deterministic SbS framework and the work of Mikulasch et al. [48, 49] and the following derivation makes the connection explicit.

We start by noting general properties of distributions over binary variables, which in our case will be spikes. Next we consider the posterior of our network as a distribution over binary variables. And we conclude by noting how this procedure grants us an extremely appealing form with which we can represent the posterior $p_\Theta(\hat{c}_t(s_t) \mid x_t)$.

We note a property of a general function $g(s)$ of a binary variable $s \in \{0, 1\}$. $g(s)$ can always be written in the following way:

$$g(s) = \delta_{s,0}\; g(s=0) + \delta_{s,1}\; g(s=1), \tag{65}$$

then, making use of the following representation of the Kronecker delta

$$\delta_{s,s'} = \begin{cases} s, & \text{if } s' = 1 \\ (1-s) & \text{if } s' = 0 \end{cases}, \tag{66}$$

we can write

$$\begin{aligned} g(s) &= (1-s)\; g(s=0) + s\; g(s=1) \\ &= s(g(s=1) - g(s=0)) + g(s=0). \end{aligned} \tag{67}$$

Next, we can write any (non-degenerate) distribution $p(s)$ over s as

$$p(s) \propto \exp\big(-\ell(s)\big), \tag{68}$$

where $\ell(s)$ denotes the negative log-probability.

Combining these facts we get that any (non-degenerate) distribution over s can be written in terms of a difference in log-probabilities

$$p(s) \propto \exp\big(- s(\ell(s=1) - \ell(s=0))\big), \tag{69}$$

Next we relate the difference in ℓ to posterior inference with binary variables. For a model $p(x_t, s_t)$, the posterior $p(s_t \mid x_t)$ is given in terms of a difference in the in log-posterior probability $\ln p(s_t \mid x_t)$. Remarkably, this difference can be written as a difference in total model probability:

$$\ln p(s_t = 1 \mid x_t) - \ln p(s_t = 0 \mid x_t) \tag{70}$$
$$= \ln p(s_t = 1, x_t) - \ln p(s_t = 0, x_t), \tag{71}$$

as we have $p(s_t \mid x_t) \propto p(s_t, x_t)$ with the normalization constant being independent of s_t. Particularly, we obtain that the posterior $p_\Theta(\hat{c}_t(s_t) \mid x_t)$ can be written as difference in the negative log-probability of the entire model $\ell_\Theta(s_t, x_t) := -\ln p_\Theta(\hat{c}_t(s_t), x_t)$.

$$p_\Theta(\hat{c}_t(s_t) \mid x_t) \propto \exp\big(- s(\ell_\Theta(s_t = 1, x_t) - \ell_\Theta(s_t = 0, x_t))\ \big). \tag{72}$$

And we obtain that the posterior is given as a difference between the log-probabilities of the entire model, a quantity we have easy access to. Indeed, this difference is equivalent in form to the spiking condition of the SbS framework, where instead of comparing the loss 62 we compare ℓ_Θ.

Metabolic costs. To complete the conceptual parallel to the SbS framework we add the final component: **Metabolic costs** $C(s_t)$ as in (90) and define the objective loss $\mathcal{L}$

$$\mathcal{L} = \underbrace{\mathcal{F}[p_\Psi(s_t \mid x_t), p_\Theta(\hat{c}(s_t), x_t)]}_{\text{Function}} + \underbrace{\langle C(s_t) \rangle_{p_\Psi(s_t \mid x_t)}}_{\text{Metabolic costs}} \tag{73}$$

We solved equation (63) by rewriting it in terms of a KL-divergence. Here we proceed in the same fashion. First we write our probabilistic loss in term of a KL-divergence

$$D_{KL}(p_\Psi(s_t \mid x_t) \mid\mid p_\Theta(\hat{c}_t(s_t) \mid x_t)) + \langle C(s_t) \rangle_{p_\Psi(s_t \mid x_t)} - \ln p_\Theta(x_t) \tag{74}$$

Next, the sum of the KL-divergence and expected cost can be rewritten as (for a derivation we refer to the appendix A.1)

$$\begin{aligned} &D_{KL}(p_\Psi(s_t \mid x_t) \mid\mid p_\Theta(\hat{c}_t(s_t) \mid x_t)) + \langle C(s_t) \rangle_{p_\Psi(s_t \mid x_t)} \\ =&D_{KL}(p_\Psi(s_t \mid x_t) \mid\mid p'_\Theta(s_t \mid x_t)) \end{aligned} \tag{75}$$

with

$$p'_\Theta(s_t \mid x_t) \propto p_\Theta(\hat{c}_t(s_t) \mid x_t)) \exp(-C(s_t)) \tag{76}$$

The solution $p_\Psi(s_t \mid x_t)$ for the minimization of (74) is therefore given by

$$p_\Psi(s_t \mid x_t) \propto \exp(-s_t(L(s_t = 1, x_t) - L(s_t = 0, x_t))), \tag{77}$$

with L being defined as $L(s_t, x_t) = \ell_\Theta(s_t, x_t) + C(s_t)$.

We remain to call the function L the *deterministic loss* as it determines the membrane potential identically in the SbS framework. Indeed, as the spiking condition (25) is what gave us the membrane potential u_t and a threshold ϑ, we obtain that at any moment, the probability of firing is equal to

$$p_\Psi(s_t = 1 \mid x_t) = \frac{e^{u_t - \vartheta}}{1 + e^{u_t - \vartheta}} \tag{78}$$

which solidifies the connection between the 1-dimensional SbS framework and inference using spiking networks.

Biological plausibility and Information costs. Having derived an expression for the spiking probability we hope that we can map this expression onto the SRM, serving as our benchmark of biological plausibility as discussed in section 1.2.2. Unfortunately, this is not the case as our benchmark supposes our spiking probabilities to be of the form

$$p_\Psi(s_t = 1 \mid x_t) = \delta t \ \rho(u_t - \vartheta). \tag{79}$$

Here, the discretization timescale δt assures an instantaneous spiking rate equal to $\rho(u_t - \vartheta)$ at time t. It is important to note, however, that equation (78) lacks this important pre-factor δt.

We can correct this discrepancy by incorporating a particular spiking cost. We will first demonstrate that this cost effectively molds our network to align with the SRM benchmark. Following this, we will deliberate its justification.

Modifying our cost to $C(s_t) \to C(s_t) + s_t \ln \frac{1}{\delta t}$ grants us the solution $p_\Psi(s_t \mid x_t)$ given by equation (77). Writing

$$p_\Psi(s_t = 1 \mid x_t) = \frac{\delta t \ e^{u_t - \vartheta}}{1 + \delta t \ e^{u_t - \vartheta}} \xrightarrow{\delta t \to 0} \delta t \ \exp(u_t - \vartheta). \tag{80}$$

which indeed mends the problem as we will always consider our discretization timescale δt to be small.

Next we argue for this specific cost from the standpoint of a living creature. For any creature, the precise timing of neural responses are useful, but only up to a certain point. This implies a time scale Δt, beyond which, any differences in the timing of spikes are considered equivalent from the creature's perspective. We can relate this to our objective if we note that the metabolic cost is weighing against the free energy, expressed in terms of information. Indeed, the exact timing of spikes carries a wealth of information that, unfortunately, the creature cannot perceive. We therefore propose to discount the information content of a single spike to conform to this timescale.

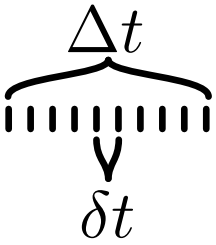

Figure 3.2: An illustration of the different timescales. From a functional perspective we do not interest ourselves in precise timings beyond a resolution of the order Δt. Nevertheless, our dynamics are defined on the much finer timescale of δt. The information content of a knowing a the exact spike timing on the timescale δt in a Δt bin is given by the log of the number of bins $\ln \frac{\Delta t}{\delta t}$ (Given that there is precisely one spike in the interval Δt)

To make this more formal we will explicitly consider the information per spike that is unattainable by the creature. The events within our system are the possible individual spikes. Assuming these spikes are distributed within a Δt period, with a bin size of δt, then the number of distinct spike timings within our Δt period is given by $\Delta t/\delta t$ (as shown in figure 3.2). If each of these timings is considered to be equally likely, with a probability of $\delta t/\Delta t$, information theory states that the information content of knowing a particular spike timing within the interval Δt is given by $-\ln(\frac{\delta t}{\Delta t})$. Based on the previous reasoning, we argue that we should discount the information content of a single spike by $\ln(\frac{\delta t}{\Delta t})$. We can incorporate this discount into our cost C by adding a term $-s_t \ln(\frac{\delta t}{\Delta t})$. Then, as we can write $-s_t \ln(\frac{\delta t}{\Delta t}) = s_t \ln(\Delta t) - s_t \ln(\delta t)$ we get that this additional cost then introduces the required δt factor to make our networks conform to the SRM.

Before we end the section we make two final comments. First, we have written $\ln(\Delta t)$ and $\ln(\delta t)$, the logarithm of a quantity with units of time. Strictly, this is not defined and we are required to choose a consistent reference timescale τ_{ref} and write $\ln(\Delta t/\tau_{\text{ref}})$. Moreover, we have to note that $\ln(\Delta t/\tau_{\text{ref}})$ can be either positive or negative, and therefore if we absorb this into the cost $C = \nu s_t$, we have to understand that a positive ν can make sense. Moving forward, we will standardize these considerations, thereby consistently generating the required pre-factor δt.

3.2.2 Example: Single neuron SbS Kalman-like filter

In this section we complete the parallel to the one-dimensional SbS framework. We show that our model can derive identical membrane dynamics, and in doing so, it performs a very similar task to that of a particle Kalman filter.

In the previous section we noticed that we derive the membrane potential from a difference in the deterministic loss L defined as $L := \ell_\Theta + C$. To continue, we obtain ℓ_Θ by defining a generative model $p_\Theta(x_t, c_t)$ in its standard decomposition of a prior

over c_t, $p_\Theta(c_t)$ and an observation model $p_\Theta(x_t \mid c_t)$

$$p_\Theta(x_t, c_t) = p_\Theta(x_t \mid c_t) p_\Theta(c_t). \tag{81}$$

We choose our prior and likelihood to be normal distributions, with x_t assumed to be normally distributed around c_t with variance $1/\beta_x$ and similarly in the prior we assume c_t to be normally distributed around a constant c_p with variance $1/\beta_c$.

$$p_\Theta(x_t \mid c_t) \propto \exp\left(-\frac{\beta_x}{2}(x_t - c_t)^2\right) \qquad p_\Theta(c_t) \propto \exp\left(-\frac{\beta_c}{2}(c_t - c_p)^2\right). \tag{82}$$

We note that the generative model itself has no notion of a time-dependency. Indeed the model assumes every time-step to be independent of another. Nevertheless, we will see that the representation $\hat{c}_t$ introduces a time-dependency.

For this specific model choice we obtain that our negative model log-probability ℓ_Θ is given by

$$\begin{aligned} \ell_\Theta(c_t, x_t) &= \ln p_\Theta(x_t, c_t) && (83)\\ &= \ln p_\Theta(x_t \mid c_t) + \ln p_\Theta(c_t) && (84)\\ &= \frac{\beta_x}{2}(x_t - c_t)^2 + \frac{\beta_c}{2}(c_t - c_p)^2 + \text{const}(\Theta, x_t) && (85) \end{aligned}$$

We choose our representation $\hat{c}_t$ similarly as (24)

$$\hat{c}_t := D \sum_{t' \le t} s_{t'} \kappa(t - t') = D\left(r_t + s_t\right), \tag{86}$$

where again $\alpha = \exp\left(-\frac{\delta t}{\tau}\right) \in (0, 1)$ is the decay constant of the leaky current with δt the timescale of our discretization, τ the time-scale of the leaky current, and D a scalar constant determining the scaling of the representation, and $s_t \in \{0, 1\}$ the activity of our neuron at time t.

Metabolic costs are defined as (28) with the additional term $\ln \delta t$,

$$C(s_t) = (\nu + \ln \delta t)\ s_t,$$

where ν is the cost of a single spike. For completeness we note that ν absorbs the factor $\ln(\Delta t)$ mentioned in the previous section 3.2.1 and is therefore allowed to take both positive and negative values.

The solution $p_\Psi(s_t \mid x_t)$ is then given by (77) combined with (80), and we obtain

$$\begin{aligned} p_\Psi(s_t \mid x_t) &\propto \delta t \exp\left(\beta_x D\left(x_t - D r_t\right) + \beta_c D\left(c_p - D r_t\right) - \frac{D^2}{2}\left(\beta_x + \beta_c\right) - \nu\right) \\ &= \delta t \exp\left(u_t - \vartheta\right) && (87) \end{aligned}$$

with

$$u_t := \beta_x D\left(x_t - D r_t\right) + \beta_c D\left(c_p - D r_t\right) \qquad \vartheta := \frac{D^2}{2}\left(\beta_x + \beta_c\right) + \nu. \tag{88}$$

so that $p_\Psi(s_t \mid x_t)$ defines a SRM with exponential escape noise.

In particular we see that if we set $\beta_c = 0$, and $\beta_x = 1$, we recover a membrane potential identical to that of the deterministic SbS framework

$$u_t^{\mathrm{SRM}} := D\left(x_t - Dr_t\right), \qquad \vartheta^{\mathrm{SRM}} := \frac{D^2}{2} + \nu. \tag{89}$$

Let us consider both the technical and the conceptual reason for why this occurs.

For the technical reason we simply have to consider the form of the deterministic loss L as determined by the log-probability ℓ_Θ of a normal distribution. The log-probability ℓ_Θ as given by (85) becomes identical (up to a constant) to the functional part of the SbS framework if we set $\beta_x = 1$ and $\beta_c = 0$. As such L in this context becomes identical to the SbS framework.

For the conceptual reason we have to consider two objectives. The network derived from the SbS objective desires to track the signal x_t given metabolic constraints. It does not integrate any prior information of the signal, nor does it need to, as the signal is assumed to be free of noise. Our framework is different, as it is derived from Bayesian considerations which inherently integrate prior and current knowledge. The term containing β_c simply indicates the importance of leaning on prior statistics.

Our model definition is almost identical to that of a Kalman-filter. As such we can hope that our dynamics can represent both the mean as well as the variance of a noisy stimulus. This effect is illustrated in figure (3.3) where we show how a single neuron is able to roughly track the mean of noisy signal.

We note that the kind of interpolation that our probabilistic representation achieves is not possible for the deterministic SbS framework. The deterministic spiking rule will always respond to the extreme values of the envelope of a noisy signal, making it unable to represent the mean or the statistics.

Furthermore, we note that our model performs reasonable inference even with a generative model that has an independence of time. Our representation effectively introduces a time-dependency.

Lastly, we discuss the effect of a metabolic cost ν on our single neuron performing approximate inference. The metabolic cost enters linearly in the threshold (88) of our network. This linearity makes it particularly easy to understand. Our networks are spike response models with exponential escape noise. This implies that we can factor out the effect of ν, and interpret it as base firing rate of the network, reflecting the cost of spiking.

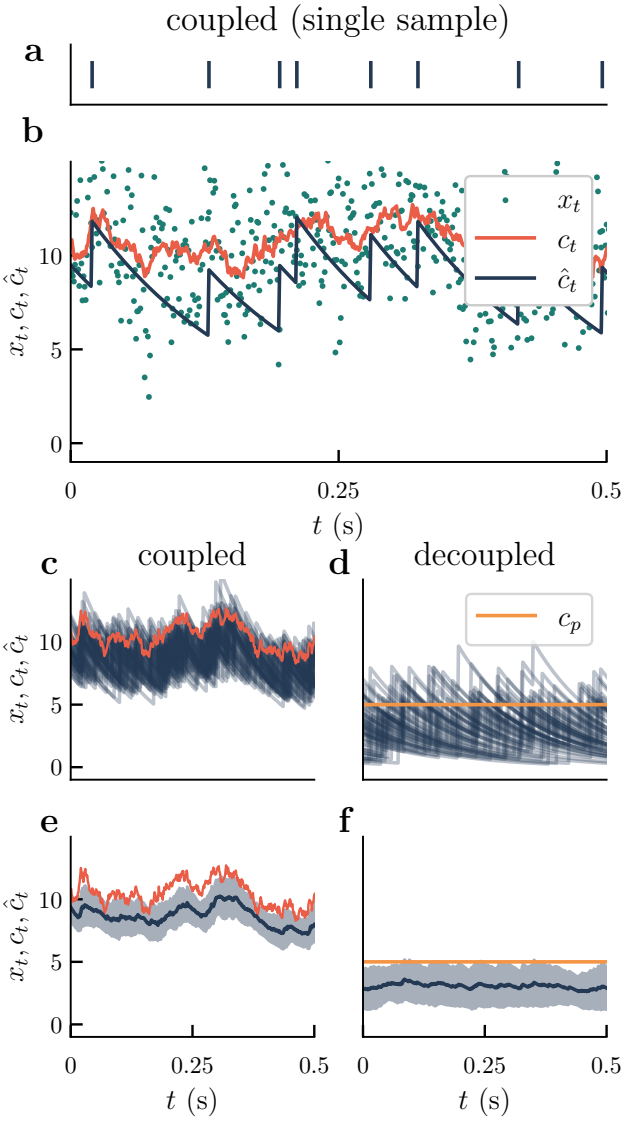

Figure 3.3: Single neuron performing approximate inference impaired by significant metabolic costs. We observe that the trajectory captures some general features of the latent signal, albeit at an offset. The offset is similar as previously exhibited in the deterministic case shown in example 2.3.1. **a,b**: A single sample of a coupled ($\beta_x \neq 0$) neuron performing approximate inference. The neuron observes the noisy signal x_t generated by adding white noise to an underlying latent signal c_t. The neural dynamics appear to produce a stochastic representation $\hat{c}_t$ of the latent c_t. **a** shows a rasterplot of activity over time and **b** shows the latent c_t, the stimulus x_t and representation $\hat{c}_t$. **c**: 50 coupled sample trajectories as in **b**, where we can see that the trajectories seem to approximate the latent cause c_t. **d**: 50 uncoupled ($\beta_x = 0$) sample trajectories, representing the model prior encoded by a single neuron. **e,f**: statistical properties of the 50 sample trajectories where the bold line indicates the mean of the samples and the shaded regions indicate a single standard deviation. (Simulation parameters given in table 2)

3.3 A multi-dimensional noisy world

In this section we expand to a network of neurons performing inference on a multi-dimensional stimulus. We proceed by expanding the theory developed for a single neuron, and showing how this relates to a network of neurons. In particular we see that our information discount on single spikes leads us to an identical spiking condition as the SbS framework. Next we briefly discuss the subtleties of learning in our network, and conclude with basic examples showing basic capabilities of the networks.

3.3.1 Theory

In this section we expand our theory from a single neuron developed in section 3.2.1 to a more general multi-dimensional setting. We will start by making analogous considerations as in the one dimensional case. We suppose we have a world model $p_\Theta(\mathbf{x}_t, \mathbf{c}_t)$ which at each point in time t explains an observation $\mathbf{x}_t$ based on some underlying latent cause $\mathbf{c}_t$. The dimension of both the observation and the latent cause can be arbitrary. Furthermore, we have a representation $\hat{\mathbf{c}}_t(\mathbf{s}_t)$ as function of neural activity. Lastly, we have a metabolic cost C as a function of neural activity and other parameters defining the representation. Together, they allow us to define the probabilistic loss $\mathcal{L}$ as a sum of the free energy and expected metabolic cost.

$$\mathcal{L} := \underbrace{\mathcal{F}}_{\text{Function}} + \underbrace{\langle C \rangle_{p_\Psi(\mathbf{s}_t|\mathbf{x}_t)}}_{\text{Metabolic costs}} . \tag{90}$$

Using the previously noted relation (16), we write the probabilistic loss as

$$\mathcal{L} = D_{KL}[p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t) \mid\mid p_\Theta(\hat{\mathbf{c}}_t(\mathbf{s}_t) \mid \mathbf{x}_t)] + \langle C(\mathbf{s}_t) \rangle_{p_\Psi(\mathbf{s}_t|\mathbf{x}_t)} - \ln p_\Theta(\mathbf{x}_t) \tag{91}$$

which we showed (A.1) to have the solution

$$p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t) \propto p_\Theta(\hat{\mathbf{c}}_t(\mathbf{s}_t) \mid \mathbf{x}_t) \exp\left(-C(\mathbf{s}_t)\right) . \tag{92}$$

We observe that the solution balances inference in the model $p_\Theta(\hat{\mathbf{c}}_t(\mathbf{s}_t) \mid \mathbf{x}_t)$ with the cost of activity $\exp\left(-C(\mathbf{s}_t)\right)$.

What remains is to extend our theory of distributions over binary variables to the multi-dimensional case. We reiterate that the following derivations are, to our knowledge, original, and have not been found in the literature.

Distributions over spikes II. First we note a property of a general function $g(\mathbf{s})$ of a vector of binary variables $\mathbf{s} \in \{0,1\}^n$. $g(\mathbf{s})$ can always we written in the following way:

$$g(\mathbf{s}) = \sum_{\mathbf{s}' \in \{0,1\}^n} \delta_{\mathbf{s},\mathbf{s}'} g(\mathbf{s}'). \tag{93}$$

furthermore we note that for binary vectors $\mathbf{s}$ and $\mathbf{s}'$ we can write the Kronecker delta $\delta_{\mathbf{s},\mathbf{s}'}$ as

$$\delta_{\mathbf{s},\mathbf{s}'} = \prod_i \delta_{s^i,s'^i}, \qquad \text{with} \qquad \delta_{s_i,s'_i} = \begin{cases} s^i, & \text{if } s'^i = 1 \\ (1-s^i) & \text{if } s'^i = 0 \end{cases}$$

We will illustrate is that these identities allow us to rewrite $g(\mathbf{s})$ in a form where terms are grouped according to the number of non-zero components in the binary vector, which we will refer to as the "order" $\mathcal{O}(\mathbf{s})$ of $\mathbf{s}$.

For clarity, we show this in the case that $n = 2$ where we demonstrate that g can be broken down into a series of terms, each corresponding to a different order (zeroth, first, and second).

$$\begin{aligned} g(\mathbf{s}) =& (1-s_1)(1-s_2)g(0,0) + s_1(1-s_2)g(1,0) \\ &+(1-s_1)s_2 g(0,1) + s_1 s_2 g(1,1) \\ \\ =& \underbrace{g(0,0)}_{\text{zeroth order}} + \underbrace{s_1(g(1,0)-g(0,0)) + s_2(g(0,1)-g(0,0))}_{\text{first order}} \\ &+ \underbrace{s_1 s_2(g(1,1)-g(1,0)-g(0,1)+g(0,0))}_{\text{second order}}. \end{aligned}$$

Significantly, this decomposition is not limited to this specific example. We note that we can express $g(\mathbf{s})$ in this form for any binary vector, where terms are grouped by the number of non-zero components, or again, the 'order' of the binary vector $\mathbf{s}$. Grouping terms of order two and above allows us to rewrite $g(\mathbf{s})$ as

$$g(\mathbf{s}) = g(\mathbf{0}) + \sum_i s^i(g(\mathbf{e}^i) - g(\mathbf{0})) + \mathcal{O}(\mathbf{s}^2) \tag{94}$$

Next, we can write any (fully-supported) distribution $p(\mathbf{s})$ over $\mathbf{s}$ as

$$p(\mathbf{s}) \propto \exp\big(-\ell(\mathbf{s})\big),$$

where now $\ell(\mathbf{s})$ denotes the negative log-probability of the distribution $p(\mathbf{s})$.

Combining these facts, we get that any (fully-supported) distribution over $\mathbf{s}$ can be written in terms of its log probability as

$$p(\mathbf{s}) \propto \exp\Big(-\sum_i s_i(\ell(\mathbf{e}_i) - \ell(\mathbf{0})) + \mathcal{O}(\mathbf{s}^2)\Big), \tag{95}$$

where $\ell(\mathbf{0})$ gets absorbed by the normalization constant. Again, we define $\ell_\Theta(\mathbf{s}_t, \mathbf{x}_t) := -\ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t(\mathbf{s}_t))$ and the deterministic loss $L(\mathbf{s}_t, \mathbf{x}_t) = \ell_\Theta(\mathbf{s}_t, \mathbf{x}_t) + C(\mathbf{s}_t)$. Subsequently we apply our newfound form (95) to rewrite the solution $p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t)$ of (92) as:

$$p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t) \propto \exp\Big(-\sum_i s_i(L(\mathbf{e}_i, \mathbf{x}_t) - L(\mathbf{0}, \mathbf{x}_t)) + \mathcal{O}(\mathbf{s}^2)\Big). \tag{96}$$

This equation shows that the multi-dimensional spiking condition of the SbS framework gets embedded in the spiking probability $p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t)$.

To continue, we apply our reasoning about discounting the information contained in single spikes events as in section 3.2.1 and redefine the metabolic cost as

$$C \to C + \ln(\delta t) \sum_i s^i, \tag{97}$$

granting us the solution for $p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t)$ as

$$p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t) \propto (\delta t)^{\sum_i s^i} \exp\Big(- \sum_i s_i (L(\mathbf{e}^i, \mathbf{x}_t) - L(\mathbf{0}, \mathbf{x}_t)) + \mathcal{O}(\mathbf{s}^2) \Big). \tag{98}$$

We notice that as δt approaches zero, the probability of simultaneous spike events approaches zero. Specifically, this implies that the probability of spiking for individual neurons *at a specific time* will become independent as the distribution will converge *in distribution* to (see A.2 for motivation)

$$p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t) = \prod_i p_\Psi(s_t^i \mid \mathbf{x}_t) \tag{99}$$

with

$$p_\Psi(s_t^i = 1 \mid \mathbf{x}_t) = \delta t \ \exp\Big(- (L(\mathbf{e}^i, \mathbf{x}_t) - L(\mathbf{0}, \mathbf{x}_t)) \Big) \tag{100}$$

Remarkably, our framework has managed to leverage considerations about metabolic costs to derive a network of neurons with the SbS spiking condition embedded in the exponential.

In the next example we show the extended capabilities of a network of multiple neurons in the objective of tracking a signal.

3.3.2 Example: Two neuron SbS Kalman-like filter

We've extended our theory to a network of neurons performing approximate inference. Here we present a small extension to the previous example of 3.2.2 where a single neuron behaved as a particle Kalman filter.

The world model $p_\Theta(x_t, c_t)$ is defined as before with

$$p_\Theta(x_t \mid c_t) \propto \exp\left(-\frac{\beta_x}{2}(x_t - c_t)^2 \right) \qquad p_\Theta(c_t) \propto \exp\left(-\frac{\beta_c}{2}(c_t - c_p)^2 \right),$$

but we choose our representation $\hat{c}(\mathbf{s}_t)$ differently. The network consists of two neurons, one coding for up, and one for down as given by the decoder matrix $\mathbf{D}$.

$$\hat{c}_t := \mathbf{D} \sum_{t' \le t} \mathbf{s}_{t'} \kappa(t - t') = \mathbf{D}\,(r_t + s_t), \quad \text{with} \quad \mathbf{D} = \begin{pmatrix} d & -d \end{pmatrix}. \tag{101}$$

for d a scalar constant. We choose our metabolic costs as

$$C(\mathbf{s}_t) = (\nu + \ln(\delta t)\ ) \sum_i s^i. \tag{102}$$

Using these definitions we apply our result (100) to derive a SRM $p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t)$

$$p_\Psi(s_t^i = 1 \mid x_t) = \delta t\ \exp(u_t^i - \vartheta_t^i) \tag{103}$$

with

$$\mathbf{u}_t := \mathbf{F}x_t + \mathbf{W}\mathbf{r}_t, \qquad \boldsymbol{\vartheta} := \frac{\mathbf{diag}(\mathbf{W})}{2} + \nu \tag{104}$$

$$\mathbf{F} := \beta_x \mathbf{D}^T, \qquad \mathbf{W} := -\beta_x \mathbf{D}^T\mathbf{D} \tag{105}$$

where x_t is treated as a one-dimensional column vector for the purpose of matrix multiplication.

The resulting dynamics are shown in figure 3.4. The first, striking feature of the network dynamics is the erratic movement of the representation as shown in figure 3.4**b**. The network is now leveraging both neurons in the representation to track the stimulus. A single sample of the representation displays erratic behavior, but figure 3.4**c,e** show that the statistical behavior of the representation is able to match the intricate details of the latent c_t.

We also notice that both the coupled and the decoupled network exhibit a slight gap between the latent cause and the representation. We can explain this by the metabolic cost on the network. Indeed we can show this by considering an extreme example of a network with a very low metabolic cost. Figure 3.5 shows the dynamics of such a network. A single sample of the network seems to show extreme erratic behavior, known in the literature as the 'ping pong' effect, where the representation keeps jumping up and down. However, if we consider the statistical properties shown in figure 3.5**c-f**, we get a near perfect between the latent cause c_t and the representation $\hat{c}_t$.

In summary, we are able to use our theoretic result to build networks of neurons that perform approximate inference on a noisy signal. We have briefly shown the possible effect of a metabolic cost on network behavior, and shown how this might constrain inference.

3.3.3 Learning

In this section we discuss the learning of the parameters of the model $p_\Theta(\mathbf{x}_t, \mathbf{c}_t)$ as well the parameters defining our representation $\hat{\mathbf{c}}$. We will show that again, we can derive learning rules that follow a gradient, as is most commonly done in variational inference.

In our extended framework, learning is in principle analagous to the SbS framework, where we simply attempt to optimize the loss at each moment in time. We do, however, have to proceed with care. The representation parameters enter in the loss both through the representation $\hat{\mathbf{c}}$ as well as the derived network $p_\Psi(\mathbf{s} \mid \mathbf{x})$ causing the gradients to possibly become complex. Futhermore, the SbS framework did not have any notion of model parameters, and we have to show that model optimization makes sense in our current, greedy framework.

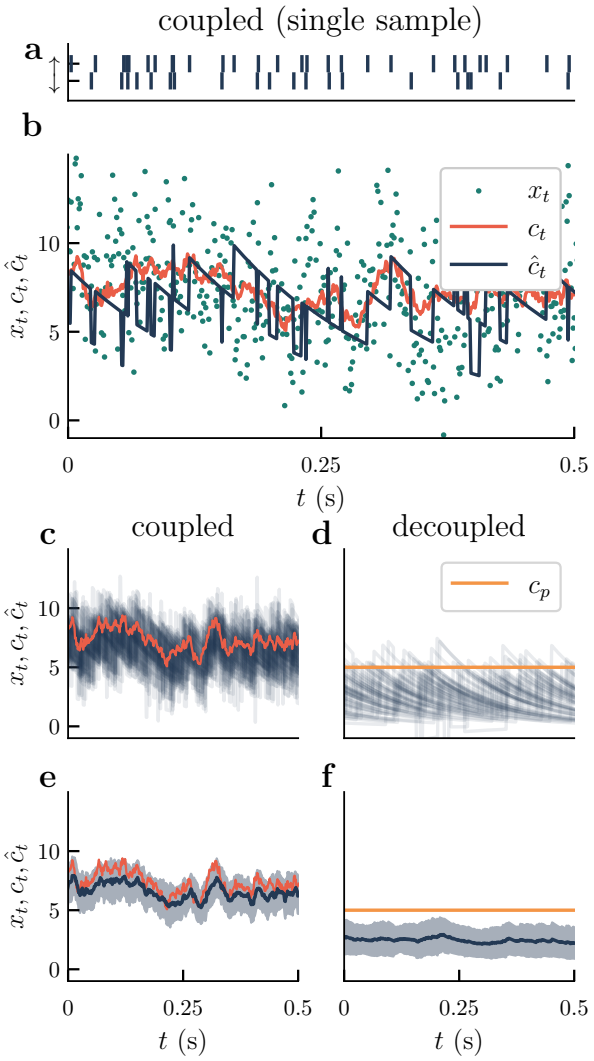

Figure 3.4: A network of two neurons performing approximate inference impaired by metabolic costs. We observe that a single sample trajectory seems to move erratically around the latent c_t, but the aggregate statistics form an excellent representation of the latent cause c_t. Note that the network still exhibits a slight offset incurred by metabolic costs. **a,b**: A single sample of a coupled ($\beta_x \neq 0$) network of two neurons performing approximate inference. The neurons observe the noisy signal x_t generated by adding white noise to an underlying latent cause c_t. The network dynamics dynamics appear to produce a stochastic representation $\hat{c}_t$ of the latent cause. **a** shows rasterplot of the activity for the two neurons over time and **b** showing the latent cause c_t, the stimulus x_t and representation $\hat{c}_t$ produced by the network. **c**: 50 coupled sample trajectories as in **b**, where we can see that the trajectories seem to approximate the latent cause. **d**: 50 uncoupled ($\beta_x = 0$) sample trajectories, representing the prior statistics of the model encoded by the network. **e,f**: statistical properties of the 50 sample trajectories where the bold line indicates the mean of the samples and the shaded regions indicate a single standard deviation. (Simulation parameters given in table 3)

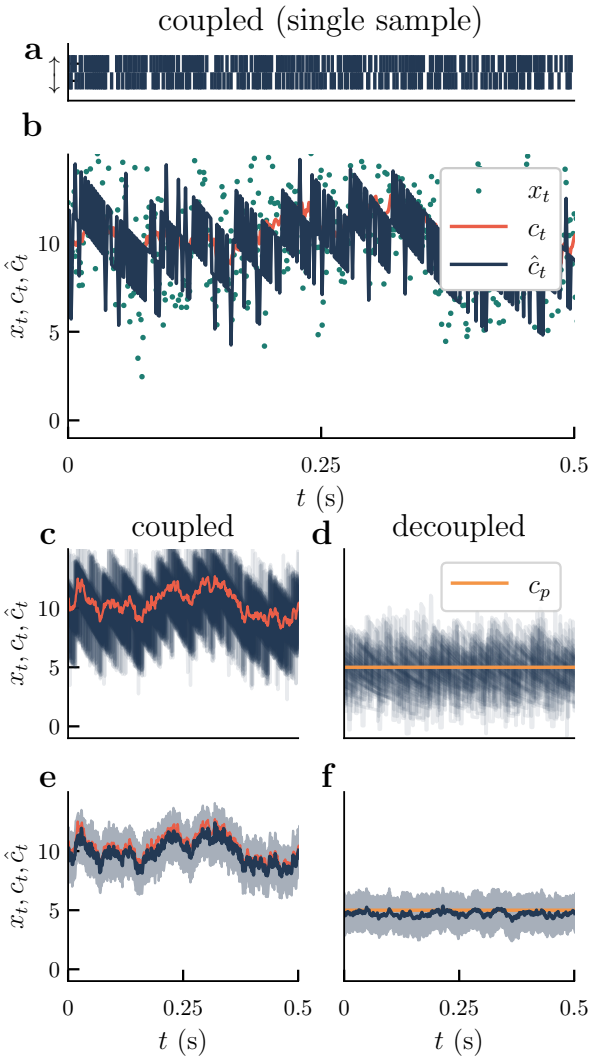

Figure 3.5: A network of two neurons performing approximate inference with a very low metabolic cost. We observe that a single sample trajectory seems to move extremely erratically around the latent cause c_t, but the aggregate statistics form an excellent representation of the latent cause. The network now produces a representation where the mean has a near perfect fit to the underlying latent cause, as it is uninhibited by metabolic costs. **a,b**: A single sample of a coupled ($\beta_x \neq 0$) network of two neurons performing approximate inference. The neurons observe the noisy signal x_t generated by adding white noise to an underlying latent cause c_t. The network dynamics dynamics appear to produce a stochastic representation $\hat{c}_t$ of the latent cause c_t. **a**: rasterplot of the activity for the two neurons over time and **b** showing the latent cause c_t, the stimulus x_t and representation $\hat{c}_t$ produced by the network. **c**: 50 coupled sample trajectories as in **b**, where we can see that the trajectories seem to approximate the latent cause c_t. **d**: 50 uncoupled ($\beta_x = 0$) sample trajectories, representing the prior statistics of the model encoded by the network. **e,f**: statistical properties of the 50 sample trajectories where the bold line indicates the mean of the samples and the shaded regions indicate a single standard deviation. (Simulation parameters given in table 4)

Model parameters Θ. Here we briefly discuss the optimization of the $\mathcal{L}$ with respect to the model parameters Θ. First we will show how we can make sense of the optimization in our greedy setting, then we consider parameters in our observation model p_Θ. In principle, the optimization of the $\mathcal{L}$ with respect to Θ aims to make the world model a better description of the observations $\mathbf{x}$, as is the objective of variational inference. We will discuss here how this fits within our framework.

The derivative of our loss $\mathcal{L}$ with respect to $\theta \in \Theta$ is given by

$$\frac{\partial}{\partial\theta}\mathcal{L} = -\frac{\partial}{\partial\theta}\left\langle \ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t)\right\rangle_{p_\Psi(\mathbf{s}_t|\mathbf{x}_t)} \tag{106}$$

$$= -\frac{\partial}{\partial\theta}\left\langle \ln p_\Theta(\mathbf{x}_t \mid \hat{\mathbf{c}}_t)\right\rangle_{p_\Psi(\mathbf{s}_t|\mathbf{x}_t)} - \frac{\partial}{\partial\theta}\left\langle \ln p_\Theta(\hat{\mathbf{c}}_t)\right\rangle_{p_\Psi(\mathbf{s}_t|\mathbf{x}_t)}. \tag{107}$$

Intuitively, one might want to follow the gradient with respect to θ and indeed, this is what is most commonly done when applying variational inference. However, our entire framework is based on a greedy approach, where we optimize the current loss $\mathcal{L}$. We can therefore wonder how we can reconcile our intuition about slow evolution of parameters with this greedy approach, just as we have done for our representation in section 2.3.

The key to reconciliation is that we can shape the optimization with respect to θ by putting a particular prior on θ, effectively introducing a cost. For example, we can assume the prior that our parameter θ_t, now denoted as a function of time, performs a random walk with variance $1/\gamma$ where γ is a scalar constant. This results in a prior of the form

$$p_\Theta(\theta_t) \propto \exp\left(-\frac{\gamma}{2\delta t}\left(\Delta\theta_t\right)^2\right). \tag{108}$$

With this prior on θ_t we effectively change θ_t to a random variable. As such, to obtain θ_t at any point in time, we have to perform a kind of posterior estimation, for which we will illustrate two techniques for our purposes. But first we write out the posterior for θ_t and get

$$p_\Theta(\theta_t \mid \theta_{t-1}) \propto \exp\left(-\frac{\gamma}{2\delta t}\left(\Delta\theta_t\right)^2 + \ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t)\right), \tag{109}$$

and by assuming δt to be small, we obtain that the posterior of θ_t is given as the discretization of a random walk with drift term proportional to the gradient of $\ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t)$,

$$p_\Theta(\theta_t \mid \theta_{t-1}) \propto \exp\left(-\frac{\gamma}{2\delta t}\left(\theta_t - \theta_{t-1} - \frac{\delta t}{\gamma}\frac{\partial}{\partial\theta}\ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t)\right)^2\right), \tag{110}$$

The first technique is again a particle filter approach where we can sample from the random walk derived above. This particle filter approach is also known under the name of 'plasticity as sampling' and has been explored by [37].

Another method would be to apply MAP estimation, tracking the most likely value of the posterior. We illustrate MAP estimation on the posterior obtained above and obtain

$$\Delta\theta_t = \frac{\delta t}{\gamma}\frac{\partial}{\partial\theta}\ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t). \tag{111}$$

We can connect this learning rule to those derived for the SbS framework. The loss consists of the free energy, which itself is an expectation value of $-\ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t)$. Therefore equation 111 mirrors the derived learning rules (41) for the SbS framework as the derivative of the loss. We note that this approach to plasticity is very common, and is a key component in the popular predictive coding framework of Friston. [26, 51]

Representation parameters. Optimization of $\mathcal{L}$ with respect to the parameters defining the representation balances between improving the model's fit to observations and the incurred cost of its implementation. Deriving learning rules in the probabilistic case might seem difficult as the these parameters appear in the brain $p_\Psi(\mathbf{s} \mid \mathbf{x})$ over which we are taking an expectation value. Nevertheless, we can avoid this caveat and derive learning rules analogously to section 2.3.

We define the representation $\hat{\mathbf{c}}_t(\Lambda, \mathbf{s}_t)$ as a function of $\mathbf{s}_t$ and other parameters, denoted by Ω and would include such objects as the decoding matrix $\mathbf{D}$.

Taking the derivative of $\mathcal{L}$ with respect to a representation parameter $\lambda \in \Lambda$ introduces immediate complications. We can write the probabilistic $\mathcal{L}$ as an expectation value over the distribution $p_\Psi(\mathbf{s}_t, \mathbf{c}_t)$. The expectation value contains a dependency on λ both within the expectation value through the representation $\hat{\mathbf{c}}_t$ and in the distribution $p_\Psi(\mathbf{s}_t, \mathbf{c}_t)$. This implies that taking a derivative with respect to the loss might become unmanageable.

To solve this issue we return to the principles of variational inference. Variational inference splits the optimization of the model and the approximative distribution into two distinct operations. In our framework we have muddied these two steps, but nevertheless we can separate them. We can treat λ as a model parameter, and ignore the dependency in $p_\Psi(\mathbf{c}_t \mid \mathbf{x}_t)$.

In the following examples we have relied on the learning rules given by (111) to optimize our networks. The examples are simple enough that this method is more than sufficient.

We should conclude by nothing that, similar to the learning rules of the decoder in the deterministic SbS framework, these learning rules are not bio-plausible. These learning rules are derived from a top-down perspective, attempting to optimize neural function.

3.3.4 Example: Disentangling a high dimensional stimulus

In this experiment we aim to see if our network can disentangle two simple latent causes from a high-dimensional stimulus $\mathbf{x}_t$. First we introduce how the stimulus $\mathbf{x}_t$ is

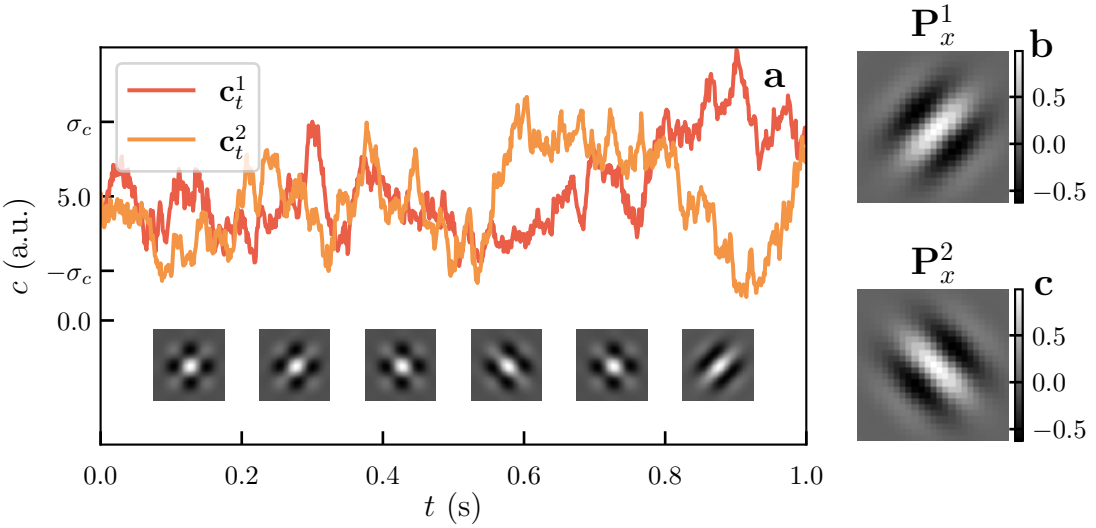

Figure 3.6: The generation of a high-dimensional stimulus. **a**: Depiction of latent causes $\mathbf{c}_t$ and generated stimuli at six different time-points. Each c_t^i is drawn as a sample from an Ornstein-Uhlenbeck process with a mean of 5, amplitude of $\sigma_c = 3$ and a correlation time of 400ms. Observations are generated as a superposition of causes with projections $\mathbf{P}_x^i$. **b,c**: The Gabor projection matrices $\mathbf{P}_x^i$ used for stimulus generation.

generated. Then we define and justify our choice of model $p_\Theta(\mathbf{x}_t, \mathbf{c}_t)$, representation $\hat{\mathbf{c}}$ and metabolic cost C to derive a spiking neural network $p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t)$. Lastly, we present and discuss the results.

Our stimulus is generated as follows: we sample a vector of two causes $\mathbf{c}_t$ independently as two independent Ornstein-Uhlenbeck processes. The stimulus is given at each point in time as a linear sum of the causes projected onto a high dimensional space

$$\mathbf{x}_t = \mathbf{P}_x \mathbf{c}_t \tag{112}$$

An example of the stimulus generation is shown in figure 3.6

In a previous example (3.2.2) we showed that a single neuron can represent the kind of latent dynamics of $\mathbf{c}_t$ underlying the stimulus. Here we investigate if a network of two neurons can learn to disentangle the high-dimensional stimulus and effectively represent the individual causes c_t^i.

We define the model $p_\Theta(\mathbf{x}_t \mid \mathbf{c}_t)$ as

$$p_\Theta(\mathbf{x}_t \mid \mathbf{c}_t) \propto \exp\left(-\frac{\beta_x}{2}(\mathbf{x}_t - \mathbf{D}_x \mathbf{c}_t)^2\right) \qquad p_\Theta(\mathbf{c}_t) \propto \exp\left(-\frac{\beta_c}{2}(\mathbf{c}_t - \mathbf{c}_p)^2\right),$$

with representation $\hat{\mathbf{c}}_t$

$$\hat{\mathbf{c}}_t := \mathbf{D}_c \sum_{t' \leq t} \mathbf{s}_{t'} \kappa(t - t') = \mathbf{D}_c \left[\mathbf{r}_t + \mathbf{s}_t\right], \quad \text{with} \quad \mathbf{D}_c = \begin{pmatrix} d & 0 \\ 0 & d \end{pmatrix}. \tag{113}$$

for d a constant and metabolic cost C

$$C(\mathbf{s}_t) = (\nu + \ln(\delta t)\ ) \sum_i s^i$$

As such, our two neurons model each cause c_t^i separately with identical prior dynamics to the previous example. The notable difference here is the introduction of the matrix $\mathbf{D}_x$, a model parameter indicating how each cause relates to the observations.

Using these definitions we apply our result (100) to derive a SRM $p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t)$

$$p_\Psi(s_t^i = 1 \mid \mathbf{x}_t) = \delta t\ \exp(u_t^i - \vartheta_t^i) \tag{114}$$

with

$$\mathbf{u}_t := \mathbf{F}\mathbf{x}_t + \mathbf{W}\mathbf{r}_t, \qquad \boldsymbol{\vartheta} := \frac{\mathbf{diag}(\mathbf{W})}{2} - \beta_c \mathbf{D}_c \mathbf{c}_p + \nu \tag{115}$$

$$\mathbf{F} := \beta_x \mathbf{D}_c^T \mathbf{D}_x^T, \qquad \mathbf{W} := -\beta_x \mathbf{D}_c^T \mathbf{D}_x^T \mathbf{D}_x \mathbf{D}_c - \beta_c \mathbf{D}_c^T \mathbf{D}_c, \tag{116}$$

The resulting network dynamics, for a random $\mathbf{D}_x$ are shown on the left side of figure 3.7.

For the network to learn and disentangle the two stimuli we have to optimize the $\mathcal{L}$ with respect to $\mathbf{D}_x$.

$$\frac{\partial}{\partial \mathbf{D}_x}\mathcal{L} = -\frac{\partial}{\partial \mathbf{D}_x} \left\langle \ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t) \right\rangle_{p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t)} \tag{117}$$

$$= \beta_x (\mathbf{x}_t - \mathbf{D}_x \mathbf{c}_t)\mathbf{c}_t^T \tag{118}$$

and slowly follow this gradient to optimize the network. The result are shown on the right hand side of figure 3.7. We conclude, that indeed the network is able to reduce the dimensionality of the stimulus and track its underlying latent causes. Furthermore, we observe that the Decoder $\mathbf{D}_x$ produces a representation closely resembling the Gabor patches used in stimulus generation, effectively encoding the stimulus generation itself.

3.3.5 Example: Learning an anticipatory code

In this experiment we aim to see if a network of neurons can learn to leverage underlying temporal regularity in a stimulus $\mathbf{x}_t$. In this example we first introduce the generation of stimulus $\mathbf{x}_t$ is generated. Then we define and justify our choice of model $p_\Theta(\mathbf{x}_t, \mathbf{c}_t)$, representation $\hat{\mathbf{c}}_t$ and metabolic cost C to derive a spiking neural network $p_\Psi(\mathbf{c}_t \mid \mathbf{x}_t)$. And lastly, we present and discuss the results.

We generate a highly predictable stimulus as follows: first we generate a two dimensional cause $\mathbf{c}_t$, the components of which are sine waves with a phase difference of $\pi/2$, then we generate our stimulus $\mathbf{x}_t$ by adding white noise on top of the two causes. The stimulus generation is depicted in figure 3.8.

Thus far, our model priors $p_\Theta(\mathbf{c}_t)$ were independent of the past. All the temporal correlations present in our network were induced by the dynamics of the representation.

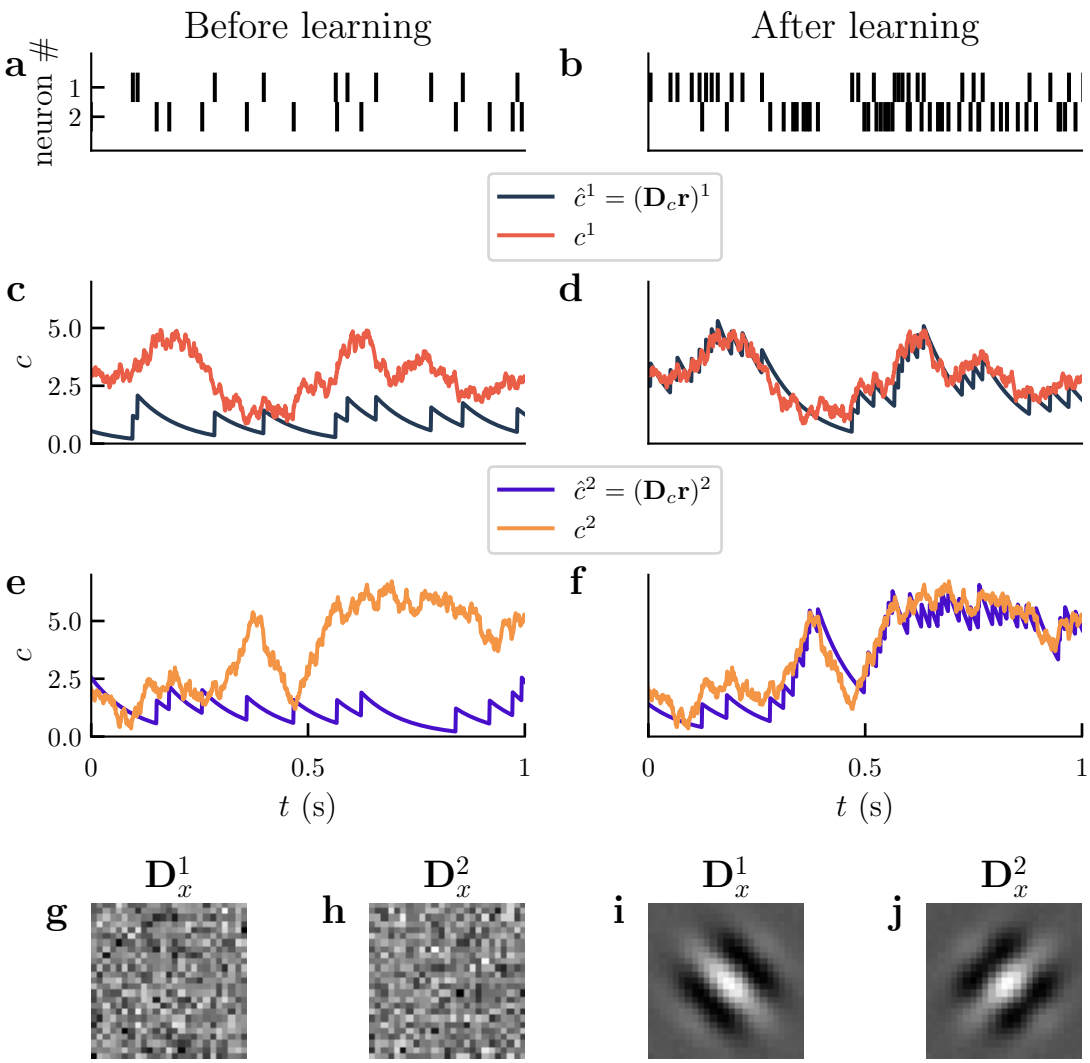

Figure 3.7: A network of two neurons performing inference on a multi-dimensional stimulus. The left side indicates the behaviour of the network before learning a representation $\mathbf{D}_x$ of the stimulus $\mathbf{x}_t$, and the right side shows the behaviour of the network after learning a representation of the stimulus. After learning, the representations $\mathbf{D}_x$ resemble the Gabor projection matrices $\mathbf{P}_x$ used in stimulus generation. **a,b**: raster plots showing the spikes for the two neurons. **c-f**: depiction of the latent cause $\mathbf{c}_t$ used in data generation, as well as the representation $\hat{\mathbf{c}}_t$ produced by the network. **c,e** show that the network has no real representation and does not faithfully reproduce the latent cause, while **d,f** show that through learning $\mathbf{D}_x$ the network is able to track the underlying latent causes used to generate the stimulus $\mathbf{x}_t$. **g,h**: randomized initial representation $\mathbf{D}_x$ before learning. **i,j**: representation $\mathbf{D}_x$ after learning closely resembling the Gabor patches used in the stimulus generation. (Simulation parameters given in table 5)

In this experiment we explicitly add a time dependency to our model $p_\Theta(\mathbf{x}_t, \mathbf{c}_t)$, so that we can attempt to capture the time-dependency of the latent cause $\mathbf{c}_t$. We define the model $p_\Theta(\mathbf{x}_t, \mathbf{c}_t)$ as

$$p_\Theta(\mathbf{x}_t \mid \mathbf{c}_t) \propto \exp\left(-\frac{\beta_x}{2}(\mathbf{x}_t - \mathbf{c}_t)^2\right) \qquad p_\Theta(\mathbf{c}_t) \propto \exp\left(-\frac{\beta_c}{2}(\mathbf{c}_t - \mathbf{D}_p\bar{\mathbf{c}}_t)^2\right),$$

where $\mathbf{D}_p$ is a matrix and $\bar{\mathbf{c}}_t$ is defined as a low-pass version of $\mathbf{c}_t$

$$\bar{\mathbf{c}}_t = \alpha_p \mathbf{c}_t + (1 - \alpha_p)\bar{\mathbf{c}}_{t-1}, \quad \alpha \in (0, 1)$$

Here $\alpha_p = \exp\left(-\frac{\delta t}{\tau_p}\right)$ denotes a time constant. The low pass of the cause $\bar{\mathbf{c}}_t$ will serve as memory for the system, and we suppose that the network will be able to leverage this memory and align the matrix $\mathbf{D}_p$ to match its own representation.

We define our representation $\hat{\mathbf{c}}$ as

$$\hat{\mathbf{c}}_t := \mathbf{D}_c \sum_{t' \leq t} \mathbf{s}_{t'} \kappa(t - t') = \mathbf{D}_c \left[\mathbf{r}_t + \mathbf{s}_t\right], \quad \text{with} \quad \mathbf{D}_c = \begin{pmatrix} d & 0 \\ 0 & d \end{pmatrix}. \tag{119}$$

for d a scalar constant. The representation simply consists of two duplicate copies of the representation used in the one-neuron filter of example 3.2.2.

Furthermore we define the metabolic cost C as

$$C(\mathbf{s}_t) = \ln(\delta t) \sum_i s^i$$

Using these definitions we apply our result (100) to derive the SRM $p_\Psi(\mathbf{s}_t \mid \mathbf{x}_t)$

$$p_\Psi(s_t^i = 1 \mid \mathbf{x}_t) = \delta t \ \exp(u_t^i - \vartheta_t^i) \tag{120}$$

with

$$\mathbf{u}_t := \mathbf{F}\mathbf{x}_t + \mathbf{W}\mathbf{r}_t + \mathbf{W}^p\bar{\mathbf{r}}_t, \qquad \boldsymbol{\vartheta} := \frac{\mathbf{diag}(\mathbf{W})}{2} - \beta_c \mathbf{D}_c \mathbf{c}_p \tag{121}$$

$$\mathbf{F} := \beta_x \mathbf{D}_c^T, \qquad \mathbf{W} := -\beta_x \mathbf{D}_c^T \mathbf{D}_c - \beta_c \mathbf{D}_c^T \mathbf{D}_c, \tag{122}$$

$$\mathbf{W}^p := \beta_c \mathbf{D}_c^T \mathbf{D}_p^T \mathbf{D}_c. \tag{123}$$

These dynamics are similar to the dynamics proposed by [59], which uses the SbS framework to derive a network which simulate Langevin sampling.

We show the resulting network dynamics for a network of 20 neurons with an optimized $\mathbf{D}_p$ in figure 3.9. Here, each neuron a has unique contribution to the encoding of the representation $\hat{\mathbf{c}}_t$ as shown in 3.9**d**. The figure shows that the network is able to track a stimulus over time, and notably, it is able to align the matrix $\mathbf{D}_p$ such that the internal prediction $\mathbf{D}_p\bar{\hat{\mathbf{c}}}_t$ nearly matches the current representation as shown in 3.9**b**.

The network is able to construct a prediction $\mathbf{D}_p\bar{\hat{\mathbf{c}}}_t$ of its internal representation $\hat{\mathbf{c}}_t$. We proceed to investigate whether the network is able to leverage this prediction to

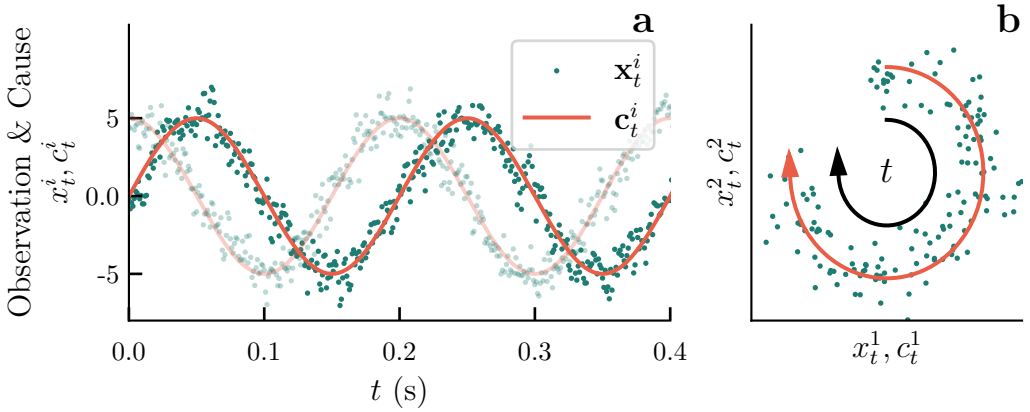

Figure 3.8: The generation of a highly predictable stimulus. **a**: depiction of the components of c_t^i of the latent causes generated as sine waves with a phase difference $\pi/4$. Observations x_t^i are generated as a superposition of c_t^i with white noise. **b**: depiction of the latent causes and the observations over time, showing that the stimulus rotates in time making it highly predictable.

autonomously sustain its highly predictable representation. We tested this by decoupling the network from the observations by setting β_x to zero and the results are shown in figure 3.10. We see that the in the first 0.5 seconds the representation $\hat{\mathbf{c}}_t$ faithfully follows the periodic motion of the underlying latent $\mathbf{c}_t$ as its still coupled to the observations $\mathbf{x}_t$. After decoupling, the network is able to sustain a kind of oscillatory motion in its representation.

We observe in figure 3.10**b** that the network seems to become more regular in its sequential spiking activity. We suspect that the spread of activity before decoupling is a result of the corrections to the representation due to a mismatch between the internal representation and the observations.

In short, we are able to show that we can derive a SRM from a model with explicit past dependence. We showed that the network can use the explicit past dependence to construct a prediction about its own representation. Furthermore, we showed that the network is able leverage this prediction to sustain oscillatory motion.

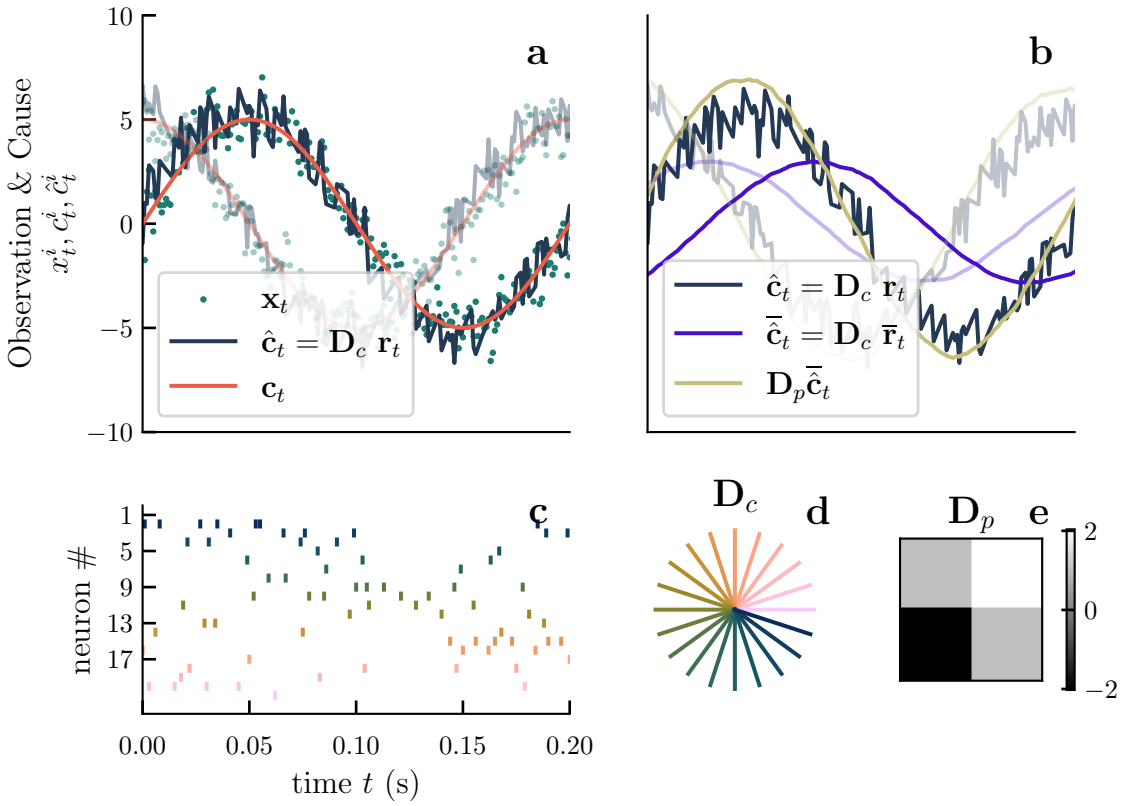

Figure 3.9: A network of 20 neurons tracking a highly predictable stimulus. **a**: depiction of the underlying latent $\mathbf{c}_t$, the observation $\mathbf{x}_t$ and the representation $\hat{\mathbf{c}}_t$ generated by the activity of the network. We see that $\hat{\mathbf{c}}_t$ closely follows the latent $\mathbf{c}_t$. The transparent curves show the second components of the listed quantities. **b**: depiction of the internal representation $\hat{\mathbf{c}}_t$, its low-pass $\overline{\hat{\mathbf{c}}}_t$ and the internal prediction $\mathbf{D}_p\overline{\hat{\mathbf{c}}}_t$. We see that the network is able to align $\mathbf{D}_p$ such that the low pass $\overline{\hat{\mathbf{c}}}_t$ becomes predictive for the representation $\hat{\mathbf{c}}_t$. **c**: raster plot depicting the activity of the twenty neurons over time. **d**: a depiction of $\mathbf{D}_c$ showing how each neuron contributes to the representation $\hat{\mathbf{c}}_t$ with colors corresponding to the raster-plot **c**. **e**: a depiction of the internal prediction matrix $\mathbf{D}_p$. (Simulation parameters given in table 6)

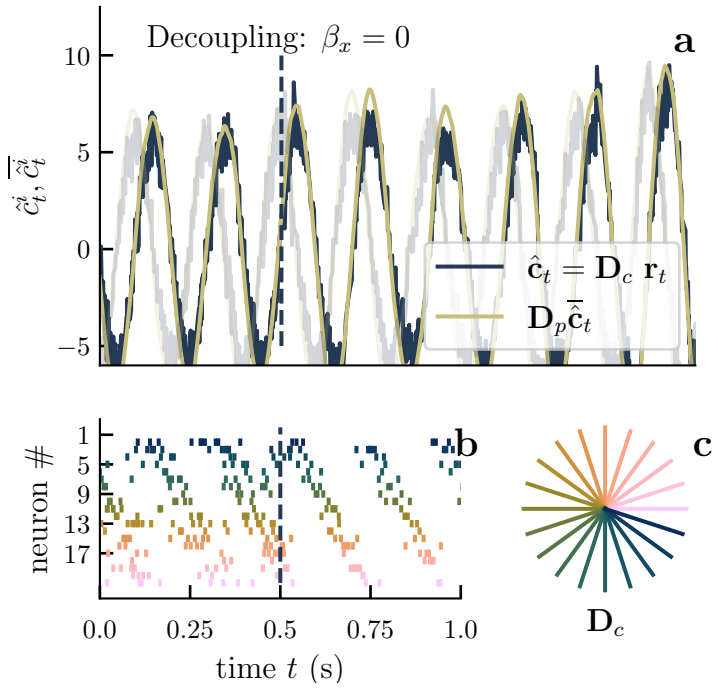

Figure 3.10: Activity of a network before and after decoupling from observations. At time $t = 0.5s$ the network will be blind from observations as we set the precision β_x to zero. The subsequent activity is a result of the network attempting to follow its own internal representation, inducing a feedback loop resulting in autonomously sustained activity. **a**: depiction of the representation $\hat{\mathbf{c}}_t$ and the networks internal prediction $\mathbf{D}_p\overline{\hat{\mathbf{c}}}_t$. **b**: raster plot depicting the activity of the neurons over time. Note the diagonal pattern in the raster plot implying sequential activity of the neurons. **c**: a depiction of $\mathbf{D}_c$ showing how each neuron contributes to the representation $\hat{\mathbf{c}}_t$ with colors corresponding to the raster-plot **b**. (Simulation parameters given in table 6)

3.4 Relation to predictive coding

In this section we discuss the relation of our theory to that of predictive coding. Predictive coding is a term that has increasingly been used in theoretical neuroscience. Unfortunately, the literature has not been consistent in its use. In this section we will both describe the relation of our extended framework to the two main interpretations as presented by [2] and [51].

3.4.1 Predictive coding as a motif

The fundamental idea of predictive coding is that the function of neurons is to respond to a disparity between a stimulus and a prediction, and this disparity is called the prediction error. Predictive coding in this sense supposes that the task of the neuron is to minimize prediction error. This way, the neuron only responds if there is a change in either the stimulus or the prediction, saving valuable resources if the environment is regular.

The idea is conceptually neat and general, however, with great generality comes confusion. As Aitchison and Lengyel [2] point out, what can be understood as a prediction error is not so clearly defined.

To illustrate this we turn to the SbS framework, where we can easily interpret the objective of the neurons. In the SbS framework, the neurons quite literally encode the prediction error. Indeed, if we consider the representation $\hat{\mathbf{x}}$ of a stimulus $\mathbf{x}$ to be a prediction, then, as the membrane potential of the SbS framework is responsive to a difference between the representation and the stimulus, the SbS framework is implementing predictive coding. For this reason, one of the original papers on the SbS frame titled "Predictive Coding of Dynamical Variables in Balanced Spiking Networks" [13] has been framed in the context of predictive coding. Because of the interpretability of the membrane potential as an explicit error computation, the SbS framework has been used to investigate the biological signatures of predictive coding [41].

Unfortunately, our extension to the SbS framework muddies the clear interpretation somewhat. Our extended framework does not simply compute the difference between a representation and a stimulus, but also between the representation and the prior, producing a sample of a posterior distribution. And so, while it is clear that our network performs an error computation, it is not immediately explained as the difference between a prediction and an observation. If we want to understand this function as predictive coding, we have to again, counter-intuitively, view the representation itself as a prediction.

Error computations are incredibly useful and arise from various theories of neural function, including the one in this work. However, as our work already shows, the interpretation of the error computation is not universal. Aitchison and Lengyel [2] propose a solution to the generality of this definition. They propose to view predictive coding and error computations in the light of motifs [3]. A motif is a pattern which seems to occur in various parts of a system. Thereby viewing the error computation

itself as emerging in service of various goals. This proposal makes considerable sense in our framework, as we can derive error computations, but with different interpretations.

3.4.2 Predictive coding of Rao, Ballard & Friston

Another popular use of the term predictive coding is to refer to the seminal work of Rao and Ballard titled "Predictive Coding in the Visual Cortex" [57] and the later generalizations of this publication by Friston et al [26, 27, 28]. The publication illustrates one of the first principled derivations of bio-plausible neural networks. Besides its capabilities, which we will not discuss here, there are two notable features of the network. The first is that the networks are rate based, meaning that the concept of a spike has been abstracted. Secondly, the neurons perform an error computation, fitting with the characterization of predictive coding as a motif discussed above.

Here we showcase how one can use the conceptual framework put forward in this work to reinterpret this of the predictive coding framework. We will first showcase a concise derivation of the predictive coding framework following the work of Millidge et al. [51]. Subsequently, we show that we can also derive the same dynamics from defining a functional objective, a representation and a metabolic cost, as per the spirit of the SbS framework. We hope that this section provides conceptual insight by providing another perspective on predictive coding.

Derivation. The following derivation is a summary of the derivation given in the work Millidge et al. [51]. In a nutshell, predictive coding is obtained from performing approximate MAP estimation[4] using variational methods and a particular choice of model and approximate distribution. Here we explain this in more detail.

To use variational inference, one needs to describe its model of the observations $p_\Theta(\mathbf{x}, \mathbf{c})$, and an approximate posterior $p_\Phi(\mathbf{c} \mid \mathbf{x})$. To obtain predictive coding, we assume $p_\Theta(\mathbf{x}, \mathbf{c}) = p_\Theta(\mathbf{x} \mid \mathbf{c}) p_\Theta(\mathbf{c})$ to be a product of normal distributions

$$p_\Theta(\mathbf{x} \mid \mathbf{c}) = \mathcal{N}(\mathbf{x}; f(\boldsymbol{\theta}_x \mathbf{c}), \mathbf{I}\sigma_x^2) \tag{124}$$

$$p_\Theta(\mathbf{c}) = \mathcal{N}(\mathbf{c}; g(\boldsymbol{\theta}_c \mathbf{r}'), \mathbf{I}\sigma_c^2) \tag{125}$$

Here f, g are arbitrary differentiable functions, $\mathbf{r}', \boldsymbol{\theta}_x, \boldsymbol{\theta}_c, \sigma_x, \sigma_c \in \Theta$ are parameters of the model and $\mathcal{N}$ denotes a normal distribution, implying here that $\mathbf{x}$ is normally distributed around $f(\boldsymbol{\theta}_x \mathbf{c})$ with covariance $\mathbf{I}\sigma_x$.

Next we make the assumption that our approximate posterior is given by a dirac-delta $p_\Phi(\mathbf{c} \mid \mathbf{x}) = \delta(\mathbf{c} - \mathbf{r})$ with center $\mathbf{r}$.

These definitions allow us to write down the free-energy

$$\mathcal{F}[p_\Phi, p_\Theta] = \langle -\ln p_\Theta(\mathbf{x}, \mathbf{c}) \rangle_{p_\Phi(\mathbf{c}|\mathbf{x})} - \langle -\ln p_\Phi(\mathbf{c} \mid \mathbf{x}) \rangle_{p_\Phi(\mathbf{c}|\mathbf{x})} \tag{126}$$

[4] In the literature one often refers to a slightly more involved technique of the 'laplace approximation' the result is the same.

Now we assume the entropy of the dirac-delta to be zero[5] and since the entropy of the dirac-delta is zero we have

$$\mathcal{F}(p_\Phi, p_\Theta) = \langle -\ln p_\Theta(\mathbf{x}, \mathbf{c}) \rangle_{p_\Phi(\mathbf{c}|\mathbf{x})} \tag{127}$$

$$= \frac{1}{2\sigma_x^2}(\mathbf{x} - f(\boldsymbol{\theta}_x \mathbf{r}))^T \boldsymbol{\Sigma}_x^{-1}(\mathbf{x} - f(\boldsymbol{\theta}_x \mathbf{r})) \tag{128}$$

$$+ \frac{1}{2\sigma_c^2}(\mathbf{r} - g(\boldsymbol{\theta}_c \bar{\mathbf{r}}))^T \boldsymbol{\Sigma}_c^{-1}(\mathbf{r} - g(\boldsymbol{\theta}_c \bar{\mathbf{r}})) + \text{const}(\sigma_x, \sigma_c) \tag{129}$$

then finally, to derive dynamics, the predictive coding framework assumes that parameters follow the negative gradient of the free-energy, granting dynamics in continuous time.

$$\frac{d\mathbf{r}}{dt} = -\frac{\partial \mathcal{F}}{\partial \mathbf{r}} \tag{130}$$

essentially performing approximate MAP estimation as discussed in 1.4. Doing this for all variables yields the dynamics of the predictive coding framework, however we focus on the center $\mathbf{r}$ of our approximative distribution $p_\Phi(\mathbf{c} \mid \mathbf{x})$. The center is interpreted as the instantaneous rate of the neuron, which implies that the firing

$$\frac{d\mathbf{r}}{dt} = \frac{1}{\sigma_x^2}(\underbrace{\mathbf{x} - f(\boldsymbol{\theta}_x \mathbf{r})}) f'(\boldsymbol{\theta}_x \mathbf{x}) \boldsymbol{\theta}_x - \frac{1}{\sigma_c^2}(\underbrace{\mathbf{r} - g(\boldsymbol{\theta}_c \bar{\mathbf{r}})}) \tag{131}$$

where we see that the dynamics are given as the sum of two error computations indicated by the brackets. These error computations arise as a consequence of the network attempting to balance the observation model $p_\Theta(\mathbf{x} \mid \mathbf{c})$ with the prior $p_\Theta(\mathbf{c})$ both of which are Normal distributions.

Reinterpreting predictive coding. There are strong similarities between the previous derivation and the foundations of the SbS framework. Here we show that we can derive equivalent dynamics of the predictive coding framework by defining a functional objective, a representation and a metabolic cost on the representation. Note that the following derivation is done for discrete time-steps.

We begin by defining function by the free energy, setting approximate inference with respect to a model $p_\Theta(\mathbf{x}_t, \mathbf{c}_t)$ as the objective. Here we make the assumption that the inference will be performed by an approximative distribution $p_\Phi(\mathbf{c}_t \mid \mathbf{x}_t)$ equal to a Dirac delta with center $\hat{\mathbf{c}}_t$

$$p_\Phi(\mathbf{c}_t \mid \mathbf{x}_t) = \delta(\mathbf{c}_t - \hat{\mathbf{c}}_t) \tag{132}$$

Next we have to define the neural representation $\hat{\mathbf{c}}_t$. We choose to define it as a function of a vector $\mathbf{r}_t$ denoting the firing rate of a population of neurons.

[5] It is strictly undefined, but here it is assumed as the dirac delta distribution carries no uncertainty.

Before we introduce a metabolic cost, lets consider what our framework implies. The functional component of our loss will be given as the free energy with an approximative distribution given by a Dirac delta

$$\mathcal{F}[p_\Phi, p_\Theta] = -\ln p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t(\mathbf{r}_t)) \tag{133}$$

implying that the minimization of $\mathcal{L}$ is simply performing *exact* MAP estimation of $p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t(\mathbf{r}_t))$ with respect to the firing rate $\mathbf{r}_t$.

We argue that the organism cannot change the firing rate of a neural population arbitrarily quickly. We therefore regularize the rate of change of the firing rate $\mathbf{r}_t$ by introducing a familiar metabolic cost

$$C(\mathbf{r}) = \frac{\gamma}{2\delta t}\|\Delta \mathbf{r}_t\|^2, \tag{134}$$

for γ again a constant and δt the discretization timescale. We have the loss $\mathcal{L} = \mathcal{F} + C$ and minimization implies the dynamics

$$\Delta \mathbf{r}_t = -\frac{\delta t}{\gamma}\frac{\partial \mathcal{F}[p_\Phi, p_\Theta]}{\partial \mathbf{r}_t} \tag{135}$$

obtaining the discretized version of the predictive coding dynamics 130. Now we conclude by choosing the model $p_\Theta(\mathbf{c}_t \mid \mathbf{x}_t)$ as in the predictive coding framework, given by equations (124) and (125), and choosing a trivial representation $\hat{\mathbf{c}}_t(\mathbf{r}_t) = \mathbf{r}_t$. This way we obtain identical, albeit discrete, dynamics from a different perspective.

We see that indeed, by defining function, neural representation and metabolic cost, we can derive identical dynamics to the predictive coding framework. The benefit is that unlike the original derivations, it is clear where the conceptual distinction between model and brain lies. This distinction may seem unnecessary, as in reality, the barrier between model and brain might not be present. Conceptually, however, one desires to separate the networks and the function they perform. This way one can possibly isolate the signatures arising from a specific kind of representation.

4 Discussion

In summary, this work provides a novel approach to derive bio-plausible spiking neural networks from the objective of performing approximate Bayesian inference. The networks perform approximate inference by construction a sampling based representation. We have demonstrated the capability of these networks to carry out simple functions, such as imitating a Kalman filter. Additionally, we have illustrated the close connection between our framework and the theory of predictive coding.

Given that our study is both in its infancy and highly conceptual, several questions remain open. We will briefly discuss the connection between our framework and the Bayesian brain hypothesis. We also intend to delve into our framework's relevance to experiments and discuss the framework's limitations. Finally, we will conclude by suggesting potential directions for future research.

4.1 Are we Bayesian?

We derived our networks from the objective of performing approximate Bayesian inference. Approximate implies that we are unable to perfectly compute the posterior distribution $p_\Theta(\mathbf{c} \mid \mathbf{x})$. A significant question arises: do our networks carry out a Bayesian computation in any form, even if it is in relation to a different model? The answer is indeed, and has implicitly already been shown. Our extended framework derives its dynamics from minimizing the KL-divergence with respect to the distribution

$$p_\Theta(\hat{\mathbf{c}}_t(\mathbf{s}_t) \mid \mathbf{x}_t) \exp(-C(\mathbf{s}_t)) \tag{136}$$

which itself is the posterior distribution of

$$p_\Theta(\mathbf{x}_t, \hat{\mathbf{c}}_t(\mathbf{s}_t)) \exp(-C(\mathbf{s}_t)) \tag{137}$$

with an effective prior on the network activity

$$p_\Theta(\hat{\mathbf{c}}_t(\mathbf{s}_t)) \exp(-C(\mathbf{s}_t)). \tag{138}$$

This showcases that our spiking networks can be seen as the result of a Bayesian computation. Simultaneously, it showcases a potential problem with the Bayesian brain hypothesis: our networks can be considered Bayesian, no matter how constrained their dynamics. Indeed, the Bayesian computation underlying our networks infer neural activity, but the resulting neural activity may be significantly constrained by cost of producing the activity. We have seen this fact in our example of the single neuron Kalman-like filter shown in figure 3.3. Here, the discrepancy between the latent cause and representation was induced by a higher metabolic cost, showing that the single neuron could not generate enough spikes to build an accurate representation.

The vivid illustration that an arbitrarily constrained network can still be considered Bayesian should imbue us with caution to the Bayesian brain hypothesis. It highlights the insufficiency of relying solely on Bayesian inference as a guiding principle for brain function. To establish brain function on principled grounds, it becomes necessary to incorporate additional constraints.

4.2 Relation to experiment

Theoretical work like ours is only useful insofar it can relate to experiments. Thus, we need to evaluate if our current efforts are substantial enough to derive meaningful predictions. Unfortunately, we believe that the theory is too immature to make significant claims. This inadequacy is in part due to the oversimplified character of our derived networks, and while we manage to construct biologically plausible neural networks, their complexity pales in comparison to that of the brain. Additionally, our framework's flexibility, partly attributed to its Bayesian foundation, allows us considerable modeling freedom. Therefore, it is important to exercise caution when interpreting our results.

Nevertheless, we believe the network is well suited to contextualize experimental phenomena, in light of a Bayesian, sampling-based approach. One example of such an effort can be seen in a precursor to our work [49], which provides a novel interpretation for locomotion's surprising effect on neural activity in the visual cortex. The interpretation suggests that locomotion forecasts changes in the visual field, therefore accounting for a part of the observations, a process referred to as 'explaining away' [55], a distinctly Bayesian feature. We suspect that the main benefit of our framework lies in this direction, where we use the framework to contextualize phenomena and provide a clean functional interpretation.

Although the theory we propose here may not yield immediate predictions, it has the potential to propose alternative interpretations to established theories regarding brain function. For instance, the predictive coding framework of Rao, Ballard & Friston [57, 26, 27, 28] has become both extremely influential and controversial [1, 50]. Our extended SbS framework can be used to contrast the framework of predictive coding, which suggests a parameter-based representation of uncertainty. One could attempt this, for instance, by constructing networks that aim to perform the same inference task, isolating the signatures associated to each framework.

4.3 Limitations

The SbS framework and our extension both rely on significant assumptions. Here we elaborate on the apparent limitations.

Greedy formulation. In our theories we have a loss function, either L or $\mathcal{L}$, which is optimized at each point in time. At first glance this might seem counter-intuitive and sub-optimal as we disregard the performance of our networks in the future. But here we point out that inference itself is agnostic to future, unknown observations, and can be seen as a greedy procedure, integrating present observations with prior knowledge.

We illustrate this point by referring back to the theory of variational inference. Bayesian inference desires to find the posterior $p_\Theta(\mathbf{c} \mid \mathbf{x})$ of a generative model $p_\Theta(\mathbf{x}, \mathbf{c})$, given observations $\mathbf{x}$. The theory of variational inference shows that the free energy

$$\mathcal{F}[p_\Phi(\mathbf{c} \mid \mathbf{x}), p_\Theta(\mathbf{x}, \mathbf{c})] \tag{139}$$

is minimized for $p_\Phi(\mathbf{c} \mid \mathbf{x}) = p_\Theta(\mathbf{c} \mid \mathbf{x})$. Showing that indeed, general inference can be seen as the greedy optimization of a loss function. Therefore, the greedy formulation is not a limitation from the perspective of performing inference.

Discrete change of variables. A key part of our framework is the choice of spike-based representation $\hat{\mathbf{c}}(\mathbf{s})$. We used the representation as a substitution of the cause $\mathbf{c}$ of the generative model $p_\Theta(\mathbf{x}, \mathbf{c})$. The problem is that it is uncertain whether this makes much sense mathematically. One can point out that we are trying to capture a continuum by a discrete set of points, in our case the different spiking events.

Our approach introduces complications at two different stages of our derivations. The first complication concerns our framing of variational inference in terms of spikes. One can understand this as performing variational inference with the approximative distribution $p_\Phi(\mathbf{c} \mid \mathbf{x})$ defined as

$$p_\Phi(\mathbf{c} \mid \mathbf{x}) = \sum_{\mathbf{s}} \delta(\mathbf{c} - \hat{\mathbf{c}}(\mathbf{s})) p_\Psi(\mathbf{s} \mid \mathbf{x}) \tag{140}$$

The problem with this formulation, is that the computations relating to the KL-divergence are not well defined. Our approach is simply the consequence of naively attempting this calculation. Note that in the case that $\mathbf{s}$ were to be continuous and the mapping $\hat{\mathbf{c}}$ were to be a diffeomorphism, this formulation would lead to a standard change of variables, motivating its formulation.

The second complication concerns the utility of the generative model after the change of variables. Take the Kalman filter as an example. The Kalman filter itself might be a highly useful way to integrate observations. After a change of variables to a spike-based representation, the model $p_\Theta(\mathbf{x}, \hat{\mathbf{c}}(\mathbf{s}))$ is no longer a Kalman filter. The latent structure of the model is now expressed in terms of discrete points of neural activity. It is currently hard to get a handle on the efficacy of the resulting model to perform inference.

Nevertheless, we have to note that general principles of variational inference still apply to distributions of discrete variables. As such, once we have crossed to a spike-based representation, all of our reasoning around optimization of the model and representation remain sound.

Deriving dynamics and bio-plausibility. The exact structure of our network is a product of the choice of functional objective, representation and metabolic cost. The choices made in this work allowed us significant simplifications. For instance, our exposition on the deterministic SbS framework had its functional functional objective defined as a square loss (22) and its representation a linear sum over neural responses (24). This combination allowed us to explicitly write out the spiking condition and obtain network dynamics, fitting the bio-plausiblity benchmark of the spike response model. However, once the functional objective or the neural representation increases in its complexity, we can no longer explicitly write out the spiking condition and derive

network dynamics. This could, for instance, happen if we desire representations as a non-linear function of network activity. Once this occurs we encounter two immediate issues.

The first issue is that we are required to introduce approximations to the spiking condition of our framework to derive network dynamics. The spiking condition can be written as a linear difference between the loss

$$L(s_t = 1) - L(s_t = 0) \tag{141}$$

which we are no longer able to compute with non-linear representations. To combat this we require approximation techniques. Our simplest proposal would be to approximate the finite difference in s_t by a derivative

$$L(s_t = 1) - L(s_t = 0) \approx \frac{\partial}{\partial s_t} L(s_t = 0)$$

treating s_t as a continuous variable and allows us to write the dynamics in terms of the derivative of the representation. We note that for choices of function, neural representation, and metabolic costs, as defined in this work, the approximation leads to near identical dynamics.

The second issue relates to the benchmark of bio-plausibility. We set the benchmark as the spike response model, which writes its membrane potentials as a linear sum of neural responses. If we turn to dynamics derived from non-linear representations, we will most certainly fail to meet this benchmark. Therefore, to address further bio-plausibility, one is required to revisit the benchmark.

4.4 Outlook

The aim of this work was to provide scaffolding to explore ideas in theoretical neuroscience. Our framework provides a novel method to derive spiking neural networks from the objective of performing approximate Bayesian inference. While this may be interesting in its own right, the real utility lies in its application in future research. Here we briefly suggest the possible research directions.

General sampling based inference Our belief about the utility of our networks is directly related to our belief about the utility of general sampling based inference. Indeed, if we had little faith that sampling based inference could match the ability of the brain, then our research would seem futile. We therefore suggest future research in the general capabilities of sampling based inference. Luckily, the interest in particle filters remains strong [63, 61], with many applications in fields such engineering and medicine.

Especially relevant would be to study the capabilities of hierarchical particle filters, as the brain is deeply hierarchical in nature. While there are some studies on the subject [19], the literature remains sparse.

Theoretical foundations Our theory attempts to perform approximate inference with a spike-based representation. The representation offers several unique challenges which make it hard to reason about its efficacy in performing inference.

The first concerns the change of variables (140) discussed in the limitations. We naively applied the change of variablies to obtain a discrete approximative distribution. Our results showed that this naive application allowed us to effectively perform approximate inference with discrete variables. To further understand coding in the brain, we would require a more general theory of discrete approximations to continuous distributions.

The second concerns more complex models and non-linear representations. Our theory showed that spike-based inference can always be written in terms of the spiking condition of the SbS framework. The spiking condition, however, is possibly no longer computable if we consider more sophisticated models and non-linear representations. Development of further theory could, for instance, facilitate a deeper understanding of the role of non-linear computations in the brain such as those occurring in the dendrites.

Applications to decision making One framework could provide a novel approach to connect neurobiology, to the common link between Bayesian inference and decision making [10]. Many standard models of decision making, such as the drift diffusion model, can be formulated in terms of a sampling-based approach to Bayesian inference [14]. Our model could therefore help bridge the gap between models of decision making and models of neural dynamics.

Applications to neurobiology Our theory, as an extension of the SbS framework, offers potential new insights into neurobiology. Leveraging the Bayesian foundation of our approach, we can naturally derive neural networks that effectively balance top-down predictions and bottom-up stimuli, surpassing the capabilities of the existing SbS framework. Future work is required to understand how the insights regarding plasticity, derived from the deterministic SbS framework, can be applied in this more generalized setting that incorporates top-down predictions. Furthermore, as previously stated, we believe our framework should be used to contrast interpretations of other theories of neural function, such as predictive coding, particularly in the context of interpreting mismatch experiments.

In conclusion, while our extended framework is still in its infancy and requires careful interpretation, it presents a unique perspective for understanding neurobiological phenomena through a Bayesian lens. Our framework is sufficiently different from established theories, such as predictive coding, opening potential avenues for contrasting them. As this theory matures, it may offer alternative perspectives on brain function and further elucidate complex neural processes.

Acknowledgments

I would like to express my deepest appreciation for all the kind people who have not only helped me bring this project to a conclusion, but also helped me gain great perspective. In particular, I want to thank my dear friends Leander Post, Jeroen Smulders, Andreas Schneider, and Jan Lehrke for their kind and insightful support. I also want to thank Fabian Mikulasch and Lucas Rudelt for facilitating this project and guiding me through my foray into wild and the highly interdisciplinary field of Neuroscience. Furthermore, I want to express my appreciation to Viola Priesemann, whose passion is greatly reflected in her group, and whose perspective on science will remain of great value to me. Lastly, I want to express my gratitude towards Greg Stephens and Clelia de Mulatier, my local supervisors in Amsterdam, who helped me facilitate this project.

References

[1] Miguel Aguilera et al. "How Particular Is the Physics of the Free Energy Principle?" In: *Physics of Life Reviews* 40 (Mar. 1, 2022), pp. 24–50. ISSN: 1571-0645. DOI: 10.1016/j.plrev.2021.11.001.

[2] Laurence Aitchison and Máté Lengyel. "With or without You: Predictive Coding and Bayesian Inference in the Brain". In: *Current Opinion in Neurobiology. Computational Neuroscience* 46 (Oct. 1, 2017), pp. 219–227. ISSN: 0959-4388. DOI: 10.1016/j.conb.2017.08.010.

[3] Uri Alon. "Network Motifs: Theory and Experimental Approaches". In: *Nature Reviews Genetics* 8.6 (6 June 2007), pp. 450–461. ISSN: 1471-0064. DOI: 10.1038/nrg2102.

[4] David GT Barrett, Sophie Denève, and Christian K Machens. "Optimal Compensation for Neuron Loss". In: *eLife* 5 (Dec. 9, 2016). Ed. by Frances K Skinner, e12454. ISSN: 2050-084X. DOI: 10.7554/eLife.12454.

[5] Guillaume Bellec et al. "Long Short-Term Memory and Learning-to-learn in Networks of Spiking Neurons". In: *Advances in Neural Information Processing Systems*. Vol. 31. Curran Associates, Inc., 2018. URL: https://proceedings.neurips.cc/paper/2018/hash/c203d8a151612acf12457e4d67635a95-Abstract.html.

[6] Guillaume Bellec et al. "A Solution to the Learning Dilemma for Recurrent Networks of Spiking Neurons". In: *Nature Communications* 11.1 (Dec. 2020), p. 3625. ISSN: 2041-1723. DOI: 10.1038/s41467-020-17236-y.

[7] Jan Benda and Andreas V. M. Herz. "A Universal Model for Spike-Frequency Adaptation". In: *Neural Computation* 15.11 (Nov. 1, 2003), pp. 2523–2564. ISSN: 0899-7667. DOI: 10.1162/089976603322385063.

[8] Pietro Berkes and Laurenz Wiskott. "Slow Feature Analysis Yields a Rich Repertoire of Complex Cell Properties". In: *Journal of Vision* 5.6 (July 20, 2005), p. 9. ISSN: 1534-7362. DOI: 10.1167/5.6.9.

[9] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Vol. 4. 4. Springer.

[10] Sebastian Bitzer et al. "Perceptual Decision Making: Drift-Diffusion Model Is Equivalent to a Bayesian Model". In: *Frontiers in Human Neuroscience* 8 (2014). ISSN: 1662-5161. URL: https://www.frontiersin.org/articles/10.3389/fnhum.2014.00102.

[11] Martin Boerlin and Sophie Denève. "Spike-Based Population Coding and Working Memory". In: *PLOS Computational Biology* 7.2 (Feb. 17, 2011), e1001080. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1001080.

[12] Martin Boerlin, Christian K. Machens, and Sophie Denève. "Predictive Coding of Dynamical Variables in Balanced Spiking Networks". In: *PLOS Computational Biology* 9.11 (Nov. 14, 2013), e1003258. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1003258.

[13] Martin Boerlin, Christian K. Machens, and Sophie Denève. "Predictive Coding of Dynamical Variables in Balanced Spiking Networks". In: *PLoS Computational Biology* 9.11 (Nov. 14, 2013). Ed. by Olaf Sporns, e1003258. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1003258.

[14] Rafal Bogacz et al. "The Physics of Optimal Decision Making: A Formal Analysis of Models of Performance in Two-Alternative Forced-Choice Tasks." In: *Psychological Review* 113.4 (2006), pp. 700–765. ISSN: 1939-1471, 0033-295X. DOI: 10.1037/0033-295X.113.4.700.

[15] Ralph Bourdoukan and Sophie Denève. "Enforcing Balance Allows Local Supervised Learning in Spiking Recurrent Networks". In: *Advances in Neural Information Processing Systems*. Vol. 28. Curran Associates, Inc., 2015. URL: https://proceedings.neurips.cc/paper/2015/hash/3871bd64012152bfb53fdf04b401193f-Abstract.html.

[16] Ralph Bourdoukan et al. "Learning Optimal Spike-Based Representations". In: *Advances in neural information processing systems* 25 (2012).

[17] Wieland Brendel et al. "Learning to Represent Signals Spike by Spike". In: *PLOS Computational Biology* 16.3 (Mar. 16, 2020), e1007692. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1007692.

[18] Nuno Calaim et al. "The Geometry of Robustness in Spiking Neural Networks". In: *eLife* 11 (May 30, 2022). Ed. by Markus Meister et al., e73276. ISSN: 2050-084X. DOI: 10.7554/eLife.73276.

[19] Phani Chavali and Arye Nehorai. "Hierarchical Particle Filtering for Multi-Modal Data Fusion with Application to Multiple-Target Tracking". In: *Signal Processing* 97 (Apr. 2014), pp. 207–220. ISSN: 01651684. DOI: 10.1016/j.sigpro.2013.10.015.

[20] Andy Clark. "Whatever next? Predictive Brains, Situated Agents, and the Future of Cognitive Science". In: *Behavioral and Brain Sciences* 36.3 (June 2013), pp. 181–204. ISSN: 0140-525X, 1469-1825. DOI: 10.1017/S0140525X12000477.

[21] Christine A. Curcio and Kimberly A. Allen. "Topography of Ganglion Cells in Human Retina". In: *Journal of Comparative Neurology* 300.1 (1990), pp. 5–25. ISSN: 1096-9861. DOI: 10.1002/cne.903000103.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum Likelihood from Incomplete Data Via the *EM* Algorithm". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1 (Sept. 1977), pp. 1–22. ISSN: 00359246. DOI: 10.1111/j.2517-6161.1977.tb01600.x.

[23] Sophie Denève and Christian K Machens. "Efficient Codes and Balanced Networks". In: *Nature Neuroscience* 19.3 (Mar. 2016), pp. 375–382. ISSN: 1097-6256, 1546-1726. DOI: 10.1038/nn.4243.

[24] Rodrigo Echeveste et al. "Cortical-like Dynamics in Recurrent Circuits Optimized for Sampling-Based Probabilistic Inference". In: *Nature Neuroscience* 23.9 (9 Sept. 2020), pp. 1138–1149. ISSN: 1546-1726. DOI: 10.1038/s41593-020-0671-1.

[25] József Fiser et al. "Statistically Optimal Perception and Learning: From Behavior to Neural Representations". In: *Trends in Cognitive Sciences* 14.3 (Mar. 2010), pp. 119–130. ISSN: 13646613. DOI: 10.1016/j.tics.2010.01.003.

[26] Karl Friston. "Learning and Inference in the Brain". In: *Neural Networks*. Neuroinformatics 16.9 (Nov. 1, 2003), pp. 1325–1352. ISSN: 0893-6080. DOI: 10.1016/j.neunet.2003.06.005.

[27] Karl Friston. "The Free-Energy Principle: A Unified Brain Theory?" In: *Nature Reviews Neuroscience* 11.2 (2 Feb. 2010), pp. 127–138. ISSN: 1471-0048. DOI: 10.1038/nrn2787.

[28] Karl Friston et al. "Active Inference and Learning". In: *Neuroscience & Biobehavioral Reviews* 68 (Sept. 1, 2016), pp. 862–879. ISSN: 0149-7634. DOI: 10.1016/j.neubiorev.2016.06.022.

[29] Justin L. Gardner. "Optimality and Heuristics in Perceptual Neuroscience". In: *Nature Neuroscience* 22.4 (4 Apr. 2019), pp. 514–523. ISSN: 1546-1726. DOI: 10.1038/s41593-019-0340-4.

[30] Wulfram Gerstner, Raphael Ritz, and J. Leo van Hemmen. "Why Spikes? Hebbian Learning and Retrieval of Time-Resolved Excitation Patterns". In: *Biological Cybernetics* 69.5 (Oct. 1, 1993), pp. 503–515. ISSN: 1432-0770. DOI: 10.1007/BF00199450.

[31] Wulfram Gerstner et al. *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. 1st ed. Cambridge University Press, July 24, 2014. ISBN: 978-1-107-06083-8 978-1-107-63519-7 978-1-107-44761-5. DOI: 10.1017/CBO9781107447615.

[32] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. Adaptive Computation and Machine Learning. Cambridge, Massachusetts: The MIT Press, 2016. 775 pp. ISBN: 978-0-262-03561-3.

[33] J J Hopfield. "Neural Networks and Physical Systems with Emergent Collective Computational Abilities." In: *Proceedings of the National Academy of Sciences* 79.8 (Apr. 1982), pp. 2554–2558. DOI: 10.1073/pnas.79.8.2554.

[34] ET Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

[35] Renaud Jolivet et al. "Predicting Spike Timing of Neocortical Pyramidal Neurons by Simple Threshold Models". In: *Journal of Computational Neuroscience* 21.1 (Aug. 1, 2006), pp. 35–49. ISSN: 1573-6873. DOI: 10.1007/s10827-006-7074-5.

[36] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1 (Mar. 1, 1960), pp. 35–45. ISSN: 0021-9223. DOI: 10.1115/1.3662552.

[37] David Kappel et al. "Synaptic Sampling: A Bayesian Approach to Neural Network Plasticity and Rewiring". In: ().

[38] David C Knill and Whitman Richards. *Perception as Bayesian Inference*. Cambridge University Press, 1996.

[39] David C. Knill and Alexandre Pouget. "The Bayesian Brain: The Role of Uncertainty in Neural Coding and Computation". In: *Trends in Neurosciences* 27.12 (Dec. 2004), pp. 712–719. ISSN: 01662236. DOI: 10.1016/j.tins.2004.10.007.

[40] Ádám Koblinger, József Fiser, and Máté Lengyel. "Representations of Uncertainty: Where Art Thou?" In: *Current Opinion in Behavioral Sciences*. Computational Cognitive Neuroscience 38 (Apr. 1, 2021), pp. 150–162. ISSN: 2352-1546. DOI: 10.1016/j.cobeha.2021.03.009.

[41] Veronika Koren and Sophie Denève. "Computational Account of Spontaneous Activity as a Signature of Predictive Coding". In: *PLOS Computational Biology* 13.1 (Jan. 23, 2017). Ed. by Kim T. Blackwell, e1005355. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1005355.

[42] Anna Kutschireiter et al. "Nonlinear Bayesian Filtering and Learning: A Neuronal Dynamics for Perception". In: *Scientific Reports* 7.1 (1 Aug. 18, 2017), p. 8722. ISSN: 2045-2322. DOI: 10.1038/s41598-017-06519-y.

[43] Ying-Hui Liu and Xiao-Jing Wang. "Spike-Frequency Adaptation of a Generalized Leaky Integrate-and-Fire Model Neuron". In: *Journal of Computational Neuroscience* 10.1 (Jan. 1, 2001), pp. 25–45. ISSN: 1573-6873. DOI: 10.1023/A:1008916026143.

[44] Wei Ji Ma et al. "Bayesian Inference with Probabilistic Population Codes". In: *Nature Neuroscience* 9.11 (11 Nov. 2006), pp. 1432–1438. ISSN: 1546-1726. DOI: 10.1038/nn1790.

[45] David J C MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge university press, 2003.

[46] Leonard A McGee and Stanley F Schmidt. "Discovery of the Kalman Filter as a Practical Tool for Aerospace And". In: ().

[47] Skander Mensi, Richard Naud, and Wulfram Gerstner. "From Stochastic Nonlinear Integrate-and-Fire to Generalized Linear Models". In: *Advances in Neural Information Processing Systems*. Vol. 24. Curran Associates, Inc., 2011. URL: https://proceedings.neurips.cc/paper/2011/hash/82489c9737cc245530c7a6ebef3753ec-Abstract.html.

[48] Fabian A. Mikulasch, Lucas Rudelt, and Viola Priesemann. "Local Dendritic Balance Enables Learning of Efficient Representations in Networks of Spiking Neurons". In: *Proceedings of the National Academy of Sciences* 118.50 (Dec. 14, 2021), e2021925118. DOI: 10.1073/pnas.2021925118.

[49] Fabian A. Mikulasch, Lucas Rudelt, and Viola Priesemann. "Visuomotor Mismatch Responses as a Hallmark of Explaining Away in Causal Inference". In: *Neural Computation* 35.1 (Jan. 1, 2023), pp. 27–37. ISSN: 0899-7667. DOI: 10.1162/neco_a_01546.

[50] Fabian A. Mikulasch et al. "Where Is the Error? Hierarchical Predictive Coding through Dendritic Error Computation". In: *Trends in Neurosciences* 46.1 (Jan. 1, 2023), pp. 45–59. ISSN: 0166-2236. DOI: 10.1016/j.tins.2022.09.007.

[51] Beren Millidge, Anil Seth, and Christopher L. Buckley. *Predictive Coding: A Theoretical and Experimental Review*. July 12, 2022. arXiv: 2107.12979 [cs, q-bio]. URL: http://arxiv.org/abs/2107.12979. preprint.

[52] Beren Millidge, Alexander Tschantz, and Christopher L. Buckley. "Whence the Expected Free Energy?" In: *Neural Computation* 33.2 (Feb. 1, 2021), pp. 447–482. ISSN: 0899-7667, 1530-888X. DOI: 10.1162/neco_a_01354.

[53] Bruno A. Olshausen and David J. Field. "Emergence of Simple-Cell Receptive Field Properties by Learning a Sparse Code for Natural Images". In: *Nature* 381.6583 (6583 June 1996), pp. 607–609. ISSN: 1476-4687. DOI: 10.1038/381607a0.

[54] Daniel Alexander Ortega and Pedro Alejandro Braun. "Information, Utility and Bounded Rationality". In: *Artificial General Intelligence*. Ed. by Jürgen Schmidhuber, Kristinn R. Thórisson, and Moshe Looks. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2011, pp. 269–274. ISBN: 978-3-642-22887-2. DOI: 10.1007/978-3-642-22887-2_28.

[55] Judea Pearl. *Causality*. Cambridge university press, 2009.

[56] Alexandre Pouget et al. "Probabilistic Brains: Knowns and Unknowns". In: *Nature Neuroscience* 16.9 (9 Sept. 2013), pp. 1170–1178. ISSN: 1546-1726. DOI: 10.1038/nn.3495.

[57] Rajesh P. N. Rao and Dana H. Ballard. "Predictive Coding in the Visual Cortex: A Functional Interpretation of Some Extra-Classical Receptive-Field Effects". In: *Nature Neuroscience* 2.1 (Jan. 1999), pp. 79–87. ISSN: 1097-6256, 1546-1726. DOI: 10.1038/4580.

[58] Simo Särkkä and Lennart Svensson. *Bayesian Filtering and Smoothing*. Vol. 17. Cambridge university press, 2023.

[59] Cristina Savin and Sophie Denève. "Spatio-Temporal Representations of Uncertainty in Spiking Neural Networks". In: ().

[60] J. David Smith. "The Study of Animal Metacognition". In: *Trends in Cognitive Sciences* 13.9 (Sept. 1, 2009), pp. 389–396. ISSN: 1364-6613. DOI: 10.1016/j.tics.2009.06.009.

[61] Simone Carlo Surace, Anna Kutschireiter, and Jean-Pascal Pfister. "How to Avoid the Curse of Dimensionality: Scalability of Particle Filters with and without Importance Weights". In: *SIAM Review* 61.1 (Jan. 2019), pp. 79–91. ISSN: 0036-1445. DOI: 10.1137/17M1125340.

[62] Pietro Vertechi, Wieland Brendel, and Christian K Machens. "Unsupervised Learning of an Efficient Short-Term Memory Network". In: *Advances in Neural Information Processing Systems*. Vol. 27. Curran Associates, Inc., 2014. URL: https://proceedings.neurips.cc/paper/2014/hash/333222170ab9edca4785c39f55221fe7-Abstract.html.

[63] Xuedong Wang et al. "A Survey of Recent Advances in Particle Filters and Remaining Challenges for Multitarget Tracking". In: *Sensors* 17.12 (12 Dec. 2017), p. 2707. ISSN: 1424-8220. DOI: 10.3390/s17122707.

A Variational Inference

A.1 Cost-Prior equivalence

We want to show that the optimisation with respect to p_Ψ of the loss

$$\mathcal{L} = \langle C(\mathbf{s}) \rangle_{p_\Psi(\mathbf{s}|\mathbf{x})} + D_{\mathrm{KL}} \left[p_\Psi(\mathbf{s} \mid \mathbf{x}) \parallel p_\Theta(\hat{\mathbf{c}}(\mathbf{s}) \mid \mathbf{x}) \right] \tag{142}$$

is equivalent to the optimisation of the KL divergence between p_Ψ and a new generative model p'_Θ

$$D_{\mathrm{KL}} \left[p_\Psi(\mathbf{s} \mid \mathbf{x}) \parallel p'_\Theta(\mathbf{c}(\mathbf{s}) \mid \mathbf{x}) \right], \tag{143}$$

with p'_Θ defined as

$$p'_\Theta(\mathbf{x}, \hat{\mathbf{c}}(\mathbf{s})) \propto p_\Theta(\mathbf{x} \mid \hat{\mathbf{c}}(\mathbf{s})) \underbrace{p_\Theta(\hat{\mathbf{c}}(\mathbf{s})) \exp(-C(\mathbf{s}))}_{\text{effective prior}}$$

and so the cost $C(\mathbf{s})$ can be understood as an additional prior on the generative model p_Θ. This result is inspired by the model of bounded rationality of Orthega and Braun [54].

Proof. The key is to write out the explicit form of the KL divergence

$$\begin{aligned}
&\langle C(\mathbf{s})\rangle_{p_\Psi(\mathbf{s}|\mathbf{x})} + D_{\mathrm{KL}}\left[p_\Psi(\mathbf{s}\mid\mathbf{x}) \parallel p_\Theta(\hat{\mathbf{c}}(\mathbf{s})\mid\mathbf{x})\right] \\
=&\langle -\log\left(\exp(-C(\mathbf{s}))\right)\rangle_{p_\Psi(\mathbf{s}|\mathbf{x})} + \left\langle \log\left(\frac{p_\Psi(\mathbf{s}\mid\mathbf{x})}{p_\Theta(\hat{\mathbf{c}}(\mathbf{s})\mid\mathbf{x})}\right)\right\rangle_{p_\Psi(\mathbf{s}|\mathbf{x})} \\
=&\left\langle \log\left(\frac{p_\Psi(\mathbf{s}\mid\mathbf{x})}{p_\Theta(\hat{\mathbf{c}}(\mathbf{s})\mid\mathbf{x})\exp(-C(\mathbf{s}))}\right)\right\rangle_{p_\Psi(\mathbf{s}|\mathbf{x})} \\
=&\left\langle \log\left(\frac{p_\Psi(\mathbf{s}\mid\mathbf{x})}{p'_\Theta(\hat{\mathbf{c}}(\mathbf{s})\mid\mathbf{x})}\right)\right\rangle_{p_\Psi(\mathbf{s}|\mathbf{x})} - \log\left(\sum_{\mathbf{s}'} p_\Theta(\hat{\mathbf{c}}(\mathbf{s}')\mid\mathbf{x})\exp(-C(\mathbf{s}'))\right) \\
=&D_{\mathrm{KL}}\left[p_\Psi(\mathbf{s}\mid\mathbf{x}) \parallel p'_\Theta(\hat{\mathbf{c}}(\mathbf{s})\mid\mathbf{x})\right] - \log Z'_\Theta(\mathbf{x}).
\end{aligned}$$

with

$$Z'_\Theta(\mathbf{x}) = \sum_{\mathbf{s}'} p_\Theta(\hat{\mathbf{c}}(\mathbf{s}')\mid\mathbf{x})\exp(-C(\mathbf{s}'))$$

As the term containing Z'_Θ does not depend on p_Ψ we have obtained our equivalence. □

A.2 Convergence in probability

Here we motivate the statement that for $\delta t \to 0$ the probability distribution

$$p'_\Psi(\mathbf{s}_t\mid\mathbf{x}_t) \propto (\delta t)^{\sum_i s^i}\exp\Big(-\sum_i s^i(L(\mathbf{e}^i,\mathbf{x}_t) - L(\mathbf{0},\mathbf{x}_t)) + \mathcal{O}(\mathbf{s}^2)\Big) \tag{144}$$

will converge in distribution to

$$p_\Psi(\mathbf{s}_t\mid\mathbf{x}_t) = \prod_i p_\Psi(s^i_t\mid\mathbf{x}_t) \tag{145}$$

with

$$p_\Psi(s^i_t = 1\mid\mathbf{x}_t) = \delta t\ \exp\Big(-(L(\mathbf{e}^i,\mathbf{x}_t) - L(\mathbf{0},\mathbf{x}_t))\Big) \tag{146}$$

As we work with discrete distributions, we simply have to check that the probability for a specific event $\mathbf{s}_t$ will converge.

We do this by noting that for decreasing δt, the probability of simultaneous spiking will go to zero with order $\mathcal{O}(\delta t^2)$ for both distributions p_Ψ and p'_Ψ. Next we note that for a vector $\mathbf{s}_t$ with a single non-zero entry, the spiking probabilities are identical up to a constant determined by normalization. Then finally, as the non-zero entries will not contribute to normalization, the probability of no spike will converge to the same value as demanded by normalization.

B Simulation details

The code for this project will be made available on GitHub. We list here the simulation parameters used to generate the figures. In all simulations we used a discretization timescale of $\delta t = 1$ms and a kernel timescale of $\tau = 150 = 1$ms.

Parameter	Value	Description
n	10	number of neurons
ν	0.002	cost on spiking
μ	0, 0.5	cost on repeated spiking
ρ	$1 \cdot 10^{-6}$	decoder

Table 1: Simulation parameters of figure 2.4

Parameter	Value	Description
ν	-3	cost on spiking
D	2.5	decoder scalar
β_x	1	observation precision
β_c	0.2	prior precision
c_p	5	prior mean

Table 2: Simulation parameters of figure 3.3

Parameter	Value	Description
ν	0.	cost on spiking
d	3	decoder scalar
β_x	1	observation precision
β_c	0.2	prior precision
c_0	5	prior mean

Table 3: Simulation parameters of figure 3.4

The initial values for $\mathbf{D}_x \sim \mathcal{N}(10^{-3}, 10^{-4})$ were chosen to be generally positive as to stimulute activity in the initial phase of learning. The values for $\mathbf{P}_x$ were chosen to be normalized Gabor patches.

Parameter	Value	Description
ν	$-3.$	cost on spiking
d	3	decoder scalar
β_x	1	observation precision
β_c	0.2	prior precision
c_0	5	prior mean

Table 4: Simulation parameters of figure 3.5

Parameter	Value	Description
n	25	number of neurons
τ	100 ms	kernel timescale
ν	-2	cost on spiking
d	1	decoder scalar
β_x	10	observation precision
β_c	0.3	prior precision
c_0	5	prior mean
γ	$1 \cdot 10^{-6}$	learning rate

Table 5: Simulation parameters of figure 3.7

Parameter	Value	Description
n	20	number of neurons
τ_p	10 ms	low-pass timescale
ν	0	cost on spiking
d	1	decoder scalar
β_x	1	observation precision
β_c	1	prior precision
γ	$1 \cdot 10^{-4}$	learning rate

Table 6: Simulation parameters of figures 3.9 and 3.10