Theoretical Analysis of Stochastic Computations and Learning in Networks of Spiking Neurons

by

Stefan Habenschuss

DISSERTATION

submitted for the degree of

Doctor Technicae

Institute for Theoretical Computer Science
Graz University of Technology

Thesis Advisor:
O.Univ.-Prof. Dipl.-Ing. Dr.rer.nat. Wolfgang Maass

defended on January 15th, 2014

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, December 2013 .....

(signature)

Abstract

Human cognition is believed to emerge in complex networks of neurons in the brain. Recent findings in cognitive science have demonstrated that many aspects of cognition and behavior are inherently variable, and can be well explained by stochastic models. Moreover, it has been suggested that such stochastic behavior may be understood on a rational basis as the footprint of efficient stochastic algorithms that are carried out in the brain, such as sample-based inference on high-dimensional and often incomplete and ambiguous inputs, stochastic memory search, and sample-based autonomous learning of a statistically optimized internal model of the environment. The neural basis of these processes, however, remains largely unknown. This thesis investigates how stochastic algorithms such as Markov Chain Monte Carlo, sample-based inference, stochastic search, and autonomous model optimization, could emerge in networks of spiking neurons on the implementation level.

First, a general theoretical framework for understanding stochastic computations in biological networks of spiking neurons is developed. It is proved that under mild conditions every stochastic network of spiking neurons has, for any given input, a unique stationary distribution of network states. On the basis of this result, a framework is proposed for storing and representing complex knowledge in the form of probability distributions over network states. The retrieval of information in this framework occurs in a sample-based manner such that the state of the network at any moment represents a sample from a stored probability distribution. This establishes a link between realistic neural network dynamics and Markov Chain Monte Carlo sampling, and a theoretical foundation for the hypothesis that the brain stores, represents and processes knowledge in the form of probability distributions.

Second, it is shown how specific knowledge can be programmed in this framework into a network of spiking neurons through the use of simple circuit motifs. In particular, it is demonstrated how hard constraint satisfaction and optimization problems can be encoded in networks of spiking networks, such that the intrinsic network dynamics automatically generates valid solutions through stochastic search.

Third, it is considered how neuronal plasticity could enable networks of stochastic spiking neurons to acquire an optimized internal model of their high-dimensional spike input streams. In networks of neurons organized in winner-take-all (WTA) circuits, a salient circuit motif in the brain, the emergence of sample-based inference and statistically optimal learning is linked to a biologically well-known plasticity mechanism: spike-timing-dependent plasticity (STDP). Furthermore, using machine learning methods it is shown that the interplay of STDP and another experimentally reported form of plasticity, intrinsic homeostatic plasticity, allows WTA circuits to process and learn from arbitrary input spike streams in a near-optimal Bayesian manner.

Zusammenfassung

Menschliche Kognition entsteht, so wird vermutet, in komplexen neuronalen Netzwerken im Gehirn. Neueste Erkenntnisse in den Kognitionswissenschaften legen nahe, dass viele Aspekte von menschlichem Verhalten von Natur aus variabel sind, und mittels stochastischer Modelle erklärt werden können. Außerdem wurde vorgeschlagen, dass manch stochastisches Verhalten auf rationaler Grundlage als "Fußabdruck" effizienter stochastischer Algorithmen im Gehirn verstanden werden kann, wie beispielsweise sample-basierte Inferenz auf hochdimensionalen und oft unvollständigen und mehrdeutigen Eingabesignalen, stochastische Suche, und autonomes Lernen eines statistisch optimierten internen Modells der Umgebung. Die neuronale Grundlage dieser Prozesse ist jedoch noch weitgehend unbekannt. Diese Arbeit untersucht, wie stochastische Algorithmen wie Markov Chain Monte Carlo, sample-basierte Inferenz, stochastische Suche und sample-basierte autonome Modelloptimierung, in neuronalen Netzwerken auf der Implementierungsebene entstehen können.

Zuerst wird ein allgemeiner theoretischer Rahmen für das Verständnis stochastischer Berechnungen in biologischen neuronalen Netzwerken entwickelt. Es wird bewiesen, dass unter milden Bedingungen jedes stochastische neuronale Netz für ein gegebenes Eingabesignal eine eindeutige stationäre Verteilung von Netzwerkzuständen besitzt. Auf der Grundlage dieses Ergebnisses wird ein Rahmen für die Speicherung und Darstellung komplexen Wissens in Form von Wahrscheinlichkeitsverteilungen über Netzwerkzustände vorgeschlagen. Der Abruf von Informationen in diesem Rahmen erfolgt in einer sample-basierten Art, sodass der Zustand des Netzwerkes zu jedem Zeitpunkt eine Probe (ein sample) von einer gespeicherten Wahrscheinlichkeitsverteilung darstellt. Dies stellt eine Verbindung zwischen realistischer neuronaler Netzwerkdynamik und Markov Chain Monte Carlo sampling her, und liefert eine theoretische Grundlage für die Hypothese, dass das Gehirn Wissen in Form von Wahrscheinlichkeitsverteilungen speichert, repräsentiert und verarbeitet.

Zweitens wird gezeigt, wie spezifisches Wissen in diesem Rahmen in einem Netzwerk von Spiking Neuronen durch die Verwendung von einfachen Schaltungsmotiven programmiert werden kann. Insbesondere wird gezeigt, wie schwere Constraint Satisfaction und Optimierungsprobleme in Netzwerken von Spiking Neuronen kodiert werden können, sodass die intrinsische Netzwerkdynamik automatisch gültige Lösungen durch stochastische Suche erzeugt.

Drittens wird untersucht, wie neuronale Plastizität es Netzwerken von stochastischen Neuronen ermöglichen könnte, ein optimiertes internes Modell von hochdimensionalen Spiking Eingangssignalen zu bilden. In Netzwerken von Neuronen, welche in winner-take-all (WTA)-Schaltungen organisiert sind, ein häufiges Schaltungsmotiv im Gehirn, wird eine Beziehung zwischen sample-basierter Inferenz und statistisch optimalem Lernen auf der einen Seite, und einem biologisch bekannten Plastizitätsmechanismus, spike-timing-dependent plasticity (STDP), auf der ande-

ren Seite, geknüpft. Darüber hinaus wird mit Methoden des maschinellen Lernens gezeigt, dass das Zusammenspiel von STDP und einer weiteren Form von experimentell beobachteter Plastizität, homöostatischer intrinsischer Plastizität, es WTA-Schaltungen ermöglicht, auf nahezu optimale Weise beliebige Spiking Eingangssignale zu verarbeiten und von diesen zu lernen.

Acknowledgments

I owe my deepest gratitude to Wolfgang Maass, my supervisor, for introducing me to the world of biological neural networks, and for the extraordinary support and guidance he has given me throughout my PhD studies. I would also like to thank Peter Auer for his friendly willingness to be the second referee of this thesis.

It was an honor and a great pleasure to work with my co-authors and colleagues, Johannes Bill, Zeno Jonke, Bernhard Nessler, and Helmut Puhr, who have contributed substantially to our joint publications, and who have helped and supported me on countless occasions during the last years. Without them, this thesis would not have been possible. I would also like to express my gratitude to Dejan Pecevski for kindly providing the event-based spiking neural network simulator NEVESIM that was used for some of the simulations in this thesis.

I am indebted to the many current and former colleagues at the institute who made the office a fun and stimulating environment: Robert Legenstein, Stefan Häusler, Michael Pfeiffer, Gerhard Neumann, Bernhard Nessler, Stefan Klampfl, Dejan Pecevski, Gregor Hörzer, Lars Büsing, Jing Fang, Johannes Bill, Elmar Rückert, David Kappel, Zeno Jonke, Martin Ratajczak, Octave Etard, Christoph Pokorny, Gernot Griesbacher and Florian Hubner. I am deeply grateful to Daniela Potzinger, Regina Heidinger, and Oliver Friedl for their friendly support, and for the warm and welcoming atmosphere they have created.

I would also like to acknowledge that this thesis would not have been possible without financial support from Graz University of Technology and various research programs of the European Union, including SECO, PASCAL2, Organic, Amarsi, Brain-I-Nets, BrainScaleS, and the Human Brain Project.

I feel immensely grateful to my family and friends for all the support and the precious time we were able to spend together. It is my greatest pleasure to thank my parents, Gabriela and Wolfgang, for their loving encouragement, advice and support in all those years.

Finally, I would like to thank Simone for her unending patience and love, and for standing by me through all ups and downs of this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The human brain is arguably the most remarkable achievement of evolution. To master the challenges of life, evolution has equipped the brain with billions of densely packed nerve cells, called *neurons*. These neurons communicate via short electrical pulses, called action potentials or *spikes*, that are transmitted along thin biological wires, called *axons*, and passed on to other neurons through trillions of *synaptic connections*. In this manner, neurons form massively complex and finely tuned networks that are believed to lend us our cognitive abilities, from perception, emotion, fine motor control, to conscious thought and complex problem solving.

Computational neuroscience – a relatively young and highly interdisciplinary research field – has set out to quantitatively understand how networks of neurons in the brain process information and compute with it: how they process and integrate sensory input from various sensory modalities (visual, auditory, tactile, etc.), how they store and retrieve memories and experience, how they generate motor plans and direct muscle movements, and, ultimately, how they give rise to higher cognitive functions. A natural way to contribute to these questions from the perspective of computer science is to identify footprints of *algorithms and representations* that underlie neural processing in the brain. A major challenge inherent in this approach is that computations appear to be organized quite differently in networks of neurons than in computers. For example, unlike synchronous signaling in modern digital computers, neurons compute and emit spikes in an inherently *asynchronous* manner (Gerstner and Kistler, 2002), without a global clock that centrally coordinates communication. Furthermore, information processing in the brain occurs in an *inherently parallel* manner, in contrast to the sequential (or weakly parallel) organization of information processing in computers.

Among the many peculiarities of biological neural networks, which make them rather unlikely computing devices from the perspective of digital computing, one of the most fundamental and remarkable features is the amazing abundance of stochasticity and noise involved in neural processing. As a recent review of the literature concluded, "Noise is an inescapable consequence of brains operating with molecular components at the nanometer scale, sensors that are sensitive to individual quanta and complex networks of noisy neurons that generate behavior" (Faisal et al., 2008). Indeed, numerous experimental data highlight inherently stochastic aspects of neurons, synapses and networks of neurons on virtually all spatial and temporal scales that have been examined (Allen and Stevens, 1994; Faisal et al., 2008; Borst, 2010; Yarom and Hounsgaard, 2011; Clarke, 2012). The fact that noise and random vari-

ability are present at microscopic scales is of little surprise due to the inherently stochastic quantum-mechanical nature of molecular interactions. However, in contrast to most macroscopic physical systems which are sufficiently well described by deterministic laws, in the brain variability appears to be a phenomenon that is maintained across scales: it is observed in single neurons and synapses, in response properties of neuronal populations, and in behavior.

Given that the evolution of the brain has taken place over millions of years, it is unlikely that such prominent phenomenon is an undesired feature of neural computation. In fact, is it possible that stochasticity plays some integral role in the information processing strategy employed by the nervous system? Consistent with such hypothesis, a closer look suggests that the amount of variability is sophistically controlled by the brain. On the molecular scale, for example, neurotransmitter release is a basically stochastic process. Curiously, the reliability of neurotransmitter release is considerably different from synapse to synapse (Branco and Staras, 2009): It is very reliable in some synapses and highly unreliable in others. Even in a single synapse the release probability appears to be adapted dynamically through complex feedback mechanisms. Also in neuronal response properties the level of variability is not a static quantity. A basic phenomenon that has been observed, for example, is that variability is significantly reduced at the onset of a new sensory stimulus (Churchland et al., 2010). Finally, also on the behavioral level variability appears to be sophistically regulated. In motor control experiments, for instance, motor variability in human subjects is observed to be significantly reduced in tasks that demand high precision movements (Selen et al., 2006). And more generally, task-relevant variables appear to be controlled by human subjects with considerably higher precision than task-irrelevant variables (Latash et al., 2001; Valero-Cuevas et al., 2009).

The above examples indicate that variability can, in principle, be up- and down-regulated by the brain. This raises an important question: In situations where variability is *not* down-regulated, what could its functional role in human behavior be?

One clue may come from how the brain interprets incomplete or ambiguous sensory input, and infers hidden causes of observations. From a computer science perspective, there exist basically two fundamentally different algorithmic approaches to perform inference on ambiguous data (in the context of a prior knowledge base): deterministic algorithms based on belief propagation or variational methods, and stochastic algorithms based on Markov Chain Monte Carlo (MCMC) sampling (Bishop, 2006). Algorithms from these two classes can be distinguished quite easily through their properties in response to ambiguous input: belief propagation or variational methods will provide a deterministic and stable result in the form of analog probabilities for each interpretation (e.g. 50%/50% in an ambiguous case with two equally likely interpretations). MCMC, on the other hand, will provide a sequence of *samples* which are, in the long run, distributed according to these

probabilities. In particular, each sample in MCMC votes for an interpretation of the data. Hence, in the example of two ambiguous interpretations, roughly an equal number of samples produced by MCMC will vote for either percept. Since samples are constructed in a sequential manner through local stochastic perturbations, switches between the interpretations occur in a stochastic manner. Curiously, the dynamics by which MCMC generates possible interpretations through stochastic perturbations appears to be very closely related to how humans respond to the presentation of ambiguous stimuli: the percept of an ambiguous image switches forth and back between the two coherent interpretations. This (and a number of related findings) has led cognitive scientists and neuroscientists to propose that one possible use of stochasticity in the brain could be to support sample-based representations of probability distributions and, in particular, sample-based inference (Gershman et al., 2009; Fiser et al., 2010).

A closely related proposal regarding the functional role of variability is that variable responses reflect an ongoing stochastic search process. For example, when asked to search their memory for as many words as they can in a given category (such as *animals*), humans produce a sequence of words in an apparently random fashion. The thinking time between two words, however, is significantly increased if the words are less related to another (i.e. the thinking time is longer between *cat* and *elephant* than between *cat* and *dog*). This has been taken as evidence for a stochastic search model of memory retrieval, according to which new words are generated internally by a random walk on structured semantic memory representations (Austerweil et al., 2012).

Finally, variability may also have important functional roles during human learning. This has been well-known in the context of human reinforcement learning (Daw et al., 2006). But recent studies have also begun to investigate potential roles of variability during unsupervised, autonomous learning. In the context of children's cognitive development, for example, it has been suggested that children's learning dynamics are indicative of a "stochastic search at two levels of abstraction – an outer loop in the space of theories, and an inner loop in the space of explanations or models generated by each theory given a particular dataset – in order to discover the theory that best explains the observed data." (Ullman et al., 2010). In the proposed model, stochastic, sample-based representations of sensory interpretations are therefore considered to play an integral role during the acquisition and maintenance of an internal model of the environment. Notably, this idea has also been tested recently in experimental neuroscience: indeed, a highly influential (and also controversial) study appears to have found first indirect evidence for the hypothesis that autonomous learning of internal models of the environment occurs through an ongoing refinement of stochastic, sample-based representations in neural circuits (Berkes et al., 2011).

It is intriguing that some of the most powerful cognitive abilities of humans, such as inference, memory search or maintenance of an internal model of the environment,

appear to rely fundamentally on stochasticity. At least on the level of behavioral data, the brain seems to exploit stochasticity in a very similar way as powerful algorithms from computer science do. While much progress has been made on the behavioral level, the neural basis of these processes remains largely unknown. Shedding light on how these processes could be implemented in networks of neurons in the brain therefore constitutes an exciting challenge for future research.

The goal of this thesis is to provide some of the many missing pieces of the puzzle, in particular regarding the question how behaviorally relevant computations such as sample-based inference and stochastic search, as well as autonomous model optimization and learning, could arise in networks of stochastic spiking neurons, in an inherently parallel and asynchronous manner. This thesis contributes to the theoretical understanding of stochastic computations and learning in networks of spiking neurons in three ways. First, a general theoretical framework for the emergence of stochastic knowledge representations in realistic biological networks with noise is developed. This provides, in particular, a rigorous theoretical foundation for the hypothesis that knowledge is stored in brain in the form of probability distributions which can be accessed in a sample-based manner. Second, it is shown how specific knowledge can be encoded within this framework in networks of spiking neurons through the use of simple circuit motifs. By this means, networks of spiking neurons acquire the capability to solve hard constraint satisfaction and optimization problems through stochastic search. Third, it is shown how stochastic neural circuits organized in winner-take-all (WTA) circuits, a common circuit motif in cortex, can learn to form efficient internal models of their high-dimensional input streams in a completely unsupervised manner through biologically plausible learning mechanisms.

1.1 Organization of the Thesis

The results of this thesis are presented in the form of four paper manuscripts which were written in collaboration with my supervisor, Wolfgang Maass, and my colleagues Johannes Bill, Zeno Jonke, Bernhard Nessler and Helmut Puhr. These manuscripts were based, to a major extent, on theoretical findings I made during my PhD studies. The manuscripts are organized such that the main findings are presented in a relatively concise format. Supporting information, as well as derivations and proofs, can be found in the Appendix. *Acknowledgments* of my collaborator's contributions to the manuscripts can be found at the end of each Chapter.

In *Chapter 2* a new stochastic computing paradigm for networks of spiking neurons is developed, that is based on the observation that neurons are inherently stochastic. It is shown that the dynamics of noisy biological spiking networks with virtually arbitrarily complex dynamics including short term plasticity, non-linear dendritic computations, and arbitrary shapes of post-synaptic potentials, can be

viewed as a sampling process from a stationary distribution of network states. In this manner, a link between realistic biological neural network dynamics and well-known Markov Chain Monte Carlo algorithms is established, that integrates previous, more specific findings of (Buesing et al., 2011; Pecevski et al., 2011) in a more general context. The theoretical basis for this link is a set of mathematical proofs which guarantee exponential ergodicity of the network dynamics under a wide range of input scenarios and otherwise very mild conditions. This provides the foundation of a generic framework for storing and representing complex knowledge in the form of probability distributions over network states. The retrieval of information occurs in a sampled-based manner: the state of the network at any moment represents a sample from a stored probability distribution. This allows various inference tasks on stored probability distributions, such as marginalization or marginal maximum a-posteriori estimation, to be carried out by simple readout operations. Altogether, this work provides a rigorous theoretical foundation for the hypothesis that the brain stores, represents and processes information in the form of probability distributions.

In *Chapter 3* it is investigated how this new stochastic computing paradigm can be put to use to solve difficult computational problems with spiking networks. It is shown how problem specific knowledge and constraints can be programmed into a network of spiking neurons using simple circuit motifs. Through its inherent stochastic dynamics, the network searches and identifies solutions which meet the problem constraints. The underlying theoretical principle of the approach is to construct a circuit such that the relative occurrence of a network state $\mathbf{x}$ is inversely related to the number of problem constraints violated by $\mathbf{x}$. Hence, network states $\mathbf{x}$ which correspond to correct solutions occur particularly often. Applications of this principle are demonstrated for satisfiability (3-SAT) and planning problems (Traveling Salesman Problem).

In *Chapter 4* it is addressed how stochastic microcircuits in the cortex can learn to optimally integrate information from many unreliable sources in a Bayesian manner. It is shown that neurons organized in winner-take-all circuits, a common circuit motif in the brain, can efficiently learn to extract and represent hidden causes of their high-dimensional inputs through a biologically plausible learning mechanism, spike-timing-dependent plasticity (STDP). In particular, STDP is shown to support the emergence of optimal decoding of input population codes in such stochastic winner-take-all networks. The basis for this result is a theoretical link between STDP and an online version of the powerful Expectation Maximization algorithm in a generative mixture model with Poisson distributed observations, which extends and generalizes previous work by (Nessler et al., 2010). Poisson distributions are particularly relevant in cortex, since they are often a good model for spike train variability. Variants of the theoretically derived STDP rule for Poisson distributed inputs can be shown to implement Bayes-optimal plasticity when observations are distributed according to other exponential family distributions, such as Gamma

or Gaussian distributions. Altogether, these results show how STDP, arguably one of the most important plasticity mechanisms in cortex, could support optimal (Bayesian) decoding of input streams in cortical microcircuits for a wide range of input distributions.

Chapter 5 builds on the learning approach of Chapter 4 and investigates how synaptic plasticity rules (such as STDP) could interact with homeostatic plasticity, another prominent plasticity mechanism in the brain, from the perspective of statistically optimal learning. Among other benefits, it is shown that the interplay of synaptic plasticity and intrinsic homeostatic plasticity allows WTA circuits to process and learn from *arbitrary* input spike streams in a near-optimal Bayesian manner, thus extending the scope of the theoretical learning approach of Chapter 4 to a much wider range of input statistics.

Chapter 2

Stochastic Computations in Cortical Microcircuit Models

Contents

Experimental data from neuroscience suggest that a substantial amount of knowledge is stored in the brain in the form of probability distributions over network states and trajectories of network states. We provide a theoretical foundation for this hypothesis by showing that even very detailed models for cortical microcircuits, with data-based diverse nonlinear neurons and synapses, have a stationary distribution of network states and trajectories of network states to which they converge exponentially fast from any initial state. We demonstrate that this convergence holds in spite of the non-reversibility of the stochastic dynamics of cortical microcircuits. We further show that, in the presence of background network oscillations, separate stationary distributions emerge for different phases of the oscillation, in accordance with experimentally reported phase-specific codes. We complement these theoretical results by computer simulations that investigate resulting computation times for typical probabilistic inference tasks on these internally stored distributions, such as marginalization or marginal maximum-a-posteriori estimation. Furthermore, we show that the inherent stochastic dynamics of generic cortical microcircuits enables them to quickly generate approximate solutions to difficult constraint satisfaction

problems, where stored knowledge and current inputs jointly constrain possible solutions. This provides a powerful new computing paradigm for networks of spiking neurons, that also throws new light on the way how networks of neurons in the brain could carry out complex computational tasks such as prediction, imagination, memory recall and problem solving.

2.1 Introduction

The question whether brain computations are inherently deterministic or inherently stochastic is obviously of fundamental importance. Numerous experimental data highlight inherently stochastic aspects of neurons, synapses and networks of neurons on virtually all spatial and temporal scales that have been examined (Allen and Stevens, 1994; Faisal et al., 2008; Borst, 2010; Yarom and Hounsgaard, 2011; Clarke, 2012). A clearly visible stochastic feature of brain activity is the trial-to-trial variability of neuronal responses, which also appears on virtually every spatial and temporal scale that has been examined (Faisal et al., 2008). This variability has often been interpreted as side-effect of an implementation of inherently deterministic computing paradigms with noisy elements, and it has been attempted to show that the observed noise can be eliminated through spatial or temporal averaging. However, more recent experimental methods, which make it possible to record simultaneously from many neurons (or from many voxels in fMRI), have shown that the underlying probability distributions of network states during spontaneous activity are highly structured and multimodal, with distinct modes that resemble those encountered during active processing. This has been shown through recordings with voltage-sensitive dyes starting with (Tsodyks et al., 1999; Kenet et al., 2003), multi-electrode arrays (Luczak et al., 2009), and fMRI (Raichle, 2010; Lewis et al., 2009). It was also shown that the intrinsic trial-to-trial variability of brain systems is intimately related to the observed trial-to-trial variability in behavior (see e.g. (Fox et al., 2007)). Furthermore, in (Kelemen and Fenton, 2010) it was shown that during navigation in a complex environment where simultaneously two spatial frames of reference were relevant, the firing of neurons in area *CA1* represented both frames in alternation, so that coactive neurons tended to relate to a common frame of reference. In addition it has been shown that in a situation where sensory stimuli are ambiguous, large brain networks switch stochastically between alternative interpretations or percepts, see (Leopold and Logothetis, 1996, 1999; Kim and Blake, 2005). Furthermore, an increase in the volatility of network states has been shown to accompany episodes of behavioral uncertainty (Karlsson et al., 2012). All these experimental data point to inherently stochastic aspects in the organization of brain computations, and more specifically to an important computational role of spontaneously varying network states of smaller and larger networks of neurons in the brain. However, one should realize that the approach to stochastic computation that we examine in this article does not postulate that

all brain activity is stochastic or unreliable, since reliable neural responses can be represented by probabilities close to 1.

The goal of this article is to provide a theoretical foundation for understanding stochastic computations in networks of neurons in the brain, in particular also for the generation of structured spontaneous activity. To this end, we prove here that even biologically realistic models C for networks of neurons in the brain have – for a suitable definition of network state – a unique stationary distribution p_C of network states. Previous work had focused in this context on neuronal models with linear sub-threshold dynamics (Brémaud and Massoulié, 1996; Borovkov et al., 2012) and constant external input (e.g. constant input firing rates). However, we show here that this holds even for quite realistic models that reflect, for example, data on nonlinear dendritic integration (dendritic spikes), synapses with data-based short term dynamics (i.e., individual mixtures of depression and facilitation), and different types of neurons on specific laminae. We also show that these results are not restricted to the case of constant external input, but rather can be extended to periodically changing input, and to input generated by arbitrary ergodic stochastic processes.

Our theoretical results imply that virtually any data-based model C, for networks of neurons featuring realistic neuronal noise sources (e.g. stochastic synaptic vesicle release) implements a Markov process through its stochastic dynamics. This can be interpreted – in spite of its non-reversibility – as a form of sampling from a unique stationary distribution p_C. One interpretation of p_C, which is in principle consistent with our findings, is that it represents the posterior distribution of a Bayesian inference operation (Hoyer and Hyvärinen, 2003; Berkes et al., 2011; Buesing et al., 2011; Pecevski et al., 2011), in which the current input (evidence) is combined with prior knowledge encoded in network parameters such as synaptic weights or intrinsic excitabilities of neurons (see (Friston, 2010; Vilares and Kording, 2011; Fiser et al., 2010; Doya et al., 2007) for an introduction to the "Bayesian brain"). This interpretation of neural dynamics as sampling from a posterior distribution is intriguing, as it implies that various results of probabilistic inference could then be easily obtained by a simple readout mechanism: For example, posterior marginal probabilities can be estimated (approximately) by observing the number of spikes of specific neurons within some time window (see related data from parietal cortex (Huk and Shadlen, 2005)). Furthermore, an approximate maximal a posteriori (MAP) inference can be carried out by observing which network states occur more often, and/or are more persistent.

A crucial issue which arises is whether reliable readouts from p_C in realistic cortical microcircuit models can be obtained quickly enough to support, e.g., fast decision making in downstream areas. This critically depends on the speed of convergence of the distribution of network states (or distribution of trajectories of network states) from typical initial network states to the stationary distribution. Since the initial network state of a cortical microcircuit C depends on past activity,

it may often be already quite "close" to the stationary distribution when a new input arrives (since past inputs are likely related to the new input). But it is also reasonable to assume that the initial state of the network is frequently unrelated to the stationary distribution p_C, for example after drastic input changes. In this case the time required for readouts depends on the expected convergence speed to p_C from – more or less – *arbitrary* initial states. We show that one can prove exponential upper bounds for this convergence speed. But even that does not guarantee fast convergence for a concrete system, because of constant factors in the theoretical upper bound. Therefore we complement this theoretical analysis of the convergence speed by extensive computer simulations for cortical microcircuit models.

The notion of a cortical microcircuit arose from the observation that "it seems likely that there is a basically uniform microcircuit pattern throughout the neocortex upon which certain specializations unique to this or that cortical area are superimposed" (Mountcastle, 1998). This notion is not precisely defined, but rather a term of convenience: It refers to network models that are sufficiently large to contain examples of the main types of experimentally observed neurons on specific laminae, and the main types of experimentally observed synaptic connections between different types of neurons on different laminae, ideally in statistically representative numbers (Douglas and Martin, 2004). Computer simulations of cortical microcircuit models are practically constrained both by a lack of sufficiently many consistent data from a single preparation and a single cortical area, and by the available computer time. In the computer simulations for this article we have focused on a relatively simple standard model for a cortical microcircuit in the somatosensory cortex (Haeusler and Maass, 2007) that has already been examined in some variations in previous studies from various perspectives (Haeusler et al., 2009; Rasch et al., 2011; Potjans and Diesmann, 2012; Bastos et al., 2012).

We show that for this standard model of a cortical microcircuit marginal probabilities for single random variables (neurons) can be estimated through sampling even for fairly large instances with 5000 neurons within a few 100 ms of simulated biological time, hence well within the range of experimentally observed computation times of biological organisms. The same holds for probabilities of network states for small sub-networks. Furthermore, we show that at least for sizes up to 5000 neurons these "computation times" are virtually independent of the size of the microcircuit model.

We also address the question to which extent our theoretical framework can be applied in the context of periodic input, for example in the presence of background theta oscillations (Dragoi and Buzsaki, 2006). In contrast to the stationary input case, we show that the presence of periodic input leads to the emergence of unique *phase-specific* stationary distributions, i.e., a separate unique stationary distribution for each phase of the periodic input. We discuss basic implications of this result and relate our findings to experimental data on theta-paced path sequences (Dragoi

and Buzsaki, 2006; Gupta et al., 2012) and bi-stable activity (Jezek et al., 2011) in hippocampus.

Finally, our theoretically founded framework for stochastic computations in networks of spiking neurons also throws new light on the question how complex constraint satisfaction problems could be solved by cortical microcircuits (Hinton et al., 1984; Davenport et al., 1994). We demonstrate this in a toy example for the popular puzzle game Sudoku. We show that the constraints of this problem can be easily encoded by synaptic connections between excitatory and inhibitory neurons in such a way that the stationary distribution p_C assigns particularly high probability to those network states which encode correct (or good approximate) solutions to the problem. The resulting network dynamics can also be understood as parallel stochastic search with anytime computing properties: Early network states provide very fast heuristic solutions, while later network states are distributed according to the stationary distribution p_C, therefore visiting with highest probability those solutions which violate only a few or zero constraints.

In order to make the results of this article accessible to non-theoreticians we present in the subsequent Results section our main findings in a less technical formulation that emphasizes relationships to experimental data. Rigorous mathematical definitions and proofs can be found in Appendix B, which has been structured in the same way as the following main Results sections in order to facilitate simultaneous access on different levels of detail.

2.2 Network states and distributions of network states

A simple notion of network state at time t simply indicates which neurons in the network fired within some short time window before t. For example, in (Berkes et al., 2011) a window size of 2 ms was selected. However, the full network state could not be analyzed there experimentally, only its projection onto 16 electrodes in area V1 from which recordings were made. An important methodological innovation of (Berkes et al., 2011) was to analyze under various conditions the probability distribution of the recorded fragments of network states, i.e., of the resulting bit vectors of length 16 (with a "1" at position i if a spike was recorded during the preceding 2 ms at electrode i). In particular, it was shown that during development the distribution over these 2^{16} network states during spontaneous activity in darkness approximates the distribution recorded during natural vision. Apart from its functional interpretation, this result also raises the even more fundamental question how a network of neurons in the brain can represent and generate a complex distribution of network states. This question is addressed here in the context of data-based models C for cortical microcircuits. We consider notions of network states y similar to (Berkes et al., 2011) (see the simple state $y_S(t)$ in Figure 2.1C) and provide a rigorous proof that under some mild assumptions any such model C represents and generates for different external inputs x associated different internal

distributions $p_C(y|x)$ of network states y. More precisely, we will show that for any specific input x there exists a unique stationary distribution $p_C(y|x)$ of network states y to which the network converges exponentially fast from any initial state.

This result can be derived within the theory of Markov processes on general state spaces, an extension of the more familiar theory of Markov chains on finite state spaces to continuous time and infinitely many network states. Another important difference to typical Markov chains (e.g. the dynamics of Gibbs sampling in Boltzmann machines) is that the Markov processes describing the stochastic dynamics of cortical microcircuit models are non-reversible. This is a well-known difference between simple neural network models and networks of spiking neurons in the brain, where a spike of a neuron causes postsynaptic potentials in other neurons - but not vice versa. In addition, experimental results show that brain networks tend to have a non-reversible dynamics also on longer time scales (e.g., stereotypical trajectories of network states (Abeles et al., 1995; Luczak et al., 2007; Buzsáki, 2010; Luczak and MacLean, 2012)).

In order to prove results on the existence of stationary distributions $p_C(y|x)$ of network states y, one first needs to consider a more complex notion of network state $y_M(t)$ at time t, which records the history of all spikes in the network C since time $t-\Theta$ (see Figure 1C). The window length Θ has to be chosen sufficiently large so that the influence of spikes before time $t-\Theta$ on the dynamics of the network after time t can be neglected. This more complex notion of network state then fulfills the *Markov property*, such that the future network evolution depends on the past only through the current Markov state. The existence of a window length Θ with the Markov property is a basic assumption of the subsequent theoretical results. For standard models of networks of spiking neurons a value of Θ around 100 ms provides already a good approximation of the Markov property, since this is a typical time during which a post-synaptic potential has a non-negligible effect at the soma of a post-synaptic neuron. For more complex models of networks of spiking neurons a larger value of Θ in the range of seconds is more adequate, in order to accommodate for dendritic spikes or the activation of $GABA_B$ receptors that may last 100 ms or longer, and the short term dynamics of synapses with time constants of several hundred milliseconds. Fortunately, once the existence of a stationary distribution is proved for such more complex notion of network state, it also holds for any simpler notion of network state (even if these simpler network states do not fulfill the Markov property), that results when one ignores details of the more complex network states. For example, one can ignore all spikes before time $t-2$ ms, the exact firing times within the window from $t-2$ ms to t, and whether a neuron fired one or several spikes. In this way one arrives back at the simple notion of network state from (Berkes et al., 2011).

Theorem 1 (Exponentially fast convergence to a stationary distribution) *Let C be an arbitrary model for a network of spiking neurons with stochastic synaptic release or some other mechanism for stochastic firing. C may consist of complex*

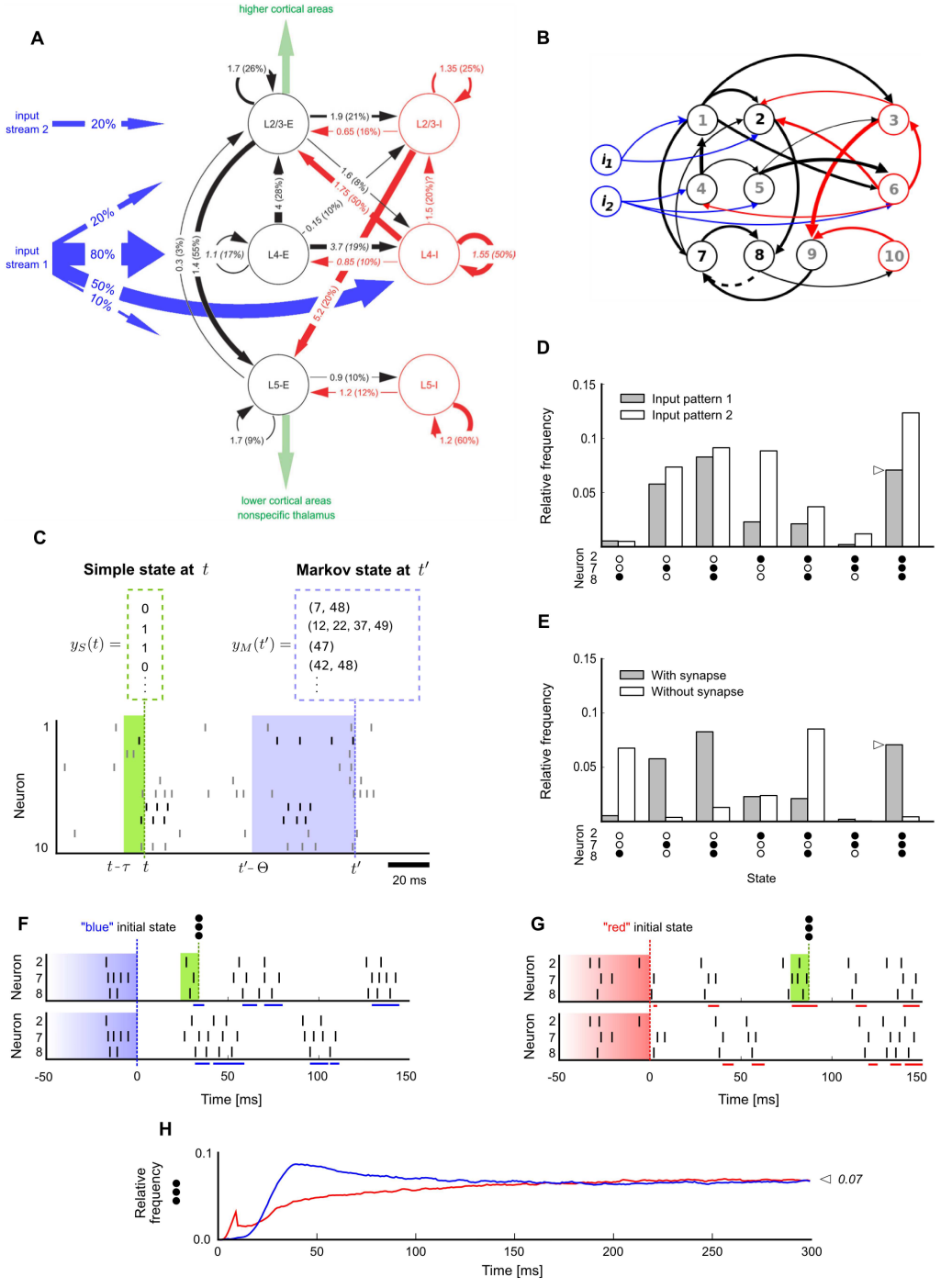

Figure 2.1: (see next page for Figure caption)

Figure 2.1: Network states and stationary distributions of network states in a cortical microcircuit model. **A**. Data-based cortical microcircuit template from (Haeusler and Maass, 2007); © 2007 by Oxford University Press, reprinted by permission of the authors and Oxford University Press. **B**. A small instantiation of this model consisting of 10 network neurons $1, \ldots, 10$ and 2 additional input neurons i_1, i_2. Neurons are colored by type (blue:input, black:excitatory, red:inhibitory). Line width represents synaptic efficacy. The synapse from neuron 8 to 7 is removed for the simulation described in E. **C**. Notions of network state considered in this article. Markov states are defined by the exact timing of all recent spikes within some time window Θ, shown here for $\Theta = 50$ ms. Simple states only record which neurons fired recently (0=no spike, 1=at least one spike within a short window τ, with $\tau = 10$ ms throughout this figure). **D**. Empirically measured stationary distribution of simple network states. Shown is the marginal distribution $p_C(\tilde{y}|x)$ for a subset of three neurons 2,7,8 (their spikes are shown in C in black), under two different input conditions (input pattern 1: i_1 firing at 10 Hz and i_2 at 50 Hz, input pattern 2: i_1 at 50 Hz and i_2 at 10 Hz). The distribution for each input condition was obtained by measuring the relative time spent in each of the simple states (0,0,0), ..., (1,1,1) in a single long trial (100 s). The zero state (0,0,0) is not shown. **E**. Effect of removing one synapse, from neuron 8 to neuron 7, on the stationary distribution of network states (input pattern 1 was presented). **F**. Illustration of trial-to-trial variability in the small cortical microcircuit (input pattern 1). Two trials starting from identical initial network states $y_M(0)$ are shown. Blue bars at the bottom of each trial mark periods where the subnetwork of neurons 2,7,8 was in simple state (1,1,1) at this time t. Note that the "blue" initial Markov state is shown only partially: it is actually longer and comprises all neurons in the network (as in panel C, but with $\Theta = 1s$). **G**. Two trials starting from a different ("red") initial network state. Red bars denote periods of state (1,1,1) for "red" trials. **H**. Convergence to the stationary distribution p_C in this small cortical microcircuit is fast and independent of the initial state: This is illustrated for the relative frequency of simple state (1,1,1) within the first 300 ms after input onset. The blue/red line shows the relative frequency of simple state (1,1,1) at each time t estimated from many (10^5) "blue"/"red" trials. The relative frequency of simple state (1,1,1) rapidly converges to its stationary value denoted by the symbol $\lhd$ (marked also in panels D and E). The relative frequency converges to the same value regardless of the initial state (blue/red).

multi-compartment neuron models with nonlinear dendritic integration (including dendritic spikes) and heterogeneous synapses with differential short term dynamics. We assume that this network C receives external inputs from a set of input neurons $i = 1 \ldots N$ which fire according to Poisson processes at different rates $x_i(t)$. The vector $x(t)$ of input rates can be either constant over time ($x(t) \equiv x$), or generated by any external Markov process that converges exponentially fast to a stationary distribution.

Then there exists a stationary distribution $p_C(y|x)$ of network states y, to which the stochastic dynamics of C converges from any initial state of the network exponentially fast. Accordingly, the distribution of subnetwork states $\tilde{y}$ of any subset of neurons converges exponentially fast to the marginal distribution $p_C(\tilde{y}|x)$ of this subnetwork.

Note that Theorem 1 states that the network embodies not only the joint distribution $p_C(y|x)$ over all neurons, but simultaneously all marginal distributions $p_C(\tilde{y}|x)$ over all possible subsets of neurons. This property follows naturally from the fact that $p_C(y|x)$ is represented in a sample-based manner (Fiser et al., 2010). As a consequence, if one is interested in estimating the marginal distribution of some subset of neurons rather than the full joint distribution, it suffices to observe the activity of the particular subnetwork of interest (while ignoring the remaining network). This is remarkable insofar, as the exact computation of marginal probabilities is in general known to be quite difficult (even NP-complete (Koller and Friedman, 2009)).

Theorem 1 requires that neurons fire stochastically. More precisely, a basic assumption required for Theorem 1 is that the network behaves sufficiently stochastic at *any point in time*, in the sense that the probability that a neuron fires in an interval $[t, t + \delta t)$ must be smaller than 1 for any t. This is indeed fulfilled by any stochastic neuron model as long as instantaneous firing rates remain bounded. It is also fulfilled by any deterministic neuron model if synaptic transmission is modeled via stochastic vesicle release with bounded release rates. Another assumption is that long-term plasticity and other long-term memory effects have a negligible impact on the network dynamics on shorter timescales which are the focus of this article (milliseconds to a few seconds). Precise mathematical definitions of all assumptions and notions involved in Theorem 1 as well as proofs can be found in Appendix B (see Lemma 2 and 3).

An illustration for Theorem 1 is given in Figure 2.1. We use as our running example for a cortical microcircuit model C the model of (Haeusler and Maass, 2007) shown in Figure 2.1A, which consists of three populations of excitatory and three populations of inhibitory neurons on specific laminae. Average strength of synaptic connections (measured as mean amplitude of postsynaptic potentials at the soma in mV, and indicated by the numbers at the arrows in Figure 2.1A) as well as the connection probability (indicated in parentheses at each arrow as % in Figure 2.1A) are based in this model on intracellular recordings from 998 pairs of identified neurons from the Thomson Lab (Thomson et al., 2002). The thickness of arrows in Figure 2.1A reflects the products of those two numbers for each connection. The nonlinear short-term dynamics of each type of synaptic connection was modeled according to data from the Markram Lab (Gupta et al., 2000; Markram et al., 1998). Neuronal integration and spike generation was modeled by a conductance-based leaky-integrate-and-fire model, with a stochastic spiking mechanism based on (Jolivet et al., 2006). See Appendix B for details.

The external input x consists in a cortical microcircuit of inputs from higher cortical areas that primarily target neurons in superficial layers, and bottom-up inputs that arrive primarily in layer 4, but also on other layers (details tend to depend on the cortical area and the species). We model two input streams in a qualitative manner as in (Haeusler and Maass, 2007). Also background synaptic

input is modeled according to (Haeusler and Maass, 2007).

Figure 2.1B shows a small instantiation of this microcircuit template consisting of 10 neurons (we had to manually tune a few connections in this circuit to facilitate visual clarity of subsequent panels). The impact of different external inputs x and of a single synaptic connection from neuron 8 to neuron 7 on the stationary distribution is shown in Figure 2.1D and E, respectively (shown is the marginal distribution $p_C(\tilde{y}|x)$ of a subset of three neurons 2,7 and 8). This illustrates that the structure and dynamics of a circuit C are intimately linked to properties of its stationary distribution $p_C(y|x)$. In fact, we argue that the stationary distribution $p_C(y|x)$ (more precisely: the stationary distribution $p_C(y|x)$ for all relevant external inputs x) can be viewed as a mathematical model for the most salient aspects of stochastic computations in a circuit C.

The influence of the initial network state on the first 150 ms of network response is shown in Figure 2.1F and G for representative trials starting from two different initial Markov states (blue/red, two trials shown for each). Variability among trials arises from the inherent stochasticity of neurons and the presence of background synaptic input. Figure 2.1H is a concrete illustration of Theorem 1: it shows that the relative frequency of a specific network state (1,1,1) in a subset of the three neurons 2,7 and 8 converges quickly to its stationary value. Furthermore, it converges to this (same) value regardless of the initial network state (blue/red).

2.3 Stationary distributions of trajectories of network states

Theorem 1 also applies to networks which generate stereotypical trajectories of network activity (Luczak et al., 2007). For such networks it may be of interest to consider not only the distribution of network states in a short window (e.g. simple states with $\tau = 10$ ms, or $\Theta = 50$ ms), but also the distribution of longer trajectories produced by the network. Indeed, since Theorem 1 holds for Markov states y_M with any fixed window length Θ, it also holds for values of Θ that are in the range of experimentally observed trajectories of network states (Mazor and Laurent, 2005; Luczak et al., 2007; Harvey et al., 2012). Hence, a generic neural circuit C automatically has a unique stationary distribution over *trajectories* of (simple) network states for any fixed trajectory length Θ. Note that this implies that a neural circuit C has simultaneously stationary distributions of trajectories of (simple) network states of various lengths for arbitrarily large Θ, and a stationary distribution of simple network states. This fact is not surprising if one takes into consideration that if a circuit C has a stationary distribution over simple network states this does *not* imply that subsequent simple network states represent independent drawings from this stationary distribution. Hence the circuit C may very well produce stereotypical trajectories of simple network states. This feature becomes even more prominent if the underlying dynamics (the Markov process) of the neural circuit is

non-reversible on several time scales.

2.4 Extracting knowledge from internally stored distributions

We address two basic types of knowledge extraction from a stationary distribution p_C of a network C: the computation of *marginal probabilities* and *maximal a posteriori (MAP) assignments*. Both computations constitute basic inference problems commonly appearing in real-world applications (Wainwright and Jordan, 2008), which are in general difficult to solve as they involve large sums, integrals, or maximization steps over a state space which grows exponentially in the number of random variables. However, already (Fiser et al., 2010; Buesing et al., 2011) noted that the *estimation* of marginal probabilities would become straightforward if distributions were represented in the brain in a sample-based manner (such that each network state at time t represents one sample from the distribution). Theorem 1 provides a theoretical foundation for how such a representation could emerge in realistic data-based microcircuit models on the implementation level: Once the network C has converged to its stationary distribution, the network state at any time t represents a sample from $p_C(y|x)$ (although subsequent samples are generally not independent). Simultaneously, the subnetwork state $\tilde{y}(t)$ of any subset of neurons represents a sample from the marginal distribution $p_C(\tilde{y}|x)$. This is particularly relevant if one interprets $p_C(y|x)$ in a given cortical microcircuit C as the posterior distribution of an implicit generative model, as suggested for example by (Berkes et al., 2011) or (Buesing et al., 2011; Pecevski et al., 2011).

In order to place the estimation of marginals into a biologically relevant context, assume that a particular component y_1 of the network state $y = (y_1, \ldots, y_K)$ has a behavioral relevance. This variable y_1, represented by some neuron n_1, could represent for example the perception of a particular visual object (if neuron n_1 is located in inferior temporal cortex (Zhang et al., 2011)), or the intention to make a saccade into a specific part of the visual field (if neuron n_1 is located in area LIP (Shadlen and Newsome, 2001)). Then the computation of the marginal

$$p_C(y_1 = 1|x) = \sum_{v_2 \in \{0,1\}, \ldots, v_K \in \{0,1\}} p_C(y_1 = 1, y_2 = v_2, \ldots, y_K = v_K|x) \qquad (2.1)$$

would be of behavioral significance. Note that this computation integrates information from the internally stored knowledge p_C with evidence about a current situation x. In general this computation is demanding as it involves a sum with exponentially many terms in the network size K.

But according to Theorem 1, the correct marginal distribution $p_C(y_1|x)$ is automatically embodied by the activity of neuron n_1. Hence the marginal probability $y_1 = 1$ can be estimated by simply observing what fraction of time the neuron

spends in the state $y_1 = 1$, while ignoring the activity of the remaining network (Buesing et al., 2011). In principle, a downstream neuron could gather this information by integrating the spike output of n_1 over time.

Marginal probabilities of subpopulations, for example $p_C(y_1 = 1, y_2 = 0, y_3 = 1|x)$, can be estimated in a similar manner by keeping track of how much time the subnetwork spends in the state (1,0,1), while ignoring the activity of the remaining neurons. A downstream network could gather this information, for example, by integrating over the output of a readout neuron which is tuned to detect the desired target pattern (1,0,1).

Notably, the estimation of marginals sketched above is guaranteed by ergodic theory to converge to the correct probability as observation time increases (due to Theorem 1 which ensures that the network is an ergodic Markov process, see Appendix B). In particular, this holds true even for networks with prominent sequential dynamics featuring, for example, stereotypical trajectories. However, note that the observation time required to obtain an accurate estimate may be longer when trajectories are present since subsequent samples gathered from such a network will likely exhibit stronger dependencies than in networks lacking sequential activity patterns. In a practical readout implementation where recent events might be weighed preferentially this could result in more noisy estimates.

Approximate maximal a posteriori (MAP) assignments to small subsets of variables $y_1, \ldots, y_m$ can also be obtained in a quite straightforward manner. For given external inputs x, the marginal MAP assignment to the subset of variables $y_1, \ldots, y_m$ (with some $m \leq K$) is defined as the set of values $\hat{v}_1, \ldots, \hat{v}_m$ that maximize

$$\sum_{v_{m+1} \in \{0,1\}, \ldots, v_K \in \{0,1\}} p_C(y_1 = \hat{v}_1, \ldots, y_m = \hat{v}_m, \;\; y_{m+1} = v_{m+1}, \ldots, y_K = v_K|x) \;\; . \tag{2.2}$$

A sample-based approximation of this operation can be implemented by keeping track of which network states in the subnetwork $n_1, \ldots, n_m$ occur most often. This could, for example, be realized by a readout network in a two stage process: first the marginal probabilities $p_C(y_1 = \hat{v}_1, y_2 = \hat{v}_2, y_3 = \hat{v}_3|x)$ of all $2^3 = 8$ subnetwork states $(0,0,0), \ldots, (1,1,1)$ are estimated (by 8 readout neurons dedicated to that purpose), followed by the selection of the neuron with maximal probability. The selection of the maximum could be achieved in a neural network, for example, through competitive inhibition. Such competitive inhibition would ideally lead to a winner-take-all function such that the neuron with the strongest stimulation (representing the variable assignment with the largest probability) dominates and suppresses all other readout neurons.

2.5 Estimates of the required computation time

Whereas many types of computations (for example probabilistic inference via the junction tree algorithm (Wainwright and Jordan, 2008)) require a certain computation time, probabilistic inference via sampling from an embodied distribution p_C belongs to the class of *anytime computing* methods, where rough estimates of the result of a computation become almost immediately available, and are automatically improved when there is more time for a decision. A main component of the convergence time to a reliable result arises from the time which the distribution of network states needs to become independent of its initial state y_0. It is well known that both, network states of neurons in the cortex (Arieli et al., 1996) and quick decisions of an organism, are influenced for a short time by this initial state y_0 (and this temporary dependence on the initial state y_0 may in fact have some behavioral advantage, since y_0 may contain information about preceding network inputs, expectations, etc.). But it has remained unknown, what range of convergence speeds for inference from p_C is produced by common models for cortical microcircuits C.

We address this question by analyzing the convergence speed of stochastic computations in the cortical microcircuit model of (Haeusler and Maass, 2007). A typical network response of an instance of the cortical microcircuit model comprising 560 neurons as in (Haeusler and Maass, 2007) is shown in Figure 2.2A. We first checked how fast marginal probabilities for single neurons converge to stationary values from different initial network Markov states. We applied the same analysis as in Figure 2.1H to the simple state ($\tau = 10$ ms) of a single representative neuron from layer 5. Figure 2.2B shows quite fast convergence of the "on"-state probability of the neuron to its stationary value from two different initial states. Note that this straightforward method of checking convergence is rather inefficient, as it requires the repetition of a large number of trials for each initial state. In addition it is not suitable for analyzing convergence to marginals for subpopulations of neurons (see Figure 2.2G).

Various more efficient *convergence diagnostics* have been proposed in the context of discrete-time Markov Chain Monte Carlo theory (Gelman and Rubin, 1992; Cowles and Carlin, 1996; Brooks et al., 2010; Gjoka et al., 2010). In the following, we have adopted the Gelman and Rubin diagnostic, one of the standard methods in applications of MCMC sampling (Gelman and Rubin, 1992). The Gelman Rubin convergence diagnostic is based on the comparison of many runs of a Markov chain when started from different randomly drawn initial states. In particular, one compares the typical variance of state distributions during the time interval $[t, 2t]$ within a single run (within-variance) to the variance during the interval $[t, 2t]$ between different runs (between-variance). When the ratio $\hat{R}$ of between- and within-variance approaches 1 this is indicative of convergence. A comparison of panels B and C of Figure 2.2 shows that in the case of marginals for single neurons this interpretation fits very well to the empirically observed convergence speed for two different initial conditions. Various values between 1.02 (Gjoka et al., 2010) and 1.2 (Kass et al.,

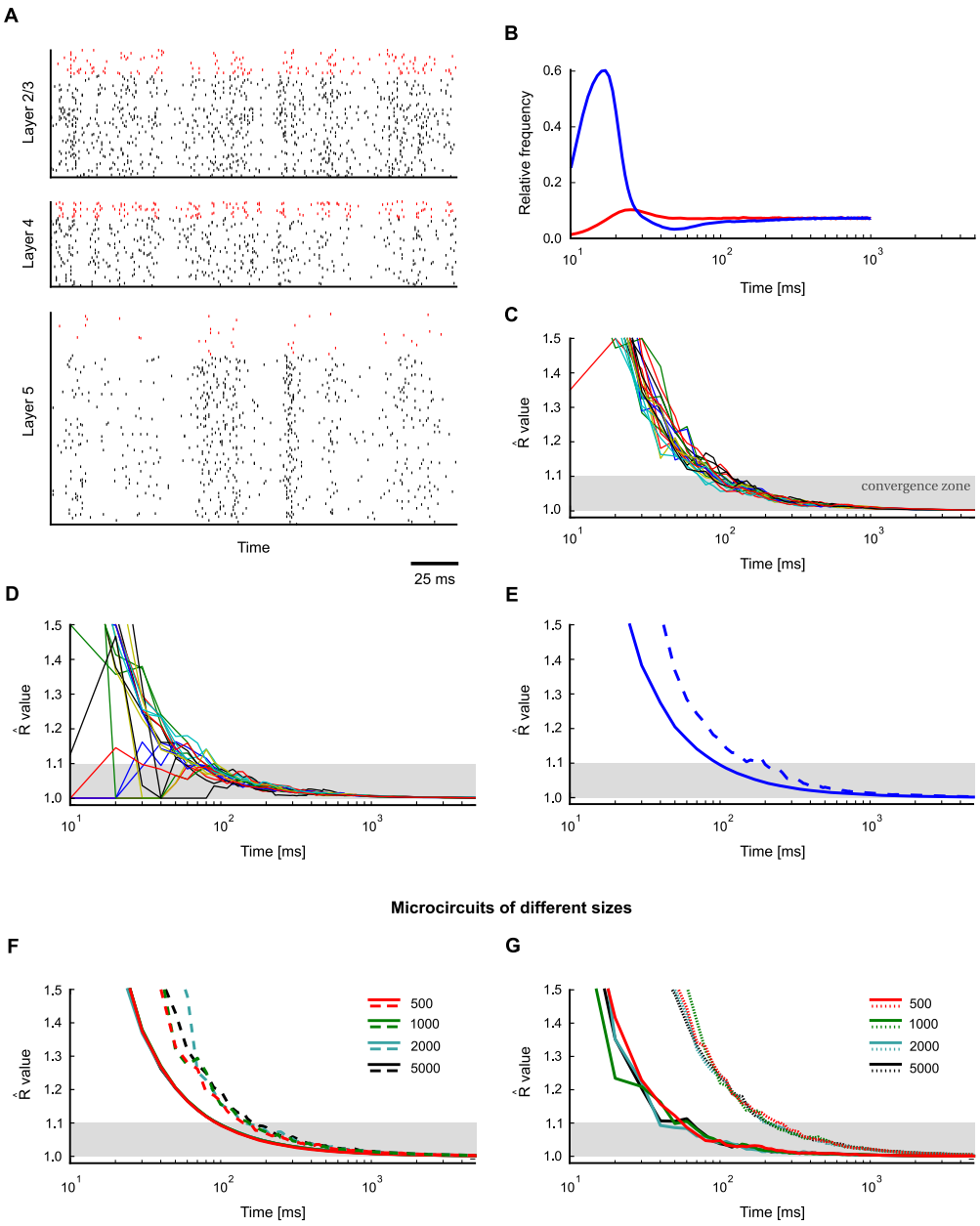

Figure 2.2: (see next page for Figure caption)

1998; Gelman et al., 2004; Brooks et al., 2010) have been proposed in the literature as thresholds below which the ratio $\hat{R}$ signals that convergence has taken place. The shaded region in Figure 2.2C-G corresponds to $\hat{R}$ values below a threshold

Figure 2.2: Fast convergence of marginals of single neurons and more complex quantities in a cortical microcircuit model. **A**. Typical spike response of the microcircuit model based on (Haeusler and Maass, 2007) comprising 560 stochastic point neurons. Spikes of inhibitory neurons are indicated in red. **B**. Fast convergence of a marginal for a representative layer 5 neuron (frequency of "on"-state, with $\tau = 10$ ms) to its stationary value, shown for two different initial Markov states (blue/red). Statistics were obtained for each initial state from 10^5 trials. **C**. Gelman-Rubin convergence diagnostic was applied to the marginals of all single neurons (simple states, $\tau = 10$ ms). In all neurons the Gelman-Rubin value $\hat{R}$ drops to a value close to 1 within a few 100 ms, suggesting generally fast convergence of single neuron marginals (shown are 20 randomly chosen neurons; see panel E for a summary of all neurons). The shaded area below 1.1 indicates a range where one commonly assumes that convergence has taken place. **D**. Convergence speed of pairwise spike coincidences (simple states (1,1) of two neurons, 20 randomly chosen pairs of neurons) is comparable to marginal convergence. **E**. Summary of marginal convergence analysis for single neurons in C: Mean (solid) and worst (dashed line) marginal convergence of all 560 neurons. Mean/worst convergence is reached after a few 100 ms. **F**. Convergence analysis was applied to networks of different sizes (500-5000 neurons). Mean and worst marginal convergence of single neurons are hardly affected by network size. **G**. Convergence properties of populations of neurons. Dotted: multivariate Gelman-Rubin analysis was applied to a subpopulation of 30 neurons (5 neurons were chosen randomly from each pool). Solid: convergence of a "random readout" neuron which receives spike inputs from 500 randomly chosen neurons in the microcircuit. It turns out that the convergence speed of such a generic readout neuron is even slightly faster than for neurons within the microcircuit (compare with panel E). A remarkable finding is that in all these cases the network size does not affect convergence speed.

of 1.1. An obvious advantage of the Gelman-Rubin diagnostic, compared with a straightforward empirical evaluation of convergence properties as in Figure 2.2B, is its substantially larger computational efficiency and the larger number of initial states that it takes into account. For the case of multivariate marginals (see Figure 2.2G), a straightforward empirical evaluation of convergence is not even feasible, since relative frequencies of 2^{30} states would have to be analyzed.

Using the Gelman-Rubin diagnostic, we estimated convergence speed for marginals of single neurons (see Figure 2.2C, mean/worst in Figure 2.2E), and for the product of the simple states of two neurons (i.e., pairwise spike coincidences) in Figure 2.2D. We found that in all cases the Gelman-Rubin value drops close to 1 within just a few 100 ms. More precisely, for a typical threshold of 1.2 convergence times are slightly below 100 ms in Figure 2.2C-E. A very conservative threshold of 1.02 yields convergence times close to 600 ms.

The above simulations were performed in a circuit of 560 neurons, but eventually one is interested in the properties of much larger circuits. Hence, a crucial question is how the convergence properties scale with the network size. To this end, we compared convergence in the cortical microcircuit model of (Haeusler and Maass, 2007) for four different sizes (500, 1000, 2000 and 5000). To ensure that

overall activity characteristics are maintained across different sizes, we adopted the approach of (Haeusler and Maass, 2007) and scaled recurrent postsynaptic potential (PSP) amplitudes inversely proportional to network size. A comparison of mean (solid line) and worst (dashed line) marginal convergence for networks of different sizes is shown in Figure 2.2F. Notably we find that the network size has virtually no effect on convergence speed. This suggests that, at least within the scope of the laminar microcircuit model of (Haeusler and Maass, 2007), even very large cortical networks may support fast extraction of knowledge (in particular marginals) from their stationary distributions $p_C(y|x)$.

In order to estimate the required computation time associated with the estimation of marginal probabilities and MAP solutions on small *subpopulations* $n_1, \ldots, n_m$, one needs to know how fast the marginal probabilities of *vector-valued* states $(y_1, \ldots, y_m)$ of subnetworks of C become independent from the initial state of the network. To estimate convergence speed in small subnetworks, we applied a multivariate version of the Gelman-Rubin method to vector-valued simple states of subnetworks (Figure 2.2G, dotted lines, evaluated for varying circuit sizes from 500 to 5000 neurons). We find that multivariate convergence of state frequencies for a population of $m = 30$ neurons is only slightly slower than for uni-variate marginals. To complement this analysis, we also investigated convergence properties of a "random readout" neuron which integrates inputs from many neurons in a subnetwork. It is interesting to note that the convergence speed of such a readout neuron, which receives randomized connections from a randomly chosen subset of 500 neurons, is comparable to that of single marginals (Figure 2.2F, solid lines), and in fact slightly faster.

2.6 Impact of different dynamic regimes on the convergence time

An interesting research question is which dynamic or structural properties of a cortical microcircuit model C have a strong impact on its convergence speed to the stationary distribution p_C. Unfortunately, a comprehensive treatment of this question is beyond the scope of this paper, since virtually any aspect of circuit dynamics could be investigated in this context. Even if one focuses on a single aspect, the impact of one circuit feature is likely to depend on the presence of other features (and probably also on the properties of the input). Nonetheless, to lay a foundation for further investigation, first empirical results are given in Figure 2.3.

As a reference point, Figure 2.3A shows a typical activity pattern and convergence speed of single marginals in the small cortical microcircuit model from Figure 2.1. To test whether the overall activity of a network has an obvious impact on convergence speed, we constructed a small network of 20 neurons (10 excitatory, 10 inhibitory) and tuned connection weights to achieve sparse overall activity (Figure 2.3B). A comparison of panels A and B suggests that overall network activity

has no significant impact on convergence speed. To test whether the presence of stereotypical trajectories of network states (similar to (Luczak et al., 2007)) has a noticeable influence on convergence, we constructed a small network exhibiting strong sequential activity patterns (see Figure 2.3C). We find that convergence speed is hardly affected, except for the first 200 ms (see Figure 2.3C). Within the scope of this first empirical investigation, we were only able to produce a significant slow-down of the convergence speed by building a network that alternated between two attractors (Figure 2.3D).

2.7 Distributions of network states in the presence of periodic input

In Theorem 1 we had already addressed one important case where the network C receives dynamic external inputs: the case when external input is generated by some Markov process. But many networks of neurons in the brain are also subject to more or less pronounced periodic inputs ("brain rhythms" (Engel et al., 2001; Buzsaki, 2009; Wang, 2010)), and it is known that these interact with knowledge represented in distributions of network states in specific ways. For instance, it had been shown in (Dragoi and Buzsaki, 2006) that the phase of the firing of place cells in the hippocampus of rats relative to an underlying theta-rhythm is related to the expected time when the corresponding location will be reached. Inhibitory neurons in hippocampus have also been reported to fire preferentially at specific phases of the theta cycle (see e.g. Figure S5 in (Kelemen and Fenton, 2010)). Moreover it was shown that different items that are held in working memory are preferentially encoded by neurons that fire at different phases of an underlying gamma-oscillation in the monkey prefrontal cortex (Siegel et al., 2009) (see (Pipa et al., 2009) for further evidence that such oscillations are behaviorally relevant). Phase coding was also reported in superior temporal sulcus during category representation (Turesson et al., 2012). The following result provides a theoretical foundation for such phase-specific encoding of knowledge within a framework of stochastic computation in networks of spiking neurons.

Theorem 2 (Phase-specific distributions of network states) *Let C be an arbitrary model for a network of stochastic spiking neurons as in Theorem 1. Assume now that the vector of input rates $x(t)$ has in addition to fixed components also some components that are periodic with a period L (such that each input neuron i emits a Poisson spike train with an L-periodically varying firing rate $x_i(t)$). Then the distribution of network states y converges for every phase l ($0 \leq l < L$) exponentially fast to a unique stationary distribution of network states $p_{C,l}(y|x)$ at this phase l of the periodic network input x.*

Hence, a circuit C can potentially store in each clearly separable phase l of an (externally) imposed oscillation a different, phase-specific, stationary distribution

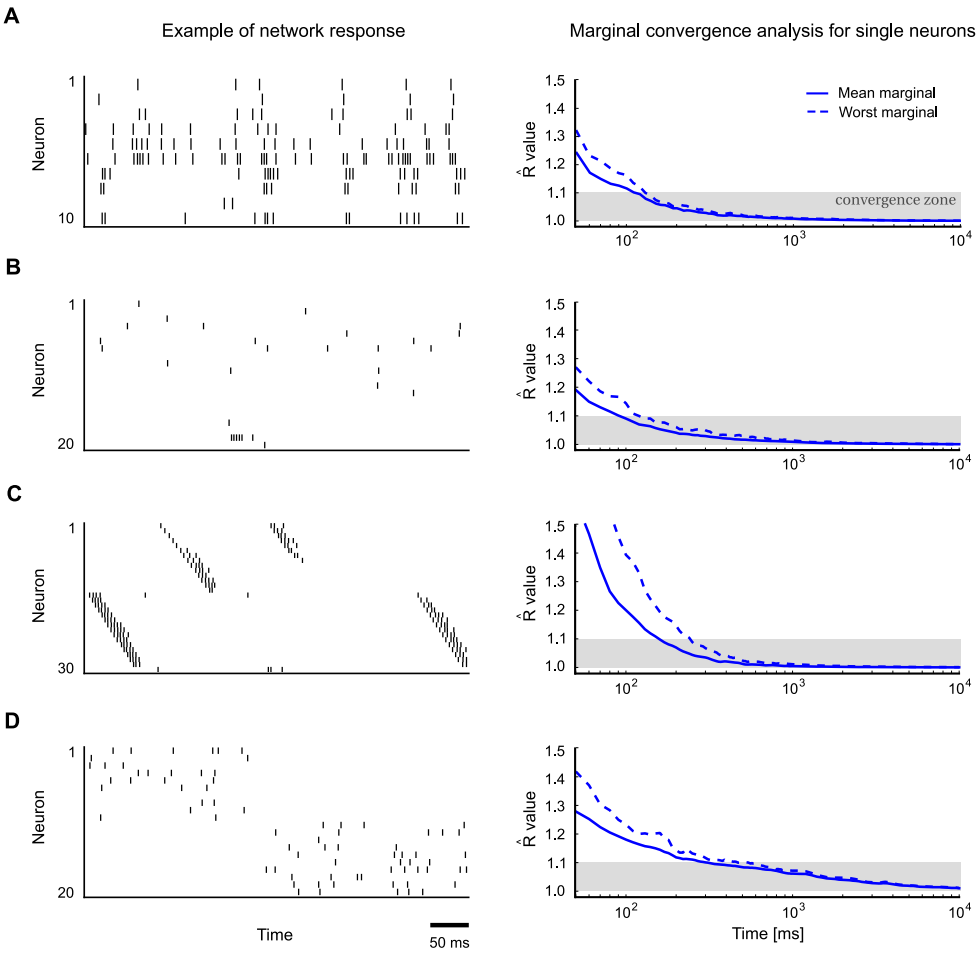

Figure 2.3: Impact of network architecture and network dynamics on convergence speed. Convergence properties for single neurons (as in Figure 2.2C) in different network architectures were assessed using univariate Gelman-Rubin analysis. Typical network activity is shown on the left, convergence speed on the right (solid: mean marginal, dashed: worst marginal). **A**. Small cortical column model from Figure 2.1 (input neurons not shown). **B**. Network with sparse activity (20 neurons). **C**. Network with stereotypical trajectories (50 neurons, inhibitory neurons not shown). Despite strongly irreversible dynamics, convergence is only slightly slower. **D**. Network with bistable dynamics (two competing populations, each comprising 10 neurons). Convergence is slower in this circuit due to low-frequency switching dynamics between two attractors.

$p_{C,l}(y|x)$. Below we will address basic implications of this result in the context of two experimentally observed phenomena: stereotypical trajectories of network states and bi-stable (or multi-stable) network activity.

Figure 2.4A-D demonstrates the emergence of phase-specific distributions in a small circuit (the same as in Figure 2.3C but with only one chain) with a built-in stereotypical trajectory similar to a spatial path sequence generated by hippocampal place cell assemblies (Dragoi and Buzsaki, 2006; Gupta et al., 2012). Figure 2.4A shows a typical spike pattern in response to rhythmic background stimulation (spikes from inhibitory neurons in red). The background oscillation was implemented here for simplicity via direct rhythmic modulation of the spiking threshold of all neurons. Note that the trajectory becomes particularly often initiated at a specific phase of the rhythm (when neuronal thresholds are lowest), like in experimental data (Dragoi and Buzsaki, 2006; Gupta et al., 2012). As a result, different phases within a cycle of the rhythm become automatically associated with distinct segments of the trajectory. One can measure and visualize this effect by comparing the frequency of network states which occur at two different phases, i.e., by comparing the stationary distributions $p_{C,l}(y|x)$ for these two phases. Figure 2.4B shows a comparison of phase-specific marginal distributions on a small subnetwork of 3 neurons, demonstrating that phase-specific stationary distributions may indeed vary considerably across different phases. Convergence to the phase-specific stationary distributions $p_{C,l}(y|x)$ can be understood as the convergence of the probability of any given state to a periodic limit cycle as a function of the phase l (illustrated in Figure 2.4C). An application of the Gelman-Rubin multivariate diagnostic suggests that this convergence takes places within a few cycles of the theta oscillation (Figure 2.4D).

Theta-paced spatial path sequences in hippocampus constitute a particularly well-studied example of phase-specific network activity(Dragoi and Buzsaki, 2006). Our theoretical framework suggests a novel interpretation of these patterns as samples from a Markov chain with a phase-dependent stationary distribution of network states induced by the theta-rhythm. A basic prediction of this interpretation is that two trajectories in successive theta cycles should exhibit significantly stronger similarities than two trajectories from randomly chosen cycles (due to inherent temporal dependencies of the Markov chain). Two trajectories from distant cycles, on the other hand, should relate to each other similarly as randomly chosen pairs of trajectories. Evidence for such an effect has been reported recently by (Gupta et al., 2012), where it was found that "sequences separated by 20 cycles approach random chance, whereas sequences separated by only a single theta cycle are more likely to be similar to each other."

The previously described theoretical framework also provides an interesting new perspective on multi-stability, a wide-spread phenomenon which has been observed in various sensory domains (Blake and Logothetis, 2002; Sterzer et al., 2009). Different authors have noted that multi-stability, both on the neuronal and perceptual level, could be understood as a side effect of sampling from a multi-modal distribution (Hoyer and Hyvärinen, 2003; Buesing et al., 2011; Gershman et al., 2012). Recent data from hippocampus suggest that oscillations, which had previously re-

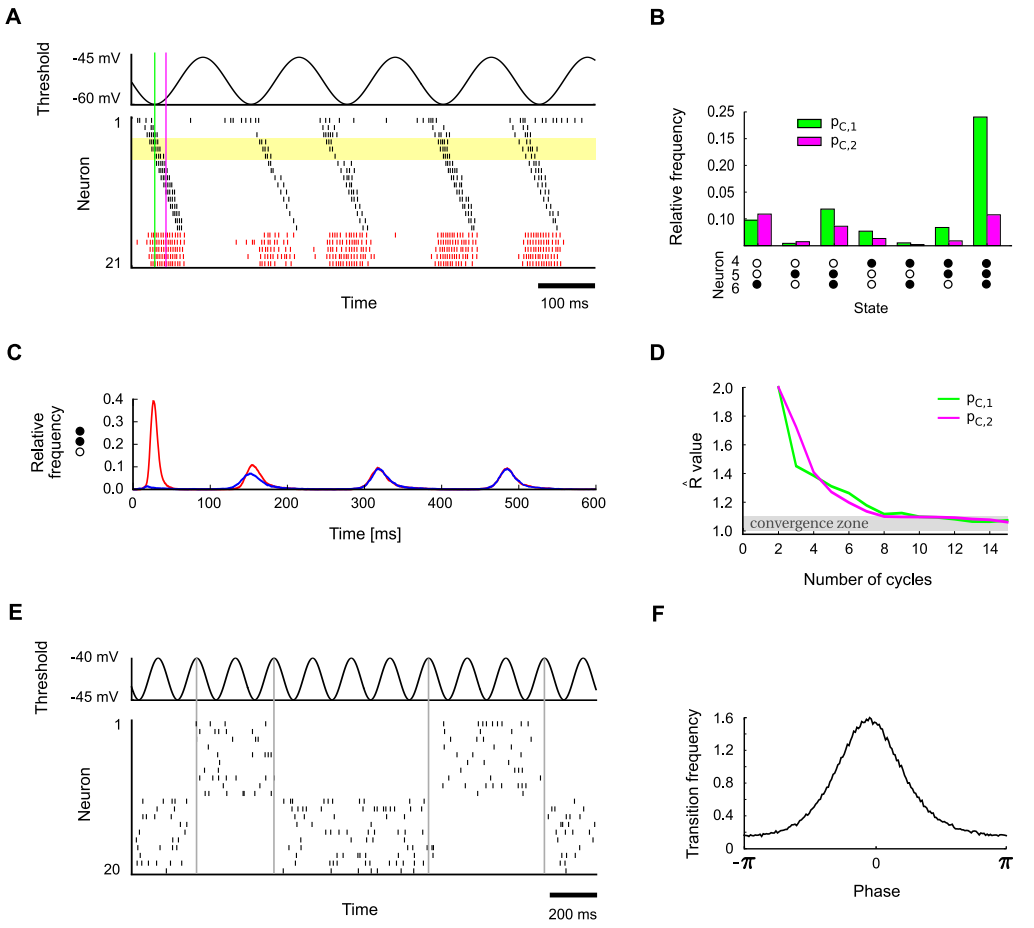

Figure 2.4: (see next page for Figure caption)

ceived little attention in this context, may play an important role here: (Jezek et al., 2011) found that switching between different attractors (= modes of the stationary distribution in our terminology) occurs preferentially at a specific phase during the theta cycle, whereas activity patterns within each cycle preferentially stayed in one attractor. Hence, the precise timing of switching between modes was found to be strongly tied to the theta rhythm. Such chunking of information in separate packages (theta cycles) has been proposed as an important constituent of neural syntax (Buzsáki, 2010).

In Figure 2.4E we reproduce phase-dependent switching in a simple network model of bi-stable dynamics (the same network as in Figure 2.3D) in the presence of a 6 Hz background oscillation. Indeed, we find that switching occurs preferentially at a specific phase of the oscillation (see Figure 2.4F) when the total firing rate of

Figure 2.4: Emergence of phase-specific stationary distributions of network states in the presence of periodic network input. **A**. A network with a built-in stereotypical trajectory is stimulated with a 6 Hz background oscillation. The oscillation (top) is imposed on the neuronal thresholds of all neurons. The trajectories produced by the network (bottom) become automatically synchronized to the background rhythm. The yellow shading marks the three neurons for which the analysis in panels B and C was carried out. The two indicated time points (green and purple lines) mark the two phases for which the phase-specific stationary distributions are considered in panels B and D (83 ms and 103 ms into the cycle, with phase-specific distributions $p_{C,1}$ and $p_{C,2}$, respectively). **B**. The empirically measured distributions of network states are observed to differ significantly at two different phases of the oscillation (phases marked in panel A). Shown is for each phase the phase-specific marginal distribution over 3 neurons (4, 5 and 6), using simple states with $\tau =$ 10 ms. The zero state (0,0,0) is not shown. The empirical distribution for each phase ϕ was obtained from a single long run, by taking into account the network states at times $\phi, \phi + T, \phi + 2T$, etc., with cycle length $T = \frac{1}{6}s$. **C**. Illustration of convergence to phase-specific stationary distributions. Shown is the relative frequency of subnetwork state (1,1,0) on the subset of neurons 4,5 and 6 over time, when the network is started from two different initial states (red/blue). In each case, the state frequency quickly approaches a periodic limit cycle. **D**. Convergence to phase-specific stationary distributions takes place within a few cycles of the underlying oscillation. Shown is the multivariate Gelman-Rubin convergence analysis to the phase-specific stationary distribution for two different phases. **E**. Bi-stable network under the influence of a 6 Hz background oscillation. **F**. In response to the periodic stimulation, transitions between the two attractors (modes) become concentrated around a specific phase of the distribution.

the network is lowest. Note that this is consistent with (Jezek et al., 2011) who found that the separation between representations in different cycles was strongest at the point of the lowest average firing rate in the population (see Figure 1b in (Jezek et al., 2011)). This phenomenon can be explained in our model by noting that the attractors are deeper during periods of high network activity. Conversely, attractors are more shallow when the population firing rate is lower, leading to an increased transition probability between attractors. If one takes a closer look at Proposition 1 and Lemma 1 in Appendix B one sees that this is also consistent with our theoretical framework: A lower population firing rate $\hat{\rho}$ translates into a smaller contraction factor $(1 - \epsilon^{\Theta})$, implying a tighter bound on the contraction speed of state distributions and thus higher transition probabilities to radically different states from the current (initial) network state.

Altogether, one sees that the presence of background oscillations has relevant functional implications on multi-stability. In particular, the presence of background oscillations in multi-stable networks facilitates both exploitation within a cycle and exploration across cycles: Within a cycle high firing rates force the network into one of the attractors, thereby avoiding interference with other attractors and facilitating the readout of a consistent network state. At the end of a cycle low firing rates allow the network to switch to different attractors, thereby promoting fast convergence

to the stationary distribution. The rhythmic deepening and flattening of attractors and the resulting phase-specific attractor dynamics could be particularly useful for the extraction of information from the circuit if downstream networks are phase-locked to the same rhythm, as reported, for example, for the interactions between neurons in hippocampus and prefrontal cortex (Siapas et al., 2005).

2.8 Generation of heuristic solutions to a constraint satisfaction problem

Whenever an inhibitory neuron fires, it reduces for a short while the probability of firing for its postsynaptic targets. In fact, new experimental data (Haider et al., 2013) show that inhibitory neurons impose quite powerful constraints on pyramidal cells. But also how pyramidal cells are embedded into their network environment imposes constraints on local network activity. From this perspective, the resulting firing patterns of a cortical microcircuit can be viewed as stochastically generated solutions of an immensely complex constraint satisfaction problem, that is defined both by external inputs x to the circuit and by the way each excitatory and inhibitory neuron is embedded into its circuit environment. Constraint satisfaction problems are from the computational perspective a particularly interesting class of problems, because many tasks that a brain has to solve, from the generation of a percept from unreliable and ambiguous sources to higher level tasks such as memory recall, prediction, planning, problem solving, and imagination, can be formulated as constraint satisfaction problems (Kumar, 1992). However, numerous constraint satisfaction problems are known to be NP-hard, thereby limiting the applicability of exact solution strategies. Instead, approximate or heuristic algorithms are commonly used in practice (for example evolutionary algorithms (Craenen et al., 2003)). Here we propose that networks C of spiking neurons with noise have an inherent capability to solve constraint satisfaction problems in an approximate (heuristic) manner through their stochastic dynamics. The key principle is that those network states y, which satisfy the largest number of local constraints, have the highest probability under the distribution $p_C(y|x)$. These constraints are imposed by the way each neuron of C is embedded into the circuit, and the current external input x which can selectively activate or deactivate specific constraints.

We have selected a specific constraint satisfaction problem for demonstrating the capability of networks of spiking neurons to generate rapidly approximate solutions to constraint satisfaction problems through their inherent stochastic dynamics: solving Sudoku puzzles (see Figure 2.5A). Sudoku is a well-suited example because it is complex enough to be representative for many problem solving tasks, and lends itself well to visual interpretation and presentation (but note that we do not aim to model here how humans solve Sudoku puzzles). The rules of the Sudoku game can be easily embedded into common models for cortical microcircuits as recurrent networks of Winner-Take-All (WTA) microcircuit motifs (Douglas and Martin, 2004).

Each WTA motif is an ensemble of pyramidal cells (on layers 2/3 or 5/6) that are subject to lateral inhibition (see Figure 2.5B). Each pyramidal cell can in fact be part of several interlocking WTA motifs (Figure 2.5B, right).

This architecture makes it easy to impose the interlocking constraints of Sudoku (and of many other constraint satisfaction problems). Each pyramidal cell (or each local group of pyramidal cells) votes for placing a particular digit into an empty field of the grid, that is not dictated by the external input x. But this pyramidal cell is subject to the constraints that only one digit can be placed into this field, and that each digit $1, \ldots, 9$ occurs only once in each column, in each row, and in each 3x3 sub-grid. Hence each pyramidal cell is simultaneously part of four inhibitory subnetworks (WTA motifs).

A specific puzzle can be entered by providing strong input x to those neurons which represent the given numbers in a Sudoku (Figure 2.5A, left). This initiates a quite intuitive dynamics: "Clamped" neurons start firing strongly, and as a consequence, neurons which code for conflicting digits in the same Sudoku field, the same row, column or 3x3 sub-grid, become strongly inhibited through di-synaptic inhibition. In many Sudoku fields this will lead to the inhibition of a large number of otherwise freely competing neurons, thereby greatly reducing the space of configurations generated by the network. In some cases, inhibition will immediately quieten all neurons except those associated with a single remaining digit (the only choice consistent with the givens). In the absence of competition, these uninhibited neurons will start firing along with the givens, thereby further constraining neighboring neurons. This form of inhibitory interaction therefore implicitly implements a standard strategy for solving easy Sudokus: checking for fields in which only one possibility remains. In harder Sudokus, however, this simple strategy alone would be typically insufficient, for example when several possibilities remain in all fields. In such cases, where inhibition leaves more than one possible digit open, a tentative digit will be automatically picked randomly by those neurons which happen to fire first among its competitors. This ensures that, instead of getting stuck, the network automatically explores potential configurations in situations where multiple possibilities remain. Altogether, through this combination of constraint enforcement and random exploration, those network states which violate few constraints (good approximate solutions) are visited with much higher probability than states with conflicting configurations. Hence, most time is spent in good approximate solutions. Furthermore, from all 9^{81} Sudoku configurations the solving configuration is visited in this process especially often.

Figure 2.5C shows a typical network run during the last 1.5 seconds (out of a total simulation time of approximately 3 s) before the correct solution was found to the Sudoku puzzle from Figure 2.5A. For this simulation we modeled lateral inhibition in each WTA motif by reciprocally connecting each neuron in the subnetwork to a single inhibitory neuron. For each of the 9 digits in a Sudoku field, we created an associated local group of four pyramidal cells. This can be seen in

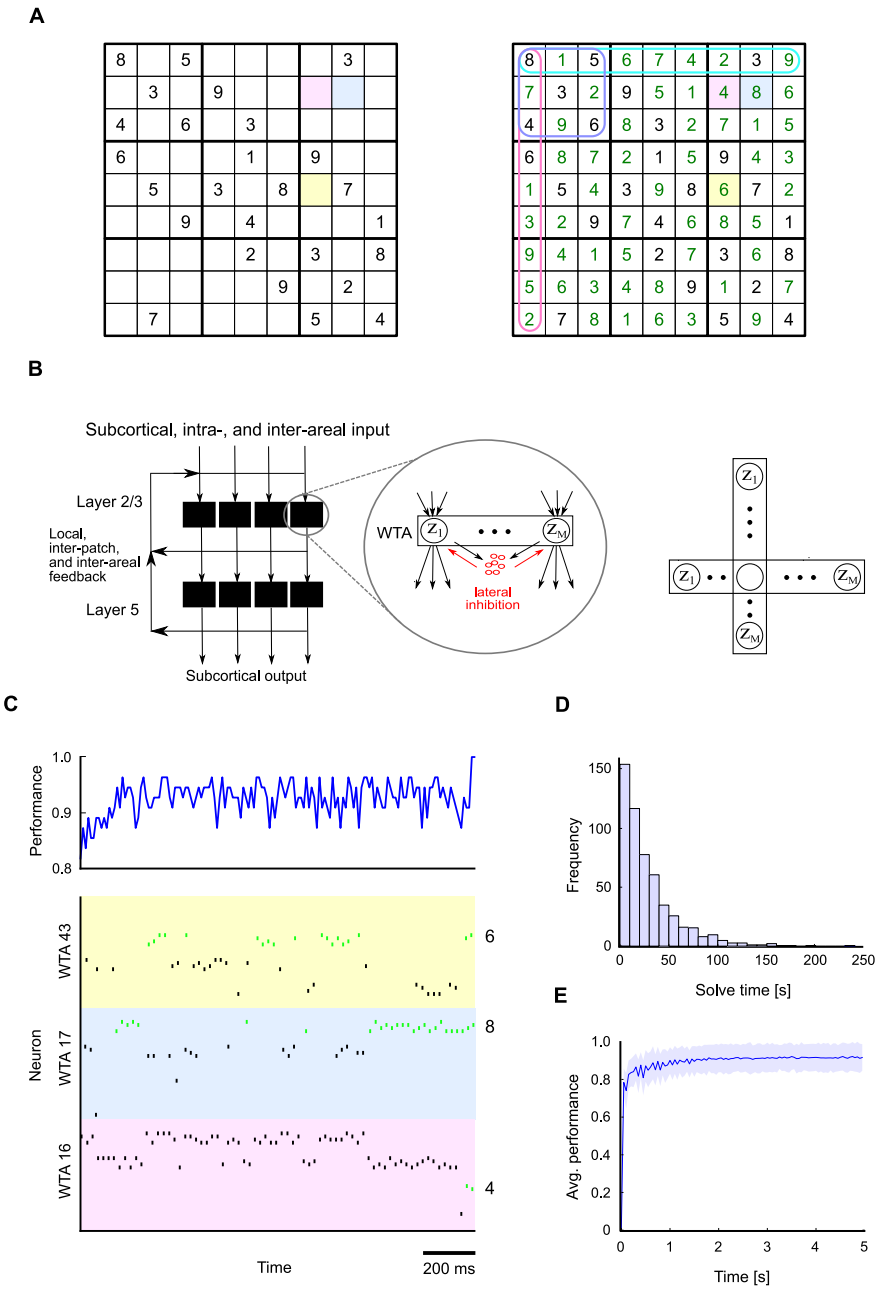

Figure 2.5: (see next page for Figure caption)

Figure 2.5C, where spike responses of pyramidal cells associated with three different Sudoku fields are shown (the three colored fields in Figure 2.5A and B). Each field

Figure 2.5: Solving Sudoku, a constraint satisfaction problem, through structured interactions between stochastically firing excitatory and inhibitory neurons. **A**. A "hard" Sudoku puzzle with 26 given numbers (left). The solution (right) is defined uniquely by the set of givens and the additional constraints that each digit must appear only once in each row, column and 3x3 subgrid. **B**. An implementation of the constraints of the Sudoku game in a spiking neural network C consists of overlapping WTA circuits. WTA circuits are ubiquitous connection motifs in cortical circuits (Douglas and Martin, 2004). A WTA circuit can be modeled by a set of M stochastically spiking output neurons z_k that are subject to lateral inhibition (left). The same pyramidal cell can be part of several such WTA motifs (right). In the Sudoku example, each digit in a Sudoku field is associated with four pyramidal cells which vote for this digit when they emit a spike. Each such pyramidal cell participates in four WTA motifs, corresponding to the constraints that only one digit can be active in each Sudoku field, and that a digit can appear only once in each row, column and 3x3 subgrid. **C**. A typical network run is shown during the last 1500 ms before the correct solution was found to the Sudoku from panel A (the total solve time was approximately $3s$ in this run, see panel D for statistics of solve times). The network performance (fraction of cells with correct values) over time is shown at the top. The spiking activity is shown for 3 (out of the 81) WTA motifs associated with the 3 colored Sudoku fields in A and B. In each of these WTA motifs there are 36 pyramidal cells (9 digits and 4 pyramidal cells for each digit). Spikes are colored green for those neurons which code for the correct digit in each Sudoku field (6, 8 and 4 in the example). **D**. Histogram of solve times (the first time the correct solution was found) for the Sudoku from panel A. Statistics were obtained from 1000 independent runs. The sample mean is 29 s. **E**. Average network performance for this Sudoku converges quickly during the first five seconds to a value of 0.9, corresponding to 90% correctly found digits (average taken over 1000 runs; shaded area: ± 2 standard deviations). Thereafter, from all possible 9^{81} configurations the network spends most time in good approximate solutions. The correct solution occurs particularly often, on average approximately 2% of the time (not shown).

has 9 possible digits, and each digit has four associated neurons. Hence, for each of the three Sudoku fields (WTA motifs), $9 \cdot 4 = 36$ neurons are shown. Spikes are colored black for those neurons which code for a wrong digit, and green for the four neurons which code for the correct digit in a Sudoku field (the correct digits in Figure 2.5C are 6, 8 and 4). The overall performance of the network (fraction of correctly solved fields) during the last 1.5 seconds before the solution is found is shown in Figure 2.5C above.

In our simulations we found that the solve time (the time until the correct solution is found for the first time) generally depends on the hardness of the Sudoku, in particular on the number of givens. For the "hard" Sudoku with 26 givens from Figure 2.5A, solve times are approximately exponentially distributed at an average of 29 seconds (Figure 2.5D). The average performance during the first five seconds of a run (obtained from 1000 independent runs) is shown in Figure 2.5E. The plot shows quick convergence to a (stationary) average performance of approximately 0.9. This demonstrates that the network spends on average most time in approximate solutions with high performance. Among these high-performance solutions,

the correct solution occurs especially often (on average 2% of the time).

2.9 Discussion

A theoretical foundation for memory-based stochastic computation in cortical microcircuits

We have shown that for common noise models in cortical microcircuits, even circuits C with very detailed and diverse non-linear neurons and synapses converge exponentially fast to a stationary distribution $p_C(y|x)$ of network states y. This holds both for external inputs x that consist of Poisson spike trains of a fixed rate, and for the case where x is periodic, or generated by some Markov process with a stationary distribution. The same mathematical framework also guarantees exponentially fast convergence to a stationary distribution of *trajectories* of network states (of any fixed time length), thereby providing a theoretical foundation for understanding stochastic computations with experimentally observed stereotypical trajectories of network states. These results extend and generalize previous work in (Brémaud and Massoulié, 1996) and (Borovkov et al., 2012) in two ways. First, previous convergence proofs had been given only for networks of simplified neurons in which the (sub-threshold) neuronal integration of pre-synaptic spikes was assumed a linear process, thereby excluding the potential effects of dendritic non-linearities or synaptic short-term dynamics. Second, previous work had focused only on the case where input is provided by neurons with fixed firing rates (a special case of Theorem 1). In addition we show that these convergence proofs can be derived from a fundamental property of stochastic spiking networks, that we have formulated as the Contraction Lemma (Lemma 1 in Appendix B).

The stationary distribution p_C provides an attractive target for investigating the stochastic computing capabilities of data-based models C for local circuits or larger networks of neurons in the brain. In contrast to the much simpler case of Boltzmann machines with non-spiking linear neurons and symmetric synaptic connections, it is unlikely that one can attain for cortical microcircuit models C a simple analytical description of p_C. But our computer simulations have shown that this is not necessarily an obstacle for encoding salient constraints for problem solving in p_C, and for merging knowledge that is encoded in p_C with online information from external inputs x in quite fast stochastic computations. In fact, the resulting paradigm for computations in cortical microcircuits supports anytime computing, where one has no fixed computation time. Instead, first estimates of computational results can be produced almost immediately, and can be rapidly communicated to other circuits. In this way, no processor (circuit) has to idle until other processors have completed their subcomputations, thereby avoiding the arguably most critical general bottleneck of massively parallel computing systems. Instead, each microcircuit C can contribute continuously to an iterative refinement of a global computation.

Estimates for the computation time of stochastic computations

Our computer simulations for a standard cortical microcircuit model C suggest that convergence to p_C is fast enough to support knowledge extraction from this distribution p_C within a few 100 ms, i.e. within the typical computation time of higher-level brain computations. These first estimates need to be corroborated by further theoretical work and computer simulations. In particular, the relationship between the structure and dynamics of cortical microcircuits and their convergence speed merits further investigation. Furthermore, in the case where p_C is a multi-modal distribution there exists an obvious tradeoff between the convergence speed to p_C and the typical duration of staying in an "attractor" (i.e., a region of the state space which has high probability under p_C). Staying longer in an attractor obviously facilitates the readout of the result of a computation by downstream networks. A number of experimental data suggest that neuromodulators can move neural circuits (at least in the prefrontal cortex) to different points on this tradeoff curve. For example it is argued in (Durstewitz, 2006, 2009) that the activation of D_1 receptors through dopamine deepens all basins of attraction, making it harder for the network state to leave an attractor. Additional molecular mechanisms that shift the tradeoff between fast sampling (exploration) and the temporal stability of found solutions are reviewed in (Arnsten et al., 2012). Another interesting perspective on convergence speed is that slow convergence may be beneficial for certain computations in specific brain areas (especially early sensory areas). Slow convergence enlarges the time span during which the network can integrate information from non-stationary external inputs (Maass et al., 2002; Nikolic et al., 2009; Klampfl et al., 2012). In addition the initial state y_0 of a network may contain information about preceding events that are computationally useful. Those considerations suggest that there exist systematic differences between the convergence speed to p_C in different neural systems C, and that it can be modulated in at least some systems C dependent on the type of computational task that needs to be solved.

Another important issue is the tradeoff between sampling time and sampling accuracy. In high-level cognitive tasks, for example, it has been argued that "approximate and quick" sample-based decisions are often better than "accurate but slow" decisions (Vul et al., 2009; Lieder et al., 2013). Of particular interest in this context is the analysis of (Lieder et al., 2013) who studied the time-accuracy tradeoff during decision making, under the assumption that the mind performs inference akin to MCMC sampling. Due to the nature of MCMC sampling, early samples before convergence (during the burn-in period) are biased towards the initial state of the system. In the absence of time pressure, the optimal strategy is therefore to wait and collect samples for a long period of time (in theory indefinitely). In the presence of even moderate time costs, however, the optimal sampling time can be shown to be finite, a result which can provide a rational explanation of the anchoring effect in cognitive science (Lieder et al., 2013) (under time pressure people's decisions are influenced by their "initial state"). Notably, the analysis of (Lieder et al., 2013)

was based on the assumption that the MCMC algorithm exhibits geometric convergence, the discrete-time equivalent to the exponential convergence speed proved in this paper for stochastic spiking networks. Applying a similar analysis to study optimal time-accuracy tradeoff points in cortical microcircuits therefore presents a promising avenue for future research.

Which probability distributions can be encoded as a stationary distribution of some neural circuit?

It had been shown in (Buesing et al., 2011) and (Pecevski et al., 2011) that, under certain assumptions on the neuron models and circuit structure, in principle every joint distribution p over discrete-valued random variables can be represented as a stationary distribution p_C of some network C of spiking neurons. Forthcoming unpublished results suggest that such internal representations of a given distribution p can even be learned from examples drawn from p. This will provide a first step towards understanding how the stationary distribution p_C of a microcircuit can be adapted through various plasticity processes to encode salient constraints, successful solution strategies (rules), and other types of knowledge. This research direction promises to become especially interesting if one takes into account that knowledge can not only be encoded in the stationary distribution of network states, but also in the simultaneously existing stationary distribution of trajectories of network states.

Relationship to attractor networks and transients between attractors

Attractor neural networks (Hopfield, 1982) were originally deterministic computational models, where gradient descent leads the network from some given initial state y_0 (the input for the computation) to the lowest point of the attractor (the output of the computation) in whose basis of attraction y_0 lies. The computational capability of an attractor neural network is substantially larger if its attractor landscape can be reconfigured on the fly by external input x, as in (Hopfield and Tank, 1986) and in the Sudoku example of this article. This usually requires that the attractors are not programmed directly into the network parameters, but emerge from some more general computational principles (e.g. constraint satisfaction). Attractor neural networks gain additional computational capability if there is some noise in the system (Rolls and Deco, 2010). This enables the network to leave after a while suboptimal solutions (Durstewitz and Deco, 2008). Alternative modeling frameworks for the transient dynamics of neural systems are provided by the liquid computing model (Maass et al., 2002), and on a more abstract level by sequences of metastable states in dynamical systems (Rabinovich et al., 2008). Here we propose to view both transient and attractor dynamics of complex data-based circuits C from the perspective of probabilistic inference, in particular as neural sampling (Buesing et al., 2011) (or more abstractly: as MCMC sampling) from their inherent

probability distribution p_C over network states (or trajectories of network states), that serves as the knowledge base of these neural systems.

A new computational framework for analyzing brain activity

We had focused in our computer simulations on the investigation of the stationary distribution p_C for models C of cortical microcircuits. But the results of Theorem 1 and Theorem 2 are of course much more general, and in principle apply to models C for networks of neurons in the whole brain (Sporns, 2011). This perspective suggests understanding spontaneous brain activity (see (Raichle, 2010)) as sampling from this global distribution in the absence of external input, and brain computations with external inputs x as sampling of brain states from conditional distribution $p_C(y|x)$, thereby merging the knowledge base p_C of the brain with incoming new information x. This computational framework could in principle explain how the brain can merge both types of information in such seemingly effortless manner, a capability that can only partially be reproduced in artificial devices with current technology. Large-scale computer simulations will be needed to test the viability of this hypothesis, in particular the relationship between the known global structure of the brain network C and properties of its stationary distribution p_C, and the convergence speed to p_C. Possibly the brain uses an important trick to speed up convergence during brain-wide sampling, for example by sampling during any concrete brain computation only from a subnetwork C' of C: those brain areas that control variables that are relevant for this computation. Functional connectivity would be explained from this perspective as opening of communication channels that support sampling from the (marginal) joint distribution of those variables that are stored within the functionally connected brain areas. Structured spontaneous brain activity (Raichle, 2010) would then receive a functional interpretation in terms of updating these marginal joint distributions on the basis of newly acquired knowledge.

Stochastic solutions of constraint satisfaction problems as a paradigm for higher level brain computation

A surprisingly large number of computational tasks that the brain has to solve, from the formation of a percept from multi-modal ambiguous sensory cues, to prediction, imagination, motor planning, rule learning, problem solving, and memory recall, have the form of constraint satisfaction problems: A global solution is needed that satisfies all or most of a set of soft or hard constraints. However, this characterization per se does not help us to understand how the brain can solve these tasks, because many constraint satisfaction problems are computationally very demanding (in fact, often NP-hard (Garey and Johnson, 1979)), even for a fast digital computer. In the Sudoku example we have shown that the inherent stochastic dynamics of cortical microcircuits provides a surprisingly simple method for generating

heuristic solutions to constraint satisfaction problems. This is insofar remarkable, as this computational organization does not require that specific algorithms are programmed into the network for solving specific types of such problems (as it is for example needed for solving Sudoku puzzles according to the ACT-R approach (Qin et al., 2012)). Rather, it suffices that salient constraints are encoded into the network (e.g. through learning) in such a way that they make certain firing patterns of a subset of neurons more or less likely.

Future work will need to investigate whether and how this approach can be scaled up to larger instances of NP-complete constraint satisfaction problems. For example, it will be interesting to see whether stochastic networks of spiking neurons can also efficiently generate heuristic solutions to energy minimization problems (Boykov et al., 2001) arising in visual processing.

Furthermore, additional research is needed to address suitable readout mechanisms that stabilize and evaluate promising candidate solutions (see (Arnsten et al., 2012) for an experimentally supported mechanism that might contribute to this function). This is an important issue since, in its current form, the network will simply continue the stochastic exploration of heuristic solutions even after it has found the optimal solution. Therefore, in the absence of additional mechanisms the network is not able to hold on to (or store) previously found (near-)optimal solutions. To solve this issue one could consider, for example, one or several networks $C_1, \ldots, C_i$ which generate in parallel heuristic solutions to a given problem. The output of these networks could then be further processed and integrated by a readout network C_{i+1} which attempts to extract a MAP solution, for example by adopting a solution from some C_j only if it has higher value than the currently stored state. Hence, the sampling networks $C_1, \ldots, C_i$ would have stationary distributions $p_{C_j}(y|x)$ which encourage exploration and broadly assign probability to many different heuristic solutions, whereas the readout network would ideally exhibit a sharply peaked stationary distribution at the global optimum of the constraint satisfaction problem. Studying the feasibility of this approach requires further research.

Relationship to models for probabilistic inference in cognitive science

A substantial number of behavioral studies in cognitive science (see e.g. (Griffiths and Tenenbaum, 2006; Vul and Pashler, 2008; Denison et al., 2009; Gershman et al., 2012; Tenenbaum et al., 2011)) have arrived at the conclusion that several of the previously discussed higher level mental operations are implemented through probabilistic inference. Some of the underlying data also suggest that probabilistic inference is implemented in the brain through some form of sampling (rather than through arithmetical approaches such as belief propagation (Koller and Friedman, 2009)). But according to (Tenenbaum et al., 2011): "The key research questions are as follows: What approximate algorithms does the mind use, how do they relate to engineering approximations in probabilistic AI, and how are they implemented in

neural circuits?" This article contributes to these fascinating questions by providing a rigorous theoretical foundation for the hypothesis that neural circuits in the brain represent complex probability distributions $p_C(y|x)$ through sampling. In addition, we have provided evidence that this form of sampling in cortical microcircuits may be fast enough to facilitate the approximate estimation of marginals or marginal MAP assignments, which commonly appear in real-world inference tasks, within a few 100 ms. A major challenge for future work will be to understand also neuronal plasticity on the implementation level from this perspective. For example, how can prior knowledge be acquired and integrated into the stationary distribution $p_C(y|x)$ of a realistic circuit C (featuring short-term plasticity, dendritic processing, etc.) in an autonomous fashion, and in a manner consistent with statistically optimal learning (Fiser et al., 2010)?

Long-term plasticity and other slower features of network dynamics

In biological networks it is reasonable to assume that the network dynamics unfolds on a continuum of time scales from milliseconds to days. Our goal in this article was to focus on stochastic computations on shorter time scales, between a few milliseconds to seconds. To this end we assumed that there exists a clear separation of time scales between fast and slow dynamical network features, thus allowing us to exclude the effect of slower dynamical processes such as long-term plasticity of synaptic weights during these shorter time scales. In network models and experimental setups where slower processes significantly influence (or interfere with) the dynamics on shorter time scales, it would make sense to extend the concept of a stationary distribution to include, for example, also the synaptic parameters as random variables. A first step in this direction has been made for neurons with linear sub-threshold dynamics and discretized synapses in (Borovkov et al., 2012).

Deterministic network models and chaos

Deterministic network models such as leaky integrate-and-fire neurons without noise (no external background noise, no synaptic vesicle noise and no channel noise) violate the assumptions of Theorem 1 and 2. Furthermore, although realistic neurons are known to possess various noise sources, the theoretical assumptions could in principle still fail if the network is not *sufficiently* stochastic: this would happen, for example, if there exists some strong input (within the limits of typical input activity) which entirely overrules the noise, leading to a firing probability 1 in some time interval $[t, t + \delta t)$ during the network simulation. Such deterministic behavior would correspond to the instantaneous firing rate of a stochastic neuron becoming infinite at some point during that interval (in violation of assumption A2, see Appendix B: Scope of theoretical results). From an empirical perspective, a simple necessary condition for sufficient stochasticity is the presence of trial-to-trial variability for each single spike produced by a network. Consider, for example, the spike

times generated by a specific neuron in a network simulation, in response to some fixed input spike train. If there exists a spike which always occurs at the exact same time during multiple repetitions of this experiment starting from identical initial states, then the assumptions of Theorem 1 and 2 are obviously violated.

For deterministic (or insufficiently stochastic) networks the question arises whether convergence to a unique stationary distribution may still occur under appropriate conditions, perhaps in some modified sense. Notably, it has been recently observed that deterministic networks may indeed lead to apparently stochastic spiking activity (Churchland and Abbott, 2012; Litwin-Kumar and Doiron, 2012). This apparent stochasticity was linked to chaotic spiking dynamics. This suggests that chaos may act as a substitute for "real" noise in deterministic networks (similar to pseudo random-number generators emulating true randomness): Chaotic systems are sensitive to small perturbances in initial conditions, and may thus exponentially amplify otherwise insignificant noise sources such as ubiquitous thermal noise (Clarke, 2012). Thus, chaos could play an important role in both emulating and amplifying stochasticity on the network level.

(Litwin-Kumar and Doiron, 2012) focused their analysis of stochasticity on firing rate fluctuations and spiking irregularity, and it remains unclear whether these networks would still appear stochastic if one takes into account full network states (as in this article). The Gelman-Rubin convergence analysis of population activity proposed in this paper could be applied to provide some insight into this question. A more thorough investigation of chaos in the context of our results would also call for a rigorous theoretical analysis of ergodic properties of chaotic spiking networks.

Further experimentally testable predictions

Our theoretical results demonstrate that every neural system C has a stationary distribution $p_C(y|x)$ of network states y. This can be tested experimentally, for various behavioral regimes and external inputs x. A first step in this direction has already been carried out in (Berkes et al., 2011) (see also the discussion in (Okun et al., 2012)). The hypothesis that p_C serves (for "neutral" external inputs x) as a prior for probabilistic inference through sampling suggests that p_C is constantly modified through prior experience (see (Zhang et al., 2012; Xu et al., 2012) for first results) and learning (see (Lewis et al., 2009) for fMRI data).

Our Theorem 2 suggests in addition that neural systems C that have a prominent rhythm (such as for example the theta oscillation in the hippocampus) are able to store *several* stationary distributions $p_{C,l}$ of network states, one for each clearly separable phase l of this rhythm. It has already been shown in a qualitative manner that in some behavioral situations certain states y appear with substantially high probability at specific phases l of the rhythm (see e.g. (Harris et al., 2003; Buzsaki, 2009; Siegel et al., 2009; Gupta et al., 2012; Turesson et al., 2012)). But a systematic experimental analysis of phase-dependent distributions of network states in the style of (Berkes et al., 2011) is missing.

Our Theorem 1 predicts in addition that a generic neural circuit C also has a stationary distribution over *trajectories* of network states. The existence of stereotypical trajectories of network states in the awake brain has been frequently reported (see e.g. (Abeles et al., 1995; Jones et al., 2007; Luczak et al., 2007; Zhang et al., 2012)). But a statistical analysis of the distribution of such trajectories, especially also during spontaneous activity, is missing. Of particular interest is the relationship between the distribution of trajectories and the stationary distribution of (simple) network states. Do some network states y typically have a high probability because they occur in some high probability trajectory? And how does the distribution of trajectories change during learning?

The model for problem solving that we have presented in Figure 2.5 suggests that external constraints have a significant and characteristic impact on the structure of the stationary distribution p_C, by reducing the probability of network states which are inconsistent with the current constraints x. In principle, this could be analyzed experimentally. In addition, this model suggests that there may be special mechanisms that prolong the time span during which a neural system C stays in a network state y with high probability under $p_C(y|x)$, in order to support a readout of y by downstream networks. These mechanisms need to be revealed through experiments.

New ideas for neuromorphic computation

The Sudoku example has shown that networks of spiking neurons with noise are in principle able to carry out quite complex computations. The constraints of many other demanding constraint satisfaction problems, in fact even of many NP-complete problems, can be encoded quite easily into circuit motifs composed of excitatory and inhibitory spiking neurons, and can be solved through the inherent stochastic dynamics of the network. This provides new computational paradigms and applications for various energy-efficient implementations of networks of spiking neurons in neuromorphic hardware, provided they can be equipped with sufficient amounts of noise. In particular, our results suggest that attractive computational properties of Boltzmann machines can be ported into spike-based hardware. These novel stochastic computing paradigms may also become of interest for other types of innovative computer hardware: Computer technology is approaching during the coming decade the molecular scale, where noise is abundantly available (whether one wants it or not) and it becomes inefficient to push through traditional deterministic computing paradigms.

Conclusion

The results of this article show that stochastic computation provides an attractive framework for the investigation of computational properties of cortical microcircuits, and of networks of microcircuits that form larger neural systems. In particu-

lar it provides a new perspective for relating the structure and dynamics of neural circuits to their computational properties. In addition, it suggests a new way of understanding the organization of brain computations, and how they are modified through learning.

2.10 Acknowledgments

This chapter is based on a joint work with Zeno Jonke (ZJ) and Wolfgang Maass (WM) that was published 2013 in PLOS Computational Biology ("Stochastic Computations in Cortical Microcircuit Models"). The initial "inverse problem" idea, to consider under what circumstances the concept of a stationary distribution may be applicable in more general networks of spiking neurons than those previously considered, came from WM who also supervised the project. I developed the theory of the paper. The broad strokes of paper organization and design of computer experiments were then conceived in a series of animated discussions, in which all three authors SH, ZJ and WM, were involved equally. ZJ developed the software simulation framework and the analysis tools and conducted the computer experiments. The simulation framework was based on NEVESIM, an event based simulator for networks of spiking neurons written by Dejan Pecevski. The figures were developed and designed by SH, ZJ and WM. The main text was written by SH and WM.

Chapter 3

Networks of Spiking Neurons with Noise can Solve Hard Computational Problems

Contents

Networks of neurons in the brain compute and communicate very differently from transistors in digital computers: with unsynchronized short pulses, called action potentials or spikes. But it has remained unknown how difficult computational problems could be solved in this way. We present here new principles of spike-based computation with noise that enable networks of spiking neurons to carry out a very efficient stochastic search in high-dimensional spaces, thereby producing fast approximate solutions to hard computational problems such as logical inference (SATISFIABILITY) and planning (TRAVELING SALESMAN PROBLEM). The underlying computational theory that we present also suggests new methods for organizing massively parallel computations in novel energy-efficient but noisy computing hardware.

3.1 Introduction

Despite the astonishing advancements of digital computing in the past decades, the human brain is still considered the most powerful, versatile and "intelligent" computing device. Most of the remarkable mental faculties of humans, from imagination, prediction, and creative problem solving to abstract thought, are unrivaled by the most powerful supercomputers. This is achieved by the brain with only ~ 25 Watt of energy consumption (Kandel et al., 1991) (several order of magnitude less

than its most powerful digital counterparts), without the need for a global clock through inherently asynchronous communication (Gerstner and Kistler, 2002), and in spite of powerful noise sources introducing random variability at virtually every step of neural computation, from spike generation, to action potential propagation, to synaptic transmission (Faisal et al., 2008). Despite decades of research, however, it is still largely unknown how complex computations, beyond mere sensory processing, could be implemented in neural circuits on the basis of such noisy asynchronous computing units.

In this article we present a theoretical framework and four new principles for circuit design with spiking neurons that demonstrate how the inherent stochasticity and asynchronous dynamics of neural circuits can be systematically exploited to solve hard computational problems. We report that the application of this new theoretical framework leads to a qualitative jump in the computational capabilities of networks of spiking neurons.

3.2 New design principles for spike-based computation

A spiking neuron responds to stimulation by emitting short pulses, called action potentials or spikes. Spikes occur asynchronously (in continuous time) and are communicated to other neurons via inhibitory or excitatory synaptic connections (Figure 3.1A, top). Biological spiking neurons are inherently noisy (Faisal et al., 2008). We model the stochastic spiking behavior of a neuron k via an instantaneous firing probability (or firing rate), $\rho_k(t)$,

$$\rho_k(t) = \frac{1}{\tau} \exp(u_k(t)) \ , \tag{3.1}$$

the magnitude of which depends on the current *membrane potential* $u_k(t)$ of the neuron. The membrane potential is defined as the weighted sum of the neuron's inputs,

$$u_k(t) = b_k + \sum_l w_{kl}\, x_k(t) \ . \tag{3.2}$$

The additional bias term b_k represents the intrinsic excitability of neuron k. After each emitted spike, neuron k enters a refractory period of length τ before it can re-spike. A spike by neuron k is transmitted via synaptic connections to all post-synaptic neurons receiving input from neuron k. The effect of a spike on a post-synaptic neuron l, the so-called post-synaptic potential (PSP), is short-lived and can be either inhibitory or excitatory, depending on the sign of the synaptic weight w_{lk}. In general, PSPs can assume complex shapes and the effective duration of a PSP may depend on various dynamically changing factors. Here we assume for mathematical tractability a rectangular shape with a fixed length $\tau = 10$ms as shown in Figure 3.1A, such that $x_k(t) = 1$ if a spike occurred within $(t - \tau, t]$, and $x_k(t) = 0$ otherwise.

The *network state* at time t is defined as the vector of neural states $\mathbf{x}(t) = (x_1(t), x_2(t), \ldots, x_N(t))$, i.e. only those neurons are set to 1 which fired recently (Figure 3.1A, middle). Due to network connectivity some network states $\mathbf{x}(t)$ will naturally occur more often on average than others. Consider the distribution of network states that can be measured empirically by observing network activity over a long period (Figure 3.1A, bottom). In general networks of spiking neurons, the resulting long-term distribution will depend on the initial state of the network at the beginning of the experiment. If the network is sufficiently stochastic, however, the distribution over network states becomes independent of the initial state. In networks composed of stochastic neurons of the type (3.1), a sufficient condition for this to occur is that all excitatory weights in the network are bounded. Note that when such a *unique stationary distribution* $p(\mathbf{x})$ exists, it reflects which network states can be most likely observed after the vanishing of transients (after convergence to equilibrium). In analogy with statistical physics, we define the energy function of a sufficiently stochastic network as $E(\mathbf{x}) = -\log p(\mathbf{x}) + \text{const}$. According to this definition low energy states correspond to likely network states after convergence to equilibrium.

We present a set of four new principles of circuit design with spiking neurons. *Principle 1* – the foundation of our framework – states that one should add sufficient stochasticity to a (possibly otherwise deterministic) network with spiking neurons so that the network has a *unique stationary distribution* $p(\mathbf{x})$ of network states (Figure 3.1A). For stochastic neurons (3.1) adding further noise is obviously not required. In order to use such a stochastic network to solve a given computational task, the circuit should then be constructed in such a manner that the stationary distribution of network states $p(\mathbf{x})$ assumes especially high values for circuit states that encode good solutions to the computational task. Equivalently, the energy function $E(\mathbf{x}) = -\log p(\mathbf{x}) + \text{const}$ of a circuit should be particularly low for states $\mathbf{x}$ representing solutions to the problem.

Principle 2 states that the energy function $E(\mathbf{x})$ over a set of *principal neurons* can be systematically shaped through the use of a few auxiliary circuit motifs (Figure 3.1B). In particular we present two circuit motifs, the winner-take-all (WTA) and the OR motif, that can be used to impose powerful higher-order constraints on the activity of principal neurons in order to encode a variety of hard computational problems. The WTA circuit motif, applied to some set of principal neurons, increases the energy (decreases the probability) of all network states where not exactly (i.e. not more and not less than) one principal neuron in the WTA circuit is active. The OR motif increases the energy of all states where none of the involved principal neurons is active. A third way of systematically shaping energies is to add bi-directional and symmetric synaptic connections between two neurons k and l which either increase or decrease the energy of network states where both involved neurons are active, $x_k = 1, x_l = 1$ (via inhibitory or excitatory connections, respectively).

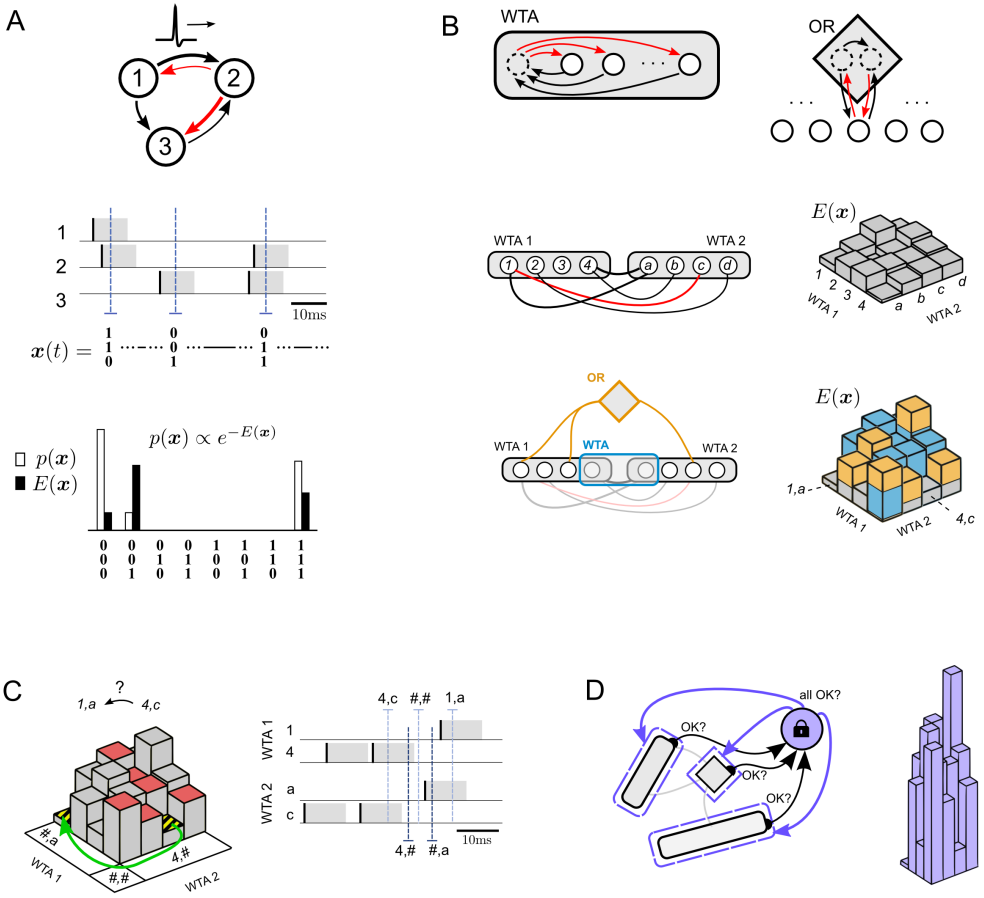

Figure 3.1: (see next page for Figure caption)

One important use of the WTA motif is to represent discrete random variables (RV): Consider a set of K principal neurons, where each neuron codes for one out of K possible values of a discrete RV. Then, by applying the WTA motif to these neurons one can ensure that most of the time only one of the principal neurons is active and, as a result, most of the time the RV has a well-defined value which can be derived from the current network state. Nevertheless, it may happen that for some short period of time none (or more than one) of the K neurons is active. During that period, the value of the represented RV is then considered undefined. The WTA motif ensures that such periods are very brief when they occur. Various computational problems, including 3-SAT and TSP, can be represented in terms of a number of discrete RVs with a finite state space. Hence, the representation of

Figure 3.1: Four new principles of circuit design with spiking neurons. **A**. Unlike transistors in a digital computer, spiking neurons communicate via brief asynchronous pulses called spikes (inhibitory/excitatory synaptic connections shown in red/black, connection line width represents synaptic strength). To study the collective behavior of spiking networks, we consider joint network states $\mathbf{x}(t)$, a binary vector where all neurons which fired recently (within the last 10ms) are assigned 1, over time t. The stationary distribution of network states, $p(\mathbf{x})$, reflects which network states can be likely observed after transients have vanished. We propose that $p(\mathbf{x})$, or equivalently the energy function $E(\mathbf{x}) = -\log p(\mathbf{x}) + \text{const}$, represents the computational output of a spiking network with noise (Principle 1). **B**. The energy function $p(\mathbf{x})$ over a set of *principal neurons* (white) can be shaped in a systematic manner by adding circuit motifs (shown in gray). Top row: Two motifs winner-take-all (WTA) and OR. These circuit motifs constrain activity patterns such that most of the time exactly one (WTA) or at least one (OR) in the set of connected principal neurons is active. Middle/bottom row: Example network consisting of two WTA circuits, each representing a discrete random variable (RV) with four possible values (middle left). Symmetric synaptic connections (only some shown) can be used to modulate the energies of different joint assignments to these two variables (middle right). Additional auxiliary circuits can be added (bottom left) to shape the energy landscape in more complex ways (bottom right). Those network states which violate the fewest of the imposed constraints have lowest energy (such as the highlighted states $1, a$ and $4, c$). The energy contributions of different auxiliary circuits sum up linearly. This facilitates the construction of complex energy landscapes through repeated use of simple circuit motifs (Principle 2). **C**. Stochastic search for low energy states is facilitated by the asymmetry of spike-based signaling. Direct transitions between the two low-energy states $1, a$ and $4, c$ are blocked due to high energy intermediate states (marked in red). An alternative route goes over a series of states where the value of one or both discrete RVs is briefly undefined (marked by #). Such "exploratory" periods of undefined RVs occur particularly frequently (and briefly) in a spiked-based communication scheme (Principle 3). **D**. In contrast to traditional stochastic search algorithms, the search process in a physical implementation cannot be "stopped" when a satisfactory solution has been found. Instead, internal temperature control is proposed as a principled alternative for high-speed computing systems: each circuit motif detects and reports to a global *lock-in* neuron whether its constraint is currently met (*OK* signals). As soon as all (or most) constraints are met the global *lock-in* neuron activates a set of additional circuit motifs which sharpen the existing energy landscape (right). This leads to a global reduction of the *temperature* of the circuit, thereby reducing exploration and forcing the network to lock into the locally best solution (Principle 4).

discrete RVs by WTA circuits is the foundation for the encoding of many problems. Specific constraints of a problem can be implemented by adding symmetric synaptic connections among principal neurons, as well as connecting additional WTA and OR circuits to different subsets of principal neurons (Figure 3.1B).

The systematic design of complex energy landscapes composed of large numbers of circuit elements calls for an understanding of how circuit elements interact with

each other. For example, what is the joint effect of two auxiliary circuit motifs which are operating on an overlapping set of principal neurons? Notably, one can show theoretically that under certain idealized conditions the energy contributions of circuit motifs sum up linearly. This occurs in particular when a) integration of synaptic inputs is linear as in (3.2) and b) the total instantaneous synaptic drive $\Delta u_{k,i}$ onto a principal neuron k due to the presence of a circuit motif C_i is given by,

$$\Delta u_{k,i}(t) = \Delta E_i(\{x_k = 0, \mathbf{x}_{\setminus k}(t)\}) - \Delta E_i(\{x_k = 1, \mathbf{x}_{\setminus k}(t)\}) \tag{3.3}$$

at any point in time during circuit operation. Since the design of large-scale circuits is greatly facilitated by linear compositionality of individual elements, (3.3) can be seen as the idealized reference functionality of a circuit motif with energy contribution $\Delta E_i(\mathbf{x})$. In practice, such reference can be used to guide circuit design; the WTA and OR circuit motifs shown in Figure 3.1B were specifically designed to approximate (3.3). A basic consequence is that the relative energy contribution of any given circuit motif is practically independent of the presence of other circuit motifs. This allows one to apply all three types of circuit motifs on different subsets of principal neurons in a combinatorial fashion to generate a rich diversity of energy landscapes in a highly controlled fashion. As a result, energy landscapes of important computational problems can be constructed in a relatively straightforward manner.

Principles 1 and 2 lay the foundation for implementing *massively parallel local search* for low energy states in complex energy landscapes through the intrinsic dynamics of spiking networks with noise.[1] *Principle 3* states that this search process is facilitated by the inherent asymmetry of spike-based signaling (Figure 3.1C). This is because asymmetric signaling, where a spike is followed by a fixed period of *on*-time whereas *off*-times are subject to random variation, alleviates one of the practical issues of stochastic local search: the presence of deep local minima in which the search process gets stuck. The benefits of spike-based signaling are particularly visible in conjunction with the WTA circuit motif: suppose that some WTA circuit represents a discrete RV in a computational problem, and each principal neuron in the WTA circuit represents one possible value of that variable, as described above. Then the WTA circuit motif permits that sometimes (randomly and briefly) none of the principal neurons in the WTA is active, and hence the value of the RV is temporarily undefined (Figure 3.1C, right). When this occurs, most of the time it has no lasting effect because the previously defined state is quickly restored. But when the transition to an undefined RV state occurs in two or more WTA circuits at approximately the same time, the principal neurons in these circuits

[1] The search is *local* in the sense that when the network moves from state $\mathbf{x}$ to some other state $\mathbf{x}'$ this always occurs through a series of small changes (a series of individual neurons turning on and off). In addition, the search process is parallel because state changes occur in a highly distributed manner across the network, thus supporting the efficient exploitation of independent substructures in the energy landscape.

are momentarily given the opportunity to reconfigure their states from scratch and explore radically different state configurations. Although such transitions to brief periods of undefined RVs occur rarely, their frequency is greatly enhanced through the asymmetry of spike-based signaling because spikes have a fixed *on* time and neurons are therefore bound to switch to an *off* state on a much more regular basis (but more briefly) than expected from the energy landscape alone. Indeed one can show theoretically that, compared with a symmetrized system which samples from the same stationary distribution $p(\mathbf{x})$[2], the stochastic dynamics of noisy spiking networks is considerably more explorative due to an increased frequency of state transitions which gap large energy barriers.

Finally, *Principle 4* proposes internal, rather than the traditional external, temperature control for regulating stochastic search as part of the spike-based computing architecture (Figure 3.1D). Temperature control, i.e. the strategic modulation of the energy landscape according to $E_T(\mathbf{x}) = E(\mathbf{x})/T$ with some temperature T, is an essential ingredient of many stochastic search algorithms (Kirkpatrick et al., 1983; Michalewicz and Fogel, 2000). High temperatures T generally lead to a flattening of the energy landscape and increased exploration, whereas low temperatures T correspond to a sharpening of the landscape and increased exploitation and drive towards (local) energy minima. We propose an internal temperature control mechanism capable of a) automatically detecting *in-situ* when an acceptable solution has been reached and b) reducing temperature once such a solution has been found, leading to decreased exploration and a quasi lock-in effect. The key advantage of such internal temperature control is that solutions are automatically detected and stabilized which facilitates readout. In particular, in the absence of stabilization the network may visit solution states arbitrarily briefly and transiently, and thus solutions may be easily missed by a sloppy readout. In the presence of a lock-in mechanism, on the other hand, good solutions are maintained and it therefore suffices to check for solutions at irregular intervals. As a practical consequence, the readout logic may run on a much slower timescale than the spiking dynamics, which may be particularly beneficial in the context of high-speed neuromorphic simulation of spiking networks.

3.3 Solving 3-SAT problems

To demonstrate these principles we applied the proposed framework to hard logical inference problems. As problem instances, hard random 3-SAT problems with a clauses-to-variables ratio 4.3 near the phase transition (Biere, 2009) are considered (Figure 3.2). Each clause (constraint) of a 3-SAT formula consists of three literals, where a literal is either a variable X_i or its negation $\overline{X_i}$ (Figure 3.2A).

[2] A symmetrized non-spike-based system which samples from the same stationary distribution $p(\mathbf{x})$ but in which transitions from *on* to *off* occur in the same stochastic manner as transitions from *off* to *on*: a continuous-time variant of Gibbs sampling.

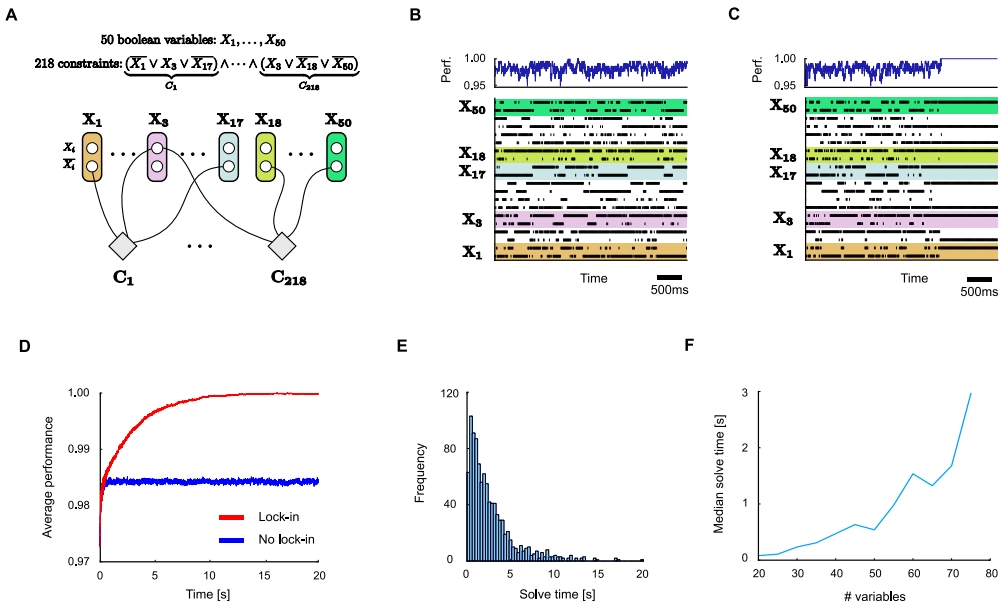

Figure 3.2: Solving logical inference (SATISFIABILITY) problems. **A**. A 3-SAT problem consisting of 50 boolean variables and 218 constraints (*clauses*). Each clause is fulfilled if at least one out of the three literals is true. For the sketched problem only one assignment to the variables exists which fulfills all clauses – the goal is to find this solution. A network implementation based on Principles 1 and 2 is shown below. **B.** Example network run during the first 4 seconds of network time (bottom: spike trains of neurons of selected WTA circuits, top: performance of current network solution over time). **C.** Example run in the presence of an additional *lock-in* circuit: when a solution is found it is automatically detected and stabilized. **D**. Average network performance over time in the absence and presence of *lock-in*. **E**. Distribution of wait times, i.e. times until the solution is found for the first time. **F.** As expected for hard NP-complete problem instances, simulation results suggest an exponential increase in wait times for hard random 3-SAT problems with increasing problem size.

A clause is fulfilled if at least one out of the three literals is true. The goal is to find an assignment to the variables which satisfies all clauses.[3] For hard problems this typically means finding one out of a handful of solutions in an astronomically large search space of $2^{\#vars}$ possible assignments. Using Principles 1 and 2, any 3-SAT problem can be encoded in a straightforward manner in a spiking network by representing each boolean variable by two neurons ($X_i/\overline{X_i}$) and a WTA circuit, and adding for each clause an OR circuit which is linked to the three literals of the clause (Figure 3.2A). Note that the network states representing correct solutions to

[3]This is the search variant of the satisfiability problem (Biere, 2009).

the problem violate the fewest circuit constraints and are therefore assigned particularly low energies (and high probability) in the energy landscape. In simulations it is observed that networks constructed in this manner quickly generate good approximate solutions to the encoded 3-SAT problem, i.e. assignments which meet many but not all constraints (Figure 3.2B). For hard problems with 50 variables a correct solution which meets all constraints is usually found for the first time after a few 100ms to a few seconds of network time (Figure 3.2B,E). Without a *lock-in* mechanism the network then continues to search for other potential solutions to the problem (Figure 3.2B). When a *lock-in* circuit is added, solutions are automatically maintained and stabilized (Figure 3.2C). As a result, the average performance of the network is considerably enhanced with *lock-in*. Regarding scalability on hard random 3-SAT problems, simulations suggest that typical wait times (the time until a solution is found by the network for the first time) scale exponentially with problem size (Figure 3.2E), as expected for hard NP-complete problem instances.

3.4 Generating solutions to Traveling Salesman Problems

We further applied the proposed framework to planning problems (Figure 3.3), in particular instances of the Traveling Salesman Problem (TSP) (Gutin and Punnen, 2002). Given a list of cities and the traveling costs for going from any city i to any other city j, the goal is to find the least costly (the "shortest") round-trip route that visits each city exactly once. The problem can be encoded in a spiking network by representing each step s in the trajectory by a WTA circuit with N_{cities} neurons, one for each city (Figure 3.3A). To encourage short routes in the energy landscape, synaptic weights between two successive steps are chosen inversely proportional to movement costs, such that low costs map onto strong excitatory synaptic connections, whereas high costs are represented by low excitatory (or inhibitory) connections. The constraint that each city must be visited only once is enforced by inhibitory connections among neurons coding for the same city at different time steps. Furthermore, to facilitate the search process, N_{resting} additional "resting" steps are introduced which allow the salesman to "rest" in a city for one time step before moving on (Figure 3.3A). Note that in the TSP optimization problem the optimality of solutions cannot be easily verified, and hence, in contrast to the 3-SAT application, the objective in this case is not necessarily to recover an *optimal* solution, but to find *good approximate* solutions. We tested the performance of the network architecture with respect to this objective on a planar 38-city problem instance ($\approx 10^{43}$ unique tours). We find that the network quickly generates good approximate solutions to the TSP problem: The average performance of generated network solutions converges within a few seconds to approximately 0.75 (where 1 corresponds to the optimal solution). Furthermore, due to fluctuations around this stationary value, performances up 0.99 are typically reached within

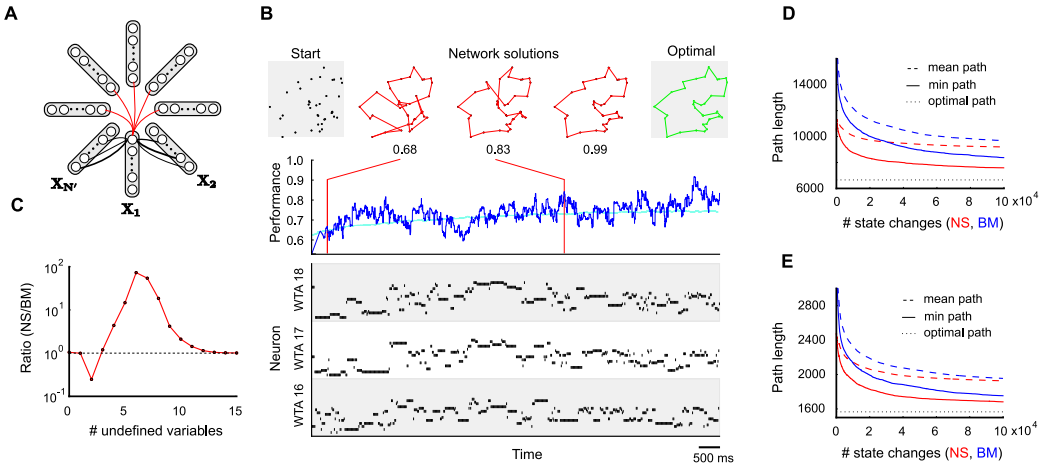

Figure 3.3: Generating approximate solutions to traveling salesman problems (TSP). **A**. Network architecture for solving TSP problems. Each step i in the trajectory of the traveling salesman is represented by a WTA circuit $\mathbf{X_i}$ with N_{cities} neurons (one for each city). The synaptic weights between two steps are chosen to reflect movement costs between each pair of cities. The constraint that each city must be visited exactly once is enforced by inhibitory connections among neurons coding for the same city at different time steps. **B**. Example application to a planar 38-city TSP instance. Top: 38-city problem; solution trajectories generated by the network; optimal solution. Bottom: spike trains of neurons in selected WTA circuits during the first few seconds of a typical run. Middle: network performance over time (dark blue: single trial, cyan: average over 100 trials). The network quickly generates good approximate solutions to the TSP problem. **C**. Advantage of asymmetric spike-based signaling (Principle 3). The number of (very brief) exploratory transitions to network states with partially undefined RVs (where no neuron is active in several WTAs i such that the value of the corresponding RVs $\mathbf{X}_i$ is undefined) is compared between the asymmetric spike-based system (NS=Neural Sampling) and an otherwise equivalent but symmetrized non-spiking system (BM=Boltzmann Machine). Intermittent transitions to exploratory states with more than 3 undefined variables $\mathbf{X}_i$ are found to occur up to 80 times more frequently in the spike-based system. **D**. Convergence is considerably faster in the spike-based system than in the symmetrized system (shown is for each system the cumulative mean/max performance as a function of the number of state changes). **E**. Similar results are found when the same analysis is repeated for an asymmetric TSP problem with 39 cities.

a few 10s. We also tested Principle 3 (the advantage of asymmetric spike-based communication) in the described setup. The results are shown in Figure 3.3C-E: Brief periods of partially undefined network states occur with increased frequency in the asymmetric spike-based architecture compared with a comparable symmetrized sampler which samples from the same stationary distribution $p(\mathbf{x})$ (Figure 3.3C). Moreover, convergence to high-performance solutions (short trajectories) is consid-

erably faster in the spike-based architecture compared to the symmetrized system (Figure 3.3D). Similar results are obtained for an asymmetric 39-city TSP problem instance (Figure 3.3E).

3.5 Discussion

In summary, we have presented a set of four new principles of circuit design with spiking neurons which enable the systematic construction of networks of spiking neurons for solving hard computational problems. In simulations, we have demonstrated this for two well-known problems, 3-SAT and the Traveling Salesman Problem. The proposed architecture for solving 3-SAT (and k-SAT) has a particularly wide range of potential applications. On the one hand, this is because every decision problem in NP can be reduced to the satisfiability problem (Cook, 1971). On the other hand, efficient solvers for satisfiability problems are in high demand in many practical applications, such as model checking and software verification (Biere, 2009) or haplotyping in genomics (Lynce and Marques-Silva, 2006). Apart from boolean k-SAT and TSP, a direct application of the proposed framework to many other important problems such as MAX-CUT, non-boolean k-SAT, the Hamiltonian path problem and graph-coloring (Karp, 1972), should be straightforward to realize and could be examined in simulations in future work. A more ambitious goal for the future is to examine to what extent the proposed framework can be realized in practice on current or future neuromorphic hardware. If successful, this would represent a major breakthrough in the pursuit of a long-standing goal: the demonstration of powerful problem solving capabilities emerging in brain-inspired computing hardware.

3.6 Acknowledgments

This chapter is based on a joint work with Zeno Jonke (ZJ) and Wolfgang Maass (WM) that was initiated after the collaboration on the recently published paper "Stochastic Computations in Cortical Microcircuit Models" described in the previous chapter. The main idea and structure of this work as well as the design of computer experiments were conceived in close collaboration of all three authors, SH, ZJ, and WM. I developed the theory of the paper. ZJ contributed the development of the software framework and conducted the computer experiments. The simulation framework was based on NEVESIM, an event based simulator for networks of spiking neurons written by Dejan Pecevski. The main text was written by SH and WM. The figures were designed by SH, ZJ and WM.

Chapter 4

Emergence of Optimal Decoding in Stochastic Winner-Take-All Networks through Spike Timing Dependent Plasticity (STDP)

Contents

The brain faces the problem to infer reliable hidden causes from large populations of noisy neurons, for example the direction of a moving object from spikes in area MT. It is known that a theoretically optimal likelihood decoding could be carried out by simple linear readout neurons if weights of synaptic connections would be set to certain values that depend on the tuning functions of sensory neurons. We show here that such theoretically optimal readout weights emerge autonomously through STDP in conjunction with lateral inhibition between readout neurons. In particular, we identify a class of optimal STDP learning rules with homeostatic plasticity, for which the autonomous emergence of optimal readouts can be explained on the basis of a rigorous learning theory. This theory shows that the considered network motif approximates Expectation Maximization for creating internal generative models for hidden causes of high-dimensional spike inputs. Notably, we find that this optimal functionality can be well approximated by a variety of STDP rules beyond those predicted by theory. Furthermore we show that this learning process is very stable, and automatically adjusts weights to changes in the number

of readout neurons, in the tuning functions of sensory neurons, and in the statistics of external stimuli.

4.1 Introduction

Uncertainty accompanies us in almost all situations in life. Whether we try to recognize a distant object, decide which path to take on a mountain hike, or read a person's face in an important negotiation: the environment often provides us with many cues, but each single cue is too unreliable to inform a decision on its own. Thus we are forced to combine different cues in a meaningful manner in order to gain sufficient certainty. The theoretical framework for solving such tasks in an optimal way is Bayesian inference. Notably, behavioral and psychophysical studies strongly support the picture that the brain implements this strategy: in numerous experiments human subjects have been shown to take into account uncertainty in a near-optimal way (Griffiths and Tenenbaum, 2006).

At the level of neural coding in early sensory areas, uncertainty is a particularly well-studied phenomenon: individual neurons which encode certain stimulus properties show significant trial-to-trial variability, making it difficult to infer the original stimulus from single neuron responses. This observation has led to the notion of population coding: the value of a single variable is encoded by a whole population of neurons, each noisy and typically broadly tuned to the external variable (Pouget et al., 2000). Experimental data suggests this coding strategy as a candidate for understanding how important variables are represented in different areas across cortex, e.g. sound location in auditory cortex (Miller and Recanzone, 2009), or stimulus location in somatosensory cortex (Petersen et al., 2002). Just how the brain reliably decodes information from populations of neurons in a near-optimal way remains one of the open key questions in computational neuroscience.

In experimental neuroscience, the computation of robust readouts from population codes has become indispensable in the analysis and interpretation of neural data: population vector analysis and maximum likelihood (ML) estimation (Pouget et al., 2000) are two frequently used methods. More recently, attempts have been made to model neural networks which exhibit near-optimal decoding capabilities (Deneve et al., 1999; Jazayeri and Movshon, 2006; Chaisanguanthum and Lisberger, 2011). The hope is that such models will advance our understanding of the neural substrates of perceptual judgments: how the noisy and broadly tuned representations found in sensory areas can be efficiently used and transformed by downstream populations to allow near-optimal performance in a variety of perceptual tasks. The theoretical framework which has been guiding the search for suitable models is Bayesian inference. Here, in contrast to previous, *static* models, we will use this perspective for the analysis of an *adaptive* cortical microcircuit, featuring spike-timing dependent plasticity.

In the Bayesian framework, an observed response of a population of M neurons,

which we will write here as $\mathbf{x} = (x_1, \ldots, x_M)^T$, is interpreted as the result of an underlying cause θ which cannot be observed directly. In visual processing this external cause could correspond to the true stimulus orientation or motion direction at some retinal location. The population code, i.e. the relation between the external cause θ and the observed population response $\mathbf{x}$, can then be represented by the conditional distribution $p(\mathbf{x}|\theta)$. This captures both deterministic dependencies on θ, usually specified in terms of tuning functions, as well as neuronal noise. The optimal way of inferring the external variable θ from noisy observations $\mathbf{x}$ is then given by Bayes' theorem:

$$\underbrace{p(\theta|\mathbf{x})}_{\text{posterior}} \propto \underbrace{p(\theta)}_{\text{prior}} \cdot \underbrace{p(\mathbf{x}|\theta)}_{\text{likelihood}} \quad . \tag{4.1}$$

In the context of population codes, the prior distribution is often assumed to be uniform, such that inference reduces to computing the *likelihood* function $L(\theta) = p(\mathbf{x}|\theta)$, which indicates how likely different θ are to have caused a given observation $\mathbf{x}$. This results in a quite straightforward decoding strategy: compute the likelihood for each possible cause θ which could have given rise to the observed population response, and choose the one with maximal likelihood. Somewhat surprisingly, the computational requirements for implementing such a readout are minimal: under a few simplifying assumptions, including Poisson firing statistics and zero noise correlations in the population pool, it was shown recently that the likelihood $L(\theta)$ of a stimulus can be written as a weighted sum of sensory responses $\mathbf{x}$ (Jazayeri and Movshon, 2006). Based on this observation, the authors argued that a readout neuron which specializes on detecting a particular stimulus value from a set of possible values, can compute the corresponding likelihood by integrating its synaptic inputs in a feed-forward manner (see the feed-forward path in Figure 4.1). The synaptic weights which are required for this operation depend on the tuning functions $f_j(\theta)$ of the sensory neurons x_j and the preferred orientation θ_k of the readout neuron:

$$w_{kj} = \log f_j(\theta_k) + \text{const.} \tag{4.2}$$

This establishes an important link between the response properties of the sensory population (the tuning functions) and the optimal weights to decode information from it, and gives clear instructions on how to construct an optimal readout network. However, (4.2) also highlights that each population of neurons in the brain must be read out differently, depending on the particular tuning functions of the neurons.

The preceding research leaves the question open how readout neurons could acquire the theoretically optimal synaptic weights (4.2). The importance of this question is underlined by recent findings which suggest that the brain constantly retunes its circuits in order to improve probabilistic inference (Bejjanki et al., 2011). How experimentally observed plasticity mechanisms at the synaptic level could account for such an improvement had remained an open question.

Here we present a learning theory for Spike-Timing Dependent Plasticity (STDP) rules in the context of spiking neurons and lateral inhibition which addresses this question: building on the analysis of previous work (Nessler et al., 2010) we show that optimal likelihood decoding of noisy population codes emerges automatically through STDP in a Winner-Take-All (WTA) circuit, a ubiquitous network motif in cortical microcircuits (Douglas and Martin, 2004). In particular, we theoretically analyze the weight dynamics of a particular form of STDP with homeostatic plasticity and show that it can be described as an attractor dynamics, where the centers of the attractors are the weight values (4.2) that are optimal from the perspective of probabilistic inference and learning. In this way we create a direct link between simple local rules for synaptic plasticity and theoretically optimal inference and learning on the network level. Whereas the analysis of Nessler et al. (2010) was restricted to the case of multinomial input variables (that are each encoded by a population of neurons of which at any time t exactly one has fired within the time window $[t-\tau, t]$ and rectangular EPSPs (modeled by a step function, rather than by a function with smooth decay), we show here that the underlying learning theory can be extended to cover the biologically more realistic case where each input neuron fires according to some Poisson-like statistics, and EPSPs have a smooth decay. This new learning theory makes it possible to derive learning curves for STDP and dependencies between current weight values and the amount of weight potentiations or depressions under STDP that are optimal for a given input statistic and EPSP shape. These analytically derived predictions for details of STDP match currently available data quite well in a number of aspects. Furthermore, we show that in practice variants of these optimal rules, and in particular plasticity rules based on typical STDP curves, approximate this optimal behavior well.

We test predictions of the new learning theory in computer simulations and show that previously derived optimal weights for likelihood decoding (4.2) emerge autonomously through STDP in conjunction with homeostatic plasticity. Based on this learning theory, we also show that an adaptive architecture for reading population codes provides attractive benefits compared to previously considered static models. In particular, we demonstrate stable and predictable behavior under challenging but biologically realistic dynamic scenarios, like changes in sensory tuning functions or neuron growth and loss. We furthermore demonstrate that selective modulation of learning leads to effects which are reminiscent of perceptual learning (Gilbert et al., 2009), where the accuracy of neural codes is selectively enhanced for behaviorally relevant stimuli.

The article is structured as follows: first, we introduce the canonical microcircuit model we consider. In this circuit model we demonstrate STDP-based autonomous learning upon exposure to population coded stimuli, and show that in such a setup the synaptic weights converge to a setting which is optimal from the perspective of likelihood decoding. We then introduce the theory which underlies this optimality,

and derive a rigorous link between local synaptic learning and theoretical optimality in the Bayesian framework. We complement these theoretical results by an extensive performance comparison of different optimal and near-optimal STDP-based learning rules. Hereupon we present results of simulations which test further predictions of the theory, and demonstrate the versatility and robustness of the considered model. We conclude with a simulation in which effects reminiscent of perceptual learning are reproduced.

In order to make accurate perceptual judgments, the brain must use the information provided by sensory areas as efficiently as possible. Since sensory neurons tend to be noisy and broadly tuned, the computation of a sparse representation of the most likely external stimulus ("hidden cause") is a nontrivial task for a network of neurons. However it was shown in Jazayeri and Movshon (2006), that for the case of a one-dimensional hidden cause θ, which could for example represent the current orientation, speed, or direction of a visual stimulus, this task could in principle be solved by an array of linear readout neurons $k = 1 \dots N$ with spiking outputs z_k, that receive synaptic inputs x_j from sensory neurons $j = 1 \dots M$ (see Figure 4.1). It was shown that the readout neurons k can compute in their membrane potential the log-likelihood that a particular hidden cause θ_k (which was assigned externally to readout neuron k in Jazayeri and Movshon (2006)) had caused the current spike output $\mathbf{x}$ of the sensory neurons. This occurs if the weights w_{kj} from sensory neuron j to readout neuron k are set according to (4.2). We show here, that these weights emerge as fixed points (equilibria) of a class of theoretically optimal STDP rules with homeostatic plasticity, provided that the readout neurons are subject to lateral inhibition. Through the same learning process each readout neuron k implicitly develops a preferred stimulus ψ_k, which then altogether allows the reconstruction of an external input variable $\theta(t)$ at any moment in time.

4.2 Adaptive, stochastic WTA architecture

For the readout neurons k a stochastic neuron model is used similar to the model recently proposed by Jolivet et al. (2006), which has been shown to explain neural data well. The neuron model is characterized by an exponential dependence of the firing probability on the current membrane potential u_k,

$$p(k \text{ fires in } [t, t + \Delta t]) = \Delta t \exp(u_k(t)) \ , \tag{4.3}$$

for small Δt. The membrane potential of a readout neuron k consists of an excitatory and an inhibitory contribution:

$$u_k(t) = \sum_{j=1}^{M} w_{kj}(t) x_j(t) - I(t) \ . \tag{4.4}$$

Excitation comes from the feed-forward connections originating in the sensory population $x_1, \dots, x_M$. In particular, the term $w_{kj}(t)x_j(t)$ represents the contribution

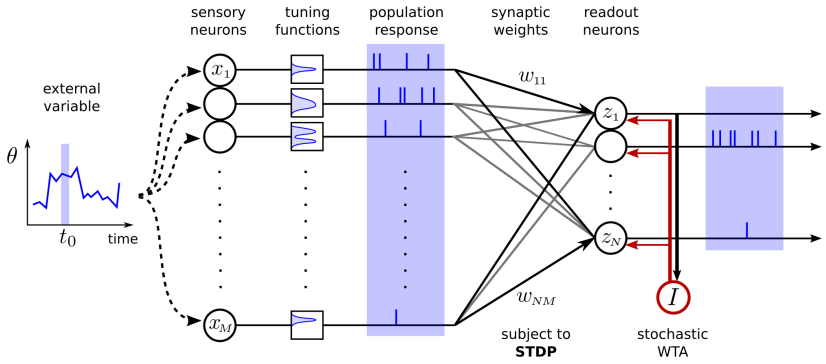

Figure 4.1: Network motif in which STDP generates optimal decoding of population codes. A time-varying external variable $\theta(t)$ is encoded by a population of sensory neurons. Each sensory neuron x_j has a characteristic average response for each value of θ: its tuning function. Based on these tuning functions they emit spikes according to a Poisson process. Readout neurons receive these spikes via feedforward connections with synaptic weights w_{kj}. A stochastic WTA circuit induces competition among readout neurons k. As a consequence, those readout neurons, which receive the greatest stimulation from the sensory neurons, fire preferentially.

of previous spikes from the j'th sensory neuron to the membrane potential u_k of the readout neuron k at time t. The term $x_j(t)$ hence models the *unweighted* output spike train of the j'th sensory neuron after filtering according to the low-pass filtering properties of the post-synaptic membrane. $\mathbf{x}(t)$ represents the collection of all input signals which are available to the readout neurons at time t. We will refer to this as the current (sensory) population response.

$I(t)$ denotes the contribution to the membrane potential due to lateral inhibition. $I(t)$ is common to all readout neurons in the circuit, and can thus be viewed as a global inhibitory signal which controls the total gain of the circuit. In computer simulations we modeled $I(t)$ such that the total firing activity of the readout circuit remained approximately constant. The resulting effect of inhibition is a normalization of circuit responses, reminiscent of normalization models in cortex (Simoncelli and Heeger, 1998; Zoccolan et al., 2005; Ohshiro et al., 2011; Louie et al., 2011). Note that the lateral inhibition introduces competition among the readout neurons, since a readout neuron with strong feed-forward input will claim a large fraction of the total firing rate, thereby suppressing other readout neurons. We will refer to the resulting network as a (stochastic) Winner-Take-All (WTA) circuit.

We focus in this article on the following theoretically motivated spike-timing-dependent plasticity rule (but see Figure 4.3 for other forms of STDP we consider),

$$\Delta w_{kj} = \eta \cdot z_k \cdot (x_j \cdot \alpha e^{-w_{kj}} - 1) \ , \tag{4.5}$$

where α is a positive constant which controls the balance between potentiation and depression, and η is a learning rate which is constant unless otherwise stated.

We define $z_k = 1$ at spike times of the WTA neuron k and $z_k = 0$ otherwise. The learning rate η is chosen small throughout this article such that learning takes place on a (much) longer time-scale than stimulus and network dynamics.

The first term of (4.5) corresponds to a prototypical spike-timing dependent long term potentiation (LTP) window (with the same shape and time constant as PSPs) with weight dependence. The second term depends on post-synaptic spikes only, and can be interpreted as a form of homeostatic plasticity providing negative feedback to the post-synaptic rate. Both terms fit well into the relatively broad phenomenological framework of STDP rules (Gerstner and Kistler, 2002; Gilson et al., 2010). But to distinguish it from more common STDP rules (which typically feature spike-timing dependent depression), we will refer to this rule as "theoretically optimal STDP".

The motivation for first considering this form of STDP comes from its notable and provable theoretical properties which will be developed in this article: (4.5) leads to stable equilibrium weight settings which can be analyzed, and have a clear interpretation from the perspective of probability theory. In particular, we will show that (4.5) is an instance of a family of learning rules which can be directly derived from the principle of adapting and optimizing an implicit generative model of the input statistics.

With regard to biological plausibility, (4.5) is in many aspects consistent with experimental studies on STDP. First, the strength and direction of learning depends on the timing difference between pre- and post-synaptic spike. For pre-before-post pairings, x_j is large at the time of the post-synaptic spike and (4.5) will typically lead to potentiation. For post-before-pre pairings, the negative part dominates and leads to depression (Bi and Poo, 1998; Sjöström et al., 2001). Also, the strength of potentiation correlates inversely with the synaptic weight before pairing. This feature is also consistent with a number of experimental studies (Bi and Poo, 1998; Sjöström et al., 2001; Liao et al., 1992; Montgomery et al., 2001). Furthermore, the amount of depression is independent of the current weight. This is consistent with the experimental results of Jacob et al. (2007), which is to the best of our knowledge the only study of this effect in-vivo. Moreover, when measured under a typical pairing STDP protocol, (4.5) will automatically shift towards LTP for higher pairing frequencies, since for each post-synaptic spike, the x_j will accumulate more pre-synaptic spikes in the causal STDP window, while the negative term remains constant. This effect is hence reminiscent of the tendency towards LTP (and abolishment of LTD) for higher frequencies found experimentally (Sjöström et al., 2001). Altogether, the agreement with these experimental STDP data is in a sense remarkable, given that (4.5) can be derived from purely statistical principles.

However, there are also a few deviations from STDP data. Most importantly, the negative contribution of (4.5) is activated for every post-synaptic spike regardless of pre-synaptic input. Hence, long-term depression (LTD) is not spiking-timing dependent and can be triggered by post-synaptic spikes alone. We address this po-

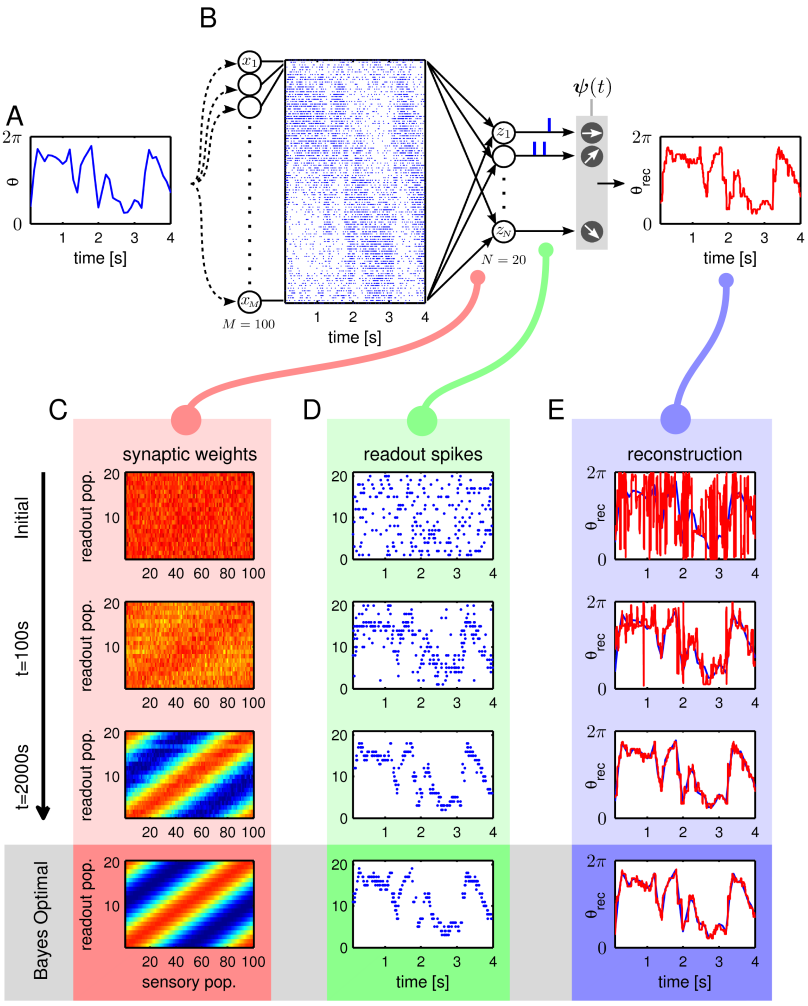

Figure 4.2: (see next page for Figure caption)

tential issue later, when we consider also a variation of (4.5) with timing-dependent depression in Figure 4.3, and it turns out that the results are qualitatively very similar. Another deviation is in regard to the frequency dependence mentioned above: despite the general shift towards LTP, the exact form of frequency dependence in experimental data (see e.g. Figure 7 in Sjöström et al. (2001)) cannot be reproduced by (4.5).

For the described network architecture a computer simulation was carried out, in which a fluctuating external variable $\theta \in [0, 2\pi]$ with a periodic boundary condition (like any angle) was presented (Figure 4.2). Concretely one could interpret this external variable as a motion direction. The changes in θ were designed to be slightly slower than the network dynamics, corresponding to the assumption that

Figure 4.2: Spike-timing dependent plasticity regulates synaptic weights towards theoretical optima for likelihood decoding. **A**. An external variable $\theta(t)$ was presented to the network of Figure 4.1A, here shown over a period of $6s$. **B**. 100 sensory neurons $(x_1, \ldots x_M)$ with broad tuning functions and Poisson firing statistics provided a noisy population code of this stimulus. Through plastic synaptic readout weights, these sensory spikes were fed into a readout network (stochastic WTA) consisting of 20 spiking neurons with lateral inhibition. The readout network produces a sparse output code. An external observer can associate each readout neuron with a preferred stimulus $\psi_k(t)$, which develops over the course of learning. Based on the preferred stimuli, the original stimulus can be reconstructed from the sparse readout spikes. **C**. The development of the weight matrix through theoretically optimal STDP is shown over 2000s during exposure to a randomly changing stimulus $\theta(t)$. Colors range from dark blue (0) to dark red (2.4). Initially, the weight matrix was randomly initialized. At $t = 100s$, the effects of learning became visible: sensory neurons which typically fired together had developed a preference for activating the same readout neuron. At $t = 2000s$ the weights had stabilized, and settled on a configuration which closely matches the theoretical Bayes-optimal solution (4.2). Note that the readout neurons in panel C (and D) are ordered by their preferred stimuli ψ_k at $t = 2000s$, to facilitate visual comparison with the optimal solution. **D**. The development of readout responses is shown over the course of learning. Initially, responses were unspecific to different stimuli. At $t = 100s$, readout neurons had developed noticeable preferences for certain stimuli. At $t = 2000s$, each readout neuron had become responsible for representing a particular preferred stimulus. The firing activity of a readout neuron reflects the posterior probability of the neuron's preferred stimulus, given the current population response. The characteristic sparse code which emerged is predicted by theory (see Bayes-optimal, bottom). **E**. The readout spikes were used to compute a reconstruction $\theta_{rec}(t)$ (shown in red) of the external stimulus $\theta(t)$ (shown in blue), in order to compare the quality of the readout at different stages of learning. The reconstruction signal at time t was obtained based on the preferred stimuli $\boldsymbol{\psi}(t)$ of those readout neurons which spiked in a 100ms time window before t (see main text). As shown, initially each readout spike conveyed rather unspecific information about the stimulus. The following specialization of readout neurons to different stimuli allowed for an increasingly accurate reconstruction.

the sensory stimulus remains stable for the duration of a few PSPs. A number of broadly tuned sensory neurons with Poisson firing statistics represented the value of this external variable over time. Initially, synaptic readout weights were randomly initialized, leading to rather unspecific responses in the readout circuit when stimulated with different θ. At $t = 100s$ of exposure to different population responses, readout neurons started to specialize on different stimuli. This was reflected both by selective firing of individual readout neurons to certain θ, and the gradual specialization of synaptic readout weights to different "patterns". This specialization can be understood as follows. During exposure to population responses, the readout network continuously emits spikes. After each readout spike, the current population response leaves a small trace in the synaptic weights of the readout neuron k which has fired. In particular, as expected from STDP, synapses which are not active at the time of the post-synaptic spike are depressed, and those with strong activity

are enhanced. The synaptic changes induced by plasticity increase the probability that the same neuron fires again when a similar population response is registered, while making it less likely to fire when significantly different patterns are active. Due to the inhibitory circuit which promotes competition, different neurons start to pick up different patterns. As a consequence, a specific pattern emerges in the synaptic weights of each readout neuron. This pattern reflects the history of population responses which made the readout neuron fire. At $t = 2000s$ the readout weights have reached a setting which was stable with minor fluctuations around some target values. One can now compare this converged weight setting with the results by Jazayeri and Movshon (2006) on optimal population decoding. As shown in Figure 4.2C, the weights, which have autonomously emerged during exposure to unknown population responses, have recovered the previously identified theoretical optima for likelihood decoding, in particular those obtained through (4.2).

What is the overall effect of this learning process? Most importantly, individual readout neurons have become specialists for individual hidden causes (or stimuli). Hence, after but not before learning, each readout spike can be considered an indication for a particular hidden cause. As a consequence one can achieve an increasingly faithful reconstruction of the original stimulus θ from readout spikes, as shown in Figure 4.2E.

4.3 Learning theory for STDP in a network with lateral inhibition

We have shown in Figure 4.2 that the spike-timing dependent rule (4.5) in a network with lateral inhibition can regulate the synaptic readout weights such that near-optimal decoding of population responses can take place. We demonstrated that a main feature of this learning process was the emergence of a sparse spike code, optimized to reveal the hidden causes of high-dimensional input spike-trains. Here we will show that this behavior can be understood in terms of a rigorous learning theory. Our theoretical analysis relies on the notion of an implicit *generative model* implemented in the network, a powerful statistical tool for the extraction of hidden causes from high-dimensional data. Using basic assumptions about the input distribution we arrive at a performance measure favoring sparse spiking in which different readout neurons fire for distinct hidden causes (with clearly distinct sensory representations). This theoretical tool allows us to track and analyze the performance of the network, in particular small changes in performance due to a single application of the learning rule. The main result is that an application of the local learning rule (4.5) is always expected to increase global performance. This increase can be understood in terms of an attractor dynamics in the weight space induced by the plasticity rule, in which the attractor centers are weight settings which are stable

under the dynamics of the network. These equilibria are characterized by,

$$w_{kj} = \log \langle x_j | k \text{ fires} \rangle + \text{const.}, \tag{4.6}$$

where we write $\langle \cdot | k \text{ fires} \rangle$ for the empirical average over all spike times of neuron k during the operation of the circuit (for a more precise definition, see Appendix D: Equilibria of STDP). Importantly, we show that these weight settings are not only the attractors of the network dynamics, but they also correspond to locally optimal weights settings from the perspective of the implicit generative model. Hence, we establish a direct link between local synaptic learning rules and theoretically optimal performance of the network.

A common problem in data analysis is to extract hidden causes from high-dimensional input $\mathbf{x}$ without supervision, i.e. without the help of a teacher signal which provides the desired outcome $\mathbf{z}$ during training. Generative models are arguably the most powerful paradigm of unsupervised learning, with examples including (probabilistic) PCA , ICA, Gaussian mixture models and Hidden Markov Models (Bishop, 2006). The rationale behind generative models is "analysis by synthesis": if you aim to understand (decode) something, learn to build (generate) it yourself. The starting point is a model $p(\mathbf{x}|\mathbf{W})$ describing the statistics of an input stream $\mathbf{x}$ as the result of a generative random process, typically involving hidden causes $\mathbf{z}$,

$$p(\mathbf{x}|\mathbf{W}) = \sum_z p(\mathbf{z}|\mathbf{W}) \cdot p(\mathbf{x}|\mathbf{z}, \mathbf{W}) \quad . \tag{4.7}$$

This generative model can understood as a two-step process. First, a hidden configuration is drawn according to the hidden probabilities $p(\mathbf{z}|\mathbf{W})$. Then, the hidden states generate the actual data $\mathbf{x}$, according to $p(\mathbf{x}|\mathbf{z}, \mathbf{W})$. The sum in (4.7) reflects the fact that the same data $\mathbf{x}$ can be generated by different hidden configurations $\mathbf{z}$.

Then, the goal of learning is to find parameters $\mathbf{W}$ which bring the distribution $p(\mathbf{x}|\mathbf{W})$ generated by the model as close as possible to the actual data distribution, which we will write here as $p^*(\mathbf{x})$. This is done by adjusting the parameters $\mathbf{W}$ of the model, usually in small steps, until the *Kullback-Leibler divergence* $\mathrm{KL}(p^*(\mathbf{x})||p(\mathbf{x}|\mathbf{W}))$, a quantity that measures the distance between the real distribution and the model, becomes minimal. During this process, an efficient representation for the hidden causes $\mathbf{z}$ emerges which is optimized to "explain" the data in $p^*(\mathbf{x})$. This hidden representation can be retrieved by evaluating the posterior distribution $p(\mathbf{z}|\mathbf{x}, \mathbf{W})$.

Here, we adopt the perspective of a generative model to analyze learning for the network shown in Figure 4.1. But what is a good choice for a generative model? First, we restrict ourselves to a sparse representation of readout neurons, meaning that only a few hidden causes should be sufficient to explain any given input. A simple way to achieve this is a mixture model, in which at any time exactly one

hidden cause (or expert) is active, and each expert is associated with a different set of input patterns. Second, since inputs to the neural network consist of spike trains with Poisson characteristics, a reasonable choice is to assign to each expert a characteristic pattern of input firing rates to which it is specialized. Formally, the generative model then takes the form,

$$p(\mathbf{x}|\mathbf{W}) = \frac{1}{N}\sum_{k=1}^{N}\prod_{j=1}^{M}\text{Poisson}(x_j;\alpha^{-1}e^{w_{kj}}) \ , \tag{4.8}$$

where $\text{Poisson}(x;\lambda)$ denotes the Poisson distribution over x with "rate" λ, and α is an arbitrary positive constant (note that our analysis extends beyond Poisson distributions, see below). The model can be understood as follows: the probability of cause k being active is $\frac{1}{N}$. If the cause k is active, the number of spikes for the sensory neuron x_j within a rectangular PSP window of length τ is Poisson-distributed with a rate which is encoded in the parameter w_{kj}. Hence, each cause k is an expert for a particular pattern of Poisson rates in the input, and this preferred pattern is encoded in the parameters $w_{k1},\ldots,w_{kM}$. Note that prior parameters instead of a fixed prior $\frac{1}{N}$ could be easily incorporated into the model. These prior parameters could then be mapped onto neural excitabilities in the neural implementation (Nessler et al., 2010).

Several steps are required in order to establish a direct correspondence between the network implementation and the generative model: first, the network elements need to be related to the variables of the generative model. We already implicitly linked the synaptic weights to the model parameters in a straightforward fashion by using the weights w_{kj} in the definition of the generative model. Similarly, we can link the readout neurons to the hidden causes of the model: each hidden cause k is represented by one readout neuron, k. Second the network should support Bayesian inference, i.e. it should respond to an input $\mathbf{x}$ by inferring likely hidden states $\mathbf{z}$, according to the posterior probabilities $p(\mathbf{z}|\mathbf{x},\mathbf{W})$. Indeed we will show that each output spike from the readout network is generated according to the correct posterior distribution. Third, network plasticity should optimize the parameters of the generative model over time, i.e. the synaptic weights should come to reflect the input statistics $p^*(\mathbf{x})$ through learning. This is the main focus of this article. We will prove this by relating spike-timing dependent learning to a standard algorithm for generative model learning, Expectation Maximization (EM). One operation of the generative model which will not be required from the network, is the actual generation of data $\mathbf{x}$. Since, in contrast to other models (see e.g. Dayan et al. (1995); Rao and Ballard (1999)), such an explicit generation of data is not needed here, we will refer to our model as an *implicit* generative model.

In the spirit of Jazayeri and Movshon (2006) and Nessler et al. (2010), Bayesian inference in the model defined by (4.8) can be related to a feed-forward neural implementation with synaptic weights w_{kj} between input and readout layer. This

is because the log-likelihood of an input $\mathbf{x}$ under the cause k is given by,

$$\log p(\mathbf{x}|k,\mathbf{W}) = \sum_{j=1}^{M} w_{kj}x_j - \sum_{j=1}^{M} \alpha^{-1}e^{w_{kj}} - \sum_{j=1}^{M} \log(x_j!) \ . \tag{4.9}$$

The second term can be ignored in practice if the input representation is homogeneous (Jazayeri and Movshon, 2006). The third term is independent of k, and hence drops out later in the normalization of the posterior distribution $p(k|\mathbf{x},\mathbf{W})$. This leaves only the first term which is simply the feed-forward sum of inputs, which a neuron k can compute in its soma. As a consequence, we can show that each spike from a readout neuron in Figure 4.1 can be interpreted as a sample from the correct posterior distribution $p(k|\mathbf{x},\mathbf{W})$ (see Appendix D).

The key component of the present learning theory is the connection between spike-timing dependent learning and the implicit generative model, in particular the optimization of its parameters $\mathbf{W}$. In machine learning, the most well-known algorithm for performing this optimization is Expectation Maximization (EM) (Bishop, 2006). What we will show here is that the operation of spike-timing dependent learning in the cortical microcircuit of Figure 4.1 can be understood as a stochastic online version of EM. This allows us to view learning as an attractor dynamics in the weight space, where the attractors correspond to local optima in the generative model perspective.

The framework of Expectation Maximization provides a general tool for deriving and analyzing learning dynamics of autonomously learning systems which fit an implicit or explicit generative model to an external distribution $p^*(\mathbf{x})$. Of particular interest is the family of online EM algorithms (Sato, 1999; Cappé and Moulines, 2009), which operate at one input at a time, thereby gradually absorbing the input statistics into the parameters of the model. The fundamental steps of online EM are inference (Expectation) and learning (Maximization). During inference, the posterior probabilities $p(\mathbf{z}|\mathbf{x},\mathbf{W})$ of hidden configurations are evaluated for the current input $\mathbf{x}$. The subsequent learning step uses both $\mathbf{x}$ and the inferred hidden configurations to perform a small update $\Delta\mathbf{W}$ which increases the model's likelihood for the current input $\mathbf{x}$. As indicated above, each firing of a neuron k in the WTA-circuit of the network provides a sample from the current posterior distribution: hence this is equivalent to a stochastic E-step. The corresponding post-synaptic spike then triggers the plasticity rule (4.5) in the synapses $w_{k1},\ldots,w_{kM}$. Through an analysis of this STDP-based update one can show that this provides a step in the direction of a correct M-step in online EM. In particular, one can prove that the expected application of the rule with a sufficiently small learning rate will always improve the performance of the network until a local optimum is reached, i.e.

$$\mathrm{KL}(p^*(\mathbf{x})||p(\mathbf{x}|\mathbf{W}+\overline{\Delta\mathbf{W}})) \leq \mathrm{KL}(p^*(\mathbf{x})||p(\mathbf{x}|\mathbf{W})) \ , \tag{4.10}$$

where $\overline{\Delta\mathbf{W}}$ is the expected update for a randomly chosen input pattern from $p^*(\mathbf{x})$ (see Appendix D).

This allows us to view learning in the considered cortical microcircuit as an attractor dynamics: the trajectories in weight space induced by spike-timing dependent learning are attracted to specially distinguished weight settings which have the property that for all synapses $\overline{\Delta w_{kj}} = 0$, i.e. the system is in equilibrium with respect to (4.5). These attractors in weight space are optimal from the perspective of the implicit generative model, in the sense that they correspond to locally optimal solutions to the problem of fitting the internal model $p(\mathbf{x}|\mathbf{W})$ to the input distribution $p^*(\mathbf{x})$.

Before moving on to the generalization of this main result to non-Poisson input statistics, a remark is in order concerning the connection between the optimization of the implicit generative model $p(\mathbf{x}|\mathbf{W})$ and the optimal readout weights for population coding, as they were derived in Jazayeri and Movshon (2006). As already suggested by the simulation results shown in Figure 4.2, these two optimality criteria are intimately connected. This is because the "true" statistics of the population code $p(\mathbf{x}|\theta)$ from which the optimal readout weights in Jazayeri and Movshon (2006) are derived, can be mapped with arbitrary precision onto a mixture model representation in the form of (4.8). The mapping is most faithful if the Kullback-Leibler divergence between the real population code and the mixture model is lowest: this is precisely what the plasticity rule optimizes. By increasing the number of readout neurons the theoretically achievable divergence can be made arbitrarily small, and the two optimality criteria become effectively equivalent.

The assumption of Poisson variability is a reasonable and popular approximation to the firing statistics of real cortical neurons. At the same time there exists ample experimental evidence for deviations from this rule, e.g. bursty or regularly firing neurons (Shinomoto et al., 2009). This raises the question whether our learning theory can be applied also to other firing statistics. In turns out that the presented theory can indeed be generalized to the exponential family of distributions, which contains many well-known parametric models like the Poisson, Normal, Gamma, and negative binomial distributions. The main result of this generalization is that each firing statistics is associated with a different, small variation of optimal STDP. Through the generalization to exponential families, the presented theory for STDP in the context of lateral inhibition becomes applicable to a wide range of biologically relevant firing statistics (see Appendix D). The power of exponential family distributions also allows to incorporate more realistic EPSP shapes: given the firing statistics of sensory neurons, together with the EPSP shape, one can construct a "tailor-made" exponential family distribution which accounts for the variability encountered in the inputs $\mathbf{x}$ (see Appendix D). This makes the theory applicable to virtually arbitrary EPSP shapes, in particular those derived from electrophysiological experiments.

4.4 A family of STDP rules leads to near-optimal readouts

These theoretical results are quite encouraging, since they establish a link between a particular form of STDP on the one hand, and the powerful statistical framework of Expectation Maximization on the other hand. But how relevant are these results for the study of autonomous learning processes in the cortex? After all, the theoretically postulated mechanism for depression in (4.5), which leads to LTD even in the absence of pre-synaptic spikes, is quite speculative. It would thus be highly interesting to know whether this term could also be replaced by timing-dependent depression without loss of functionality. Furthermore, plasticity mechanisms in the brain appear to be heterogeneous and noisy (Sjöström et al., 2001; Caporale and Dan, 2008), in contrast to precise theoretical rules. If one assumes a given PSP shape and Poisson variability, is (4.5) really the only rule which can be guaranteed to lead to the emergence of optimal weights?

In this section we provide further results to address these questions. First, the convergence results (and the link to EM) which hold for (4.5) can in fact be easily generalized to a whole family of optimal STDP rules with homeostatic plasticity,

$$\Delta w_{kj} = \eta \cdot z_k \cdot \left(f(w_{kj}) \cdot x_j \cdot \alpha e^{-w_{kj}} - f(w_{kj}) \right) \quad , \tag{4.11}$$

where $f(w_{kj})$ can be any strictly positive function. For $f(w_{kj}) = 1$, the original learning rule is recovered, which yields LTD proportional to the post-synaptic rate $\langle z_k \rangle$. This form of LTD is thus reminiscent of synaptic scaling (Turrigiano, 2010), except that in synaptic scaling, updates scale with the current weight (Abbott and Nelson, 2000). Indeed, a learning rule with synaptic scaling can be obtained by letting $f(w_{kj}) = w_{kj}$. We verified the correctness of this (optimal) variant of (4.5) in Figure 4.3B.

Furthermore, we tested two variations which are not provably optimal from the perspective of the presented learning theory, but approximate the functionality of the optimal rules well: As shown in Figure 4.3C, in another variation of (4.5) the time constant of the causal STDP window was chosen twice as long as the PSP decay constant (while its magnitude was halved). The resulting performance is virtually optimal. Finally, we tested the performance of an STDP rule with common timing-dependent depression (Figure 4.3D), with additional superimposed noise. The resulting weights are slightly less pronounced and more noisy, but the readout performance is comparable to the other variants (compare RMSE values in the bottom row). Notably, this behavior is also quite robust against variations of parameters of this STDP rule (see Figure D.1).

Although these variants only cover a few cases of special interest, they make clear that the main finding of this article, the emergence of optimal readouts, is not an artifact of a specific definition of the learning rule, but in practice a property of a whole family of STDP learning rules. The most important features which appear

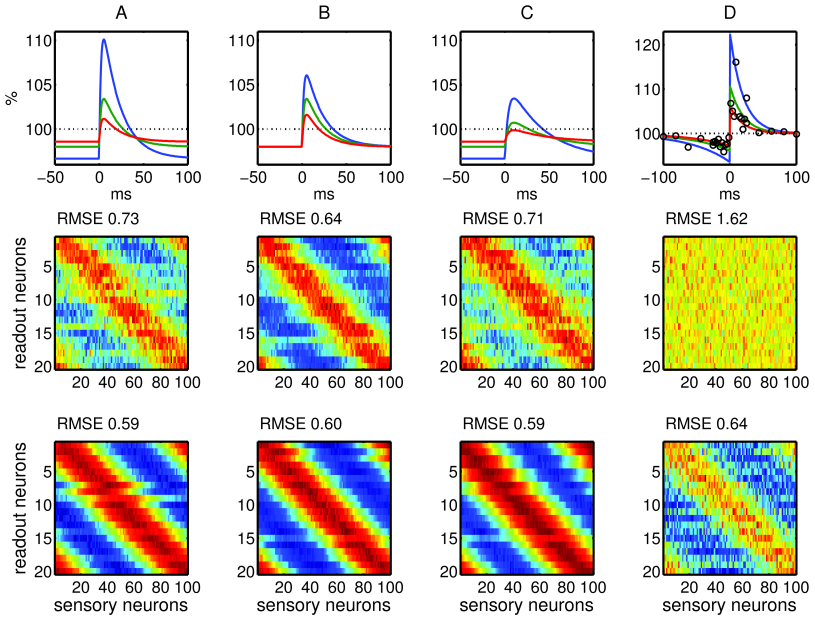

Figure 4.3: A variety of STDP rules leads to (near-) optimal readouts. Top row: Illustration of the four STDP rules which were tested. To illustrate weight dependence, the STDP curves are shown for different weights (blue: 0.5, green: 1, red: 1.5). **A** and **B**. Theoretically optimal rules: (4.5) and a variant with synaptic scaling, respectively. **C**. A modification of (4.5) with the time-constant of the causal STDP branch doubled. **D**. STDP with common timing-dependent depression and superimposed noise. Black dots in top row represent noisy samples (weights for these samples drawn randomly from a normal distribution with mean 1 and standard deviation 0.4). Middle row: synaptic weights after 500s of learning. As a performance measure, the reconstruction RMSE is shown (root mean squared error of reconstruction signal in a 2000s test run). Bottom row: weights after 3000s. The RMSE of the optimal decoder is 0.58. All variants yield near-optimal performance.

to characterize this family of STDP rules are a) pronounced weight dependence of spike-timing dependent potentiation (in accordance with experimental data) and b) some of form of homeostatic regulation of weights, either implemented explicitly (e.g. through synaptic scaling) or implicitly through spike-timing dependent depression.

4.5 Maintenance of optimality in spite of drastic network changes

To our knowledge, this article provides the first "adaptive" neural approach to optimal decoding from sensory populations in the literature. As a consequence, the canonical microcircuit considered here is substantially more flexible than previous, static models for optimal decoding. In the following computer simulations, we have

selected two biologically motivated dynamic scenarios which demonstrate its functional advantages. The first scenario is based on the observation that plasticity in sensory cortex appears to persist throughout adulthood, as suggested by numerous studies in different species and cortical areas (Trachtenberg et al., 2002; Goel and Lee, 2007). Hence, the tuning functions of sensory neurons, especially in superficial layers, are not fixed but subject to constant change. This constitutes a challenge to downstream populations which rely on the representation provided by sensory areas. We will show that the canonical microcircuit considered here gracefully handles a scenario of changing sensory tuning functions: any change in the input distribution is automatically detected by the network, resulting in an appropriate adjustment of the weight settings. As a consequence of this constant readaptation, the readout representation of the external stimulus can remain remarkably invariant to changes in the sensory representation. The second scenario deals with neurogenesis and neuron death, two phenomena which have been reported repeatedly in adult cortex (Morrison and Hof, 2007). We studied the effects of neuron growth and death in the readout population, showing that through learning, the readout network can a) minimize the detrimental effects of cell death and b) maximize the gain in representation accuracy brought by neurogenesis.

In the first scenario, we studied the consequences of changing tuning functions in the sensory population on the readout network (see Figure 4.4). For this we divided the sensory population into three groups (G1-G3). Initial sensory tuning functions were chosen such that the whole range of the input stimulus was represented uniformly in each group. The simulation was then split into three phases: a static phase (0s-1000s), a dynamic phase (1000s - 2000s) and a consolidation phase (2000s - 3000s). During the static phase, sensory tuning functions were kept constant and the readout network learned an optimal readout representation based on the initial sensory code. The resulting weight matrices are qualitatively identical to those obtained in Figure 4.2C. In the dynamic phase, tuning functions in all three groups of sensory neurons started to change. In the first group tuning functions broadened, in the second group they narrowed, and in the third group tuning functions narrowed and simultaneously started to develop a second mode (see Figure 4.4B). This change in the sensory population code was automatically "detected" by the readout network causing the synaptic readout weights to take up pursuit of a quickly moving target: the attractor centers of the learning dynamics which depend on and thus drift in parallel with sensory tuning functions. A snapshot of the readout weight matrix at time $1600s$ in the middle of the dynamic phase is shown in Figure 4.4C. In the consolidation phase, sensory tuning functions were fixed again. This allowed the readout network to converge to the new optimal setting, thereby reinterpreting all sensory neurons according to their new properties. This is particularly visible in Group 3 (G3), where each sensory neuron had developed two modes in the tuning function. The corresponding optimal decoding strategy is to connect each sensory neuron in G3 strongly with those

readout neurons which correspond to the two modes. As shown in Figure 4.4C, this strategy was found autonomously through learning. Figure 4.4D illustrates the preferred stimuli of five exemplary readout neurons during the dynamic phase. One should note that, in spite of drastic changes in the sensory representation (e.g. the development of bimodal tuning functions, see Figure 4.4B), the preferred stimuli of readout neurons hardly change during the adaptation period. Hence, learning facilitates the decoupling between sensory and readout representations with respect to the external stimulus.

In a second scenario (see Figure 4.5) we studied the effect of neuron growth and death in the readout population. In a computer simulation, a microcircuit was setup with five readout neurons, and the weights were allowed to converge to a stable setting upon stimulation with input stimuli analogous to Figure 4.2. At $t = 800s$ and $t = 1600s$ the growth of 5 new readout neurons with randomly initialized weights was simulated yielding a jump in the mean squared error (MSE) of the reconstruction, see Figure 4.5B. In theory, a larger number of readout neurons allows for a more fine-grained representation, and hence for a more accurate readout. However, this requires that the weights of all neurons are adjusted appropriately. In particular the newly formed neurons should learn to respond to the most poorly represented regions of the external stimulus. Figure 4.5A shows that this optimal strategy automatically emerges through learning: the preferred stimuli ψ_k of the newly grown neurons quickly learned to fill in the "spaces" in between the existing neurons. This is also reflected by a lower mean squared error of the reconstruction signal (Figure 4.5B, compare at $t = 800s$ and $t = 1600s$). A similar effect is observed after simulating the new growth of another five neurons with random weight initialization. At $t = 2400s$ the sudden death of 10 neurons was simulated. Theoretically, this substantially reduces the achievable readout accuracy. However, the loss in accuracy is smallest if the surviving neurons rearrange such that their preferred stimuli ψ_k are uniformly distributed over the input range. As shown in Figure 4.5A, this is the strategy which is automatically implemented by learning in the network ($2400s$-$3200s$). Hence, the detrimental effects of neuron loss on the readout representation are minimized through learning (see the corresponding reduction of reconstruction MSE in Figure 4.5B).

4.6 Improved representation of behaviorally relevant inputs

The ability to dynamically allocate resources for representing important peripheral inputs in a use-dependent manner is a hallmark of cerebral cortex (Buonomano and Merzenich, 1998). Indeed, cortical representations are highly plastic, and the dynamic changes in cortical representation upon manipulations in input or during task learning are thought to underlie the phenomenon of perceptual learning: the improvement of sensory abilities during training (Seitz and Watanabe, 2005). Al-

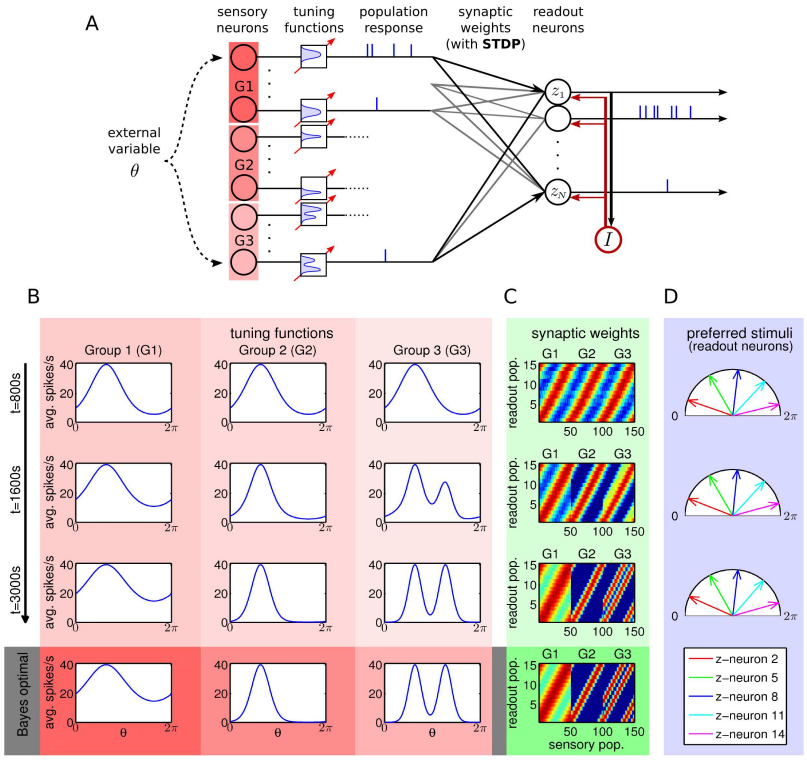

Figure 4.4: Canonical microcircuit maintains optimality in spite of changing tuning functions in the sensory population. **A**. The sensory population was divided into three groups, G1-G3, each group responsive to the whole range of input stimuli. **B**. The simulation then comprised a static phase (0s-1000s), a dynamic phase (1000s - 2000s) and a consolidation phase (2000s - 3000s). In the static phase (first row), constant sensory tuning functions were used analogous to Figure 4.2. In the dynamic phase (second row), tuning functions started to change. Each group of sensory neurons developed different tuning functions. In G1 tuning functions broadened, in G2 they narrowed, and in G3 they became bimodal. In the consolidation phase (third row), tuning functions were fixed again. **C**. The change in the sensory population code during the dynamic phase (second row) resulted in a shift in the attractor centers of the learning dynamics. This became visible in the synaptic readout weights, which slowly tracked the changes in the sensory population (see weight matrix at $t = 1600s$). During the consolidation phase, the synaptic weights then converged to a new optimal representation. Colors range from dark blue (0) to dark red (1.9). **D**. The preferred directions of five representative z-neurons are shown over the course learning. Although the sensory representation drastically changes over the considered period (B), the readout representation remains remarkably stable.

though the exact neural mechanisms which give rise to task-related improvements are largely unknown, experimental evidence suggests an important implication of synaptic plasticity of intra-cortical connections (Buonomano and Merzenich, 1998) in conjunction with top-down "relevance" or reward signals (Seitz and Watanabe,

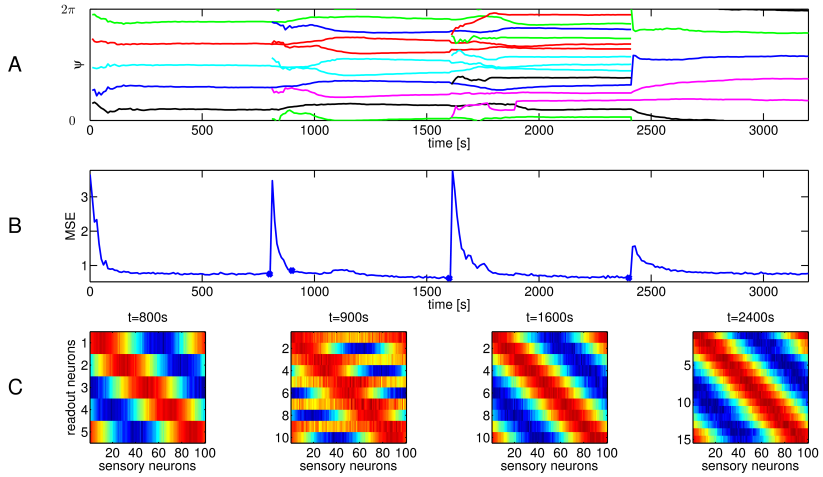

Figure 4.5: STDP enables robust decoding despite neuron growth and death. **A**. The preferred stimuli ψ_k of readout neurons are shown over time. The circuit was initially setup with five readout neurons. Through learning these readout neurons became experts with uniformly distributed preferred stimuli ($t = 800s$). At $t = 800s$, five readout neurons were added to the circuit with randomly initialized weights. They quickly specialized on the most weakly represented ranges of the stimulus space ($t = 1600s$). A similar effect could be observed after adding another five neurons, leading to a yet refined representation (see $1600s - 2400s$). At $t = 2400s$, the cell death of ten readout neurons was simulated. This resulted in large representation gaps, i.e. poorly represented regions of the input stimulus. Through learning, the preferred stimuli of WTA neurons rearranged towards an optimized configuration again ($2400s - 3200s$). **B**. Mean-squared reconstruction error over time. Performance plunged whenever randomly initialized readout neurons were inserted (at $t = 800s$ and $t = 1600s$), before recovering again through learning. Neurons loss at $t = 2400s$ also degraded performance. Although learning minimized the ensuing detrimental effects, the final MSE with 5 neurons ($t = 3200s$) is markedly higher than with 15 neurons ($t = 2400s$). Blue dots correspond to selected time points in C. **C**. Weight matrices at different stages of the simulation (time points indicated by blue dots in B). Colors: dark blue (0) - dark red (2.1). An optimized weight setting for five readout neurons is shown at $t = 800s$. At $t = 900s$, the synaptic weights of five newly grown neurons, amidst previously existing neurons, are depicted shortly after random initialization. Optimized weight settings for 10 and 15 neurons are shown at $t = 1600s$ and $t = 2400s$, respectively.

2005). The latter have been hypothesized to be communicated to local cortical circuits via diffuse neuromodulatory signals, such as "acetylcholine, norepinephrine or dopamine, which gate learning and thus restrict sensory plasticity" (Seitz et al., 2009). Here we will show that, by incorporating such a modulatory signal into the considered cortical microcircuit model, the internal representation of stimuli becomes automatically focused on behaviorally relevant inputs, i.e. those inputs which are consistently paired with high levels of the modulatory signal. The resulting allocation of cortical resources in proportion to relevance is strongly reminiscent

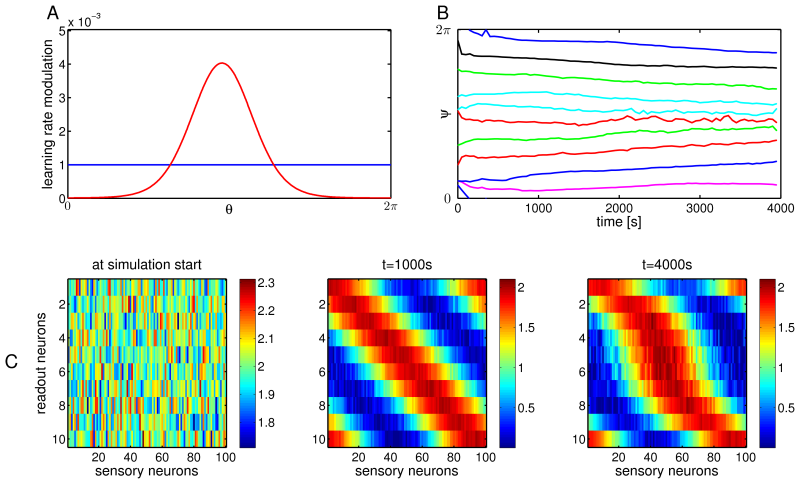

Figure 4.6: Improved representation of behaviorally relevant inputs. **A**. For $1000s$ weights were regulated by (4.5) with a constant learning rate (no modulation, blue line). For the following $3000s$ the learning rate was modulated in dependence of the external variable (red curve), thereby simulating stimulus selective modulation of learning. The external variable $\theta(t)$ and the tuning functions of sensory neurons were simulated as in Figure 4.2. **B**. The simulated cortical microcircuit responded to the modulation by increasing the representation density for those regions which were paired with strong modulation, at the cost of reduced representation density for other regions. **C**. The weight setting which emerged when gating was present ($t = 4000s$) differs from the unmodulated case ($t = 1000s$) by a stronger focus on those population responses which coincided with periods of high "relevance". In particular, after modulated learning more readout neurons receive their strongest synaptic connections from sensory neurons which respond to relevant stimuli (∼sensory neurons 40-60). These readout neurons correspond to the dense region of preferred directions in B. As a consequence of the reallocation of resources, a greater readout resolution is achieved for those regions of the high-dimensional input space which are behaviorally relevant.

of experimentally observed training-dependent cortical map plasticity (Buonomano and Merzenich, 1998).

In a computer simulation, we incorporated a global modulatory signal into the cortical microcircuit considered above, acting as a soft gate for learning. The gating was implemented as a multiplicative factor on the learning rate. During an initial reference phase, modulation remained constant during exposure to stimuli, leading as expected via learning to a uniform internal representation (see Figure 4.6B and Figure 4.6C at $t = 1000s$). Modulation was activated at $t = 1000s$. The modulation was chosen in dependence of the current external stimulus, such that certain ranges of θ were paired with higher levels of the modulatory signal than others (see Figure 4.6A, red curve). As shown, those population responses which coincided with higher levels were stronger imprinted in the synaptic network weights, and consequently, the internal representation started to shift towards a denser concentration

around more "relevant" θs (Figure 4.6B and Figure 4.6C at $t = 4000s$). In contrast, stimuli paired with lower levels of modulation became more crudely represented by the network. Hence, the resulting density of representation in the distribution of preferred stimuli ψ_k is directly connected to the strength of modulation imposed for different θ.

Such use-dependent allocation of cortical resources is a ubiquitous feature of cortical plasticity, and has been shown to correlate with significant performance improvements on corresponding tasks (Buonomano and Merzenich, 1998). Our results indicate that the modulation of learning in a local cortical microcircuit model through a diffuse global signal is indeed sufficient to reproduce relevance-dependent allocation of representational resources. This provides a simple mechanism which could underlie the characteristic improvements in sensory abilities associated with perceptual learning.

4.7 Discussion

Ensembles of pyramidal cells with lateral inhibition on layers 2/3 and layer 5/6 (Fino and Yuste, 2011) constitute a universal network motif of cortical microcircuits in many different cortical areas and different species. We propose that these network motifs acquire through STDP a rather universal computational function, that appears to be essential for many, if not all, cortical areas: the compression of high-dimensional noisy spike inputs into lower-dimensional sparse representations of the most likely hidden causes of these high-dimensional spike inputs. This yields in particular an emergent optimal decoding of population codes in the sense of Jazayeri and Movshon (2006). This emergent computational function of stochastic WTA circuits is very stable, because a rigorous learning theory shows that the theoretically optimal values of synaptic weights are attractors in the dynamics of synaptic weights under the considered class of STDP rules with homeostatic plasticity. A remarkable feature of the underlying learning theory is that it creates a link between local synaptic learning rules such as STDP, and the most powerful known abstract method for autonomous (i.e., unsupervised) learning: Expectation Maximization (EM). This link holds provably for the class of optimal STDP rules, and in practice also approximately for STDP rules which are not directly covered by theory (see Figure 4.3).

Furthermore the learning theory for STDP that we have introduced provides a new benchmark for analyzing and understanding from a functional perspective the large variety of parameters and learning curves for STDP that have been found at different synapses (Dan and Poo, 2006). This learning theory proposes that the theoretically optimal version of STDP (from the perspective of autonomous generation of implicit generative models for high-dimensional noisy spike inputs through Expectation-Maximization) depends both on the firing statistics of presynaptic neurons, and on the average shape of EPSPs at the soma in a systematic

and predictable manner.

In particular, the learning theory for STDP that we have presented throws a new light on the question of downstream decoding of information conveyed by populations of noisy spiking neurons. It had already been shown in Jazayeri and Movshon (2006) that simple linear neurons could in principle carry out a theoretically optimal maximum likelihood decoding, provided that suitable values are chosen for their synaptic weights. The learning theory that we have presented shows that these optimal weight values are in fact attractors with regard to the dynamics of synaptic weights under STDP in stochastic WTA circuits. This learning theory for STDP provides therefore also a possible explanation for the problem how downstream neurons can maintain optimal decoding of population codes of sensory neurons in spite of ubiquitous changes in the tuning functions of sensory neurons (see Figure 4.4) and changes in the number of neurons involved in this decoding (see Figure 4.5). Furthermore if one assumes that synaptic plasticity is gated by neuromodulators or network activity, so that the learning rate of STDP is increased for behaviorally relevant stimuli, the downstream decoding network automatically adapts its resolution in order to achieve a finer representation of behaviorally relevant ranges of external stimuli (see Figure 4.6).

An interesting aspect which we did not study in this article is whether "self-organizing maps", similar to the well-documented orientation maps in cat visual cortex (Hubel and Wiesel, 1963), could emerge in the presented model. Indeed, one could think of clever changes in the model to induce such effects, e.g. adding lateral excitatory connections between neighboring neurons. This would likely facilitate the emergence of locally smooth maps. On the other hand, it is not clear at all whether a downstream "user" of the network output actually requires such a smooth map representation: indeed, many rodents, including rats, mice and squirrels, can live without smooth maps (e.g. Van Hooser et al. (2005)). Furthermore, from a purely functional standpoint, neurons in the presented model with similar tuning (e.g. specialized on similar motion directions) will generally have non-negligible signal correlation (see e.g. Figure 4.2D). Hence, downstream neurons which receive connections from the WTA neurons could quite easily detect groups of neurons which code for similar input stimuli, and establish selective synaptic connections to "functionally neighboring" neurons even if their spatial arrangement is scrambled.

The WTA mechanism in this article relies on an exponential input-output non-linearity of the form e^u to fit the requirements of Bayesian inference. Indeed, this is precisely the input-output relation which has been found empirically by Jolivet et al. (2006), who fit a stochastic spike response model to predict spike timings in pyramidal cells from L5 rat somatosensory cortex. Other authors have suggested a power relation of the form u^p between membrane potential and output rate, e.g. Priebe et al. (2004). Although we expect that power relations would suffice to induce the desired competition, it would be interesting to study the effect of different non-linearities in future work.

One of the model assumptions is conditional independence of inputs, and hence learning is only guaranteed to be optimal if there are no noise correlations in the input (although convergence is guaranteed regardless). Many studies have found moderate noise correlations in cortex (Smith and Kohn, 2008; Huang and Lisberger, 2009). On the other hand, the recent study by Ecker et al. (2010) found that in primary visual cortex even nearby neurons with similar tuning show close to zero correlation (and the authors argue that earlier findings may need to be reevaluated under more precise experimental paradigms). Nonetheless, we tested how non-zero noise correlations affect network performance. Figure D.2 shows that for a moderately correlated input population code the performance remains quite high, although not optimal. This has to be expected from a learner which relies on the independence assumption, and in accordance with the findings of Graf et al. (2011) who showed that linear readouts can be further improved if input correlations are taken into account. Whether input correlations could be efficiently exploited in the context of the present article, i.e. whether and how the correlation structure in the inputs could be learned autonomously in an unsupervised manner (Graf et al. (2011) used supervised learning) by a spiking network with local plasticity rules, remains an open question for future work.

Another model assumption we made is that the stimulus distribution is uniform. Note that this does not invalidate the presented results for non-uniformly distributed inputs: even in a non-uniform stimulus setting, learning is still guaranteed to converge to an optimal solution with respect to the fixed prior distribution (since the assumption is made on the level of the implicit generative model, not on the input distribution $p^*(\mathbf{x})$). The solution produced by the network for a non-uniform stimulus distribution resembles Figure 4.6: a greater number of neurons specializes on the more likely region of the stimulus (see Figure D.3). This is precisely what is expected from a maximum likelihood learner with a fixed prior model distribution. Another important question is whether the network also adapts its implicit prior distribution over the stimulus in an optimal manner. In other words, does the WTA network automatically become biased towards activating those neurons whose preferred directions occur more often in the input? Indeed, this turns out to be the case: Since high probability stimuli will automatically attract more WTA neurons during the specialization process, the implicit prior distribution of the network favors high probability stimuli after learning. Indeed this phenomenon is also quantitatively consistent with the prediction of a Bayesian framework (Figure D.3).

Related work

As a model for decoding of high-dimensional population codes in the brain, the key novelty of the present work is plasticity. In particular, this article demonstrates to the best of our knowledge for the first time that population codes can not only be read out efficiently (this had already been shown by Deneve et al. (1999), Jazayeri

and Movshon (2006) and Chaisanguanthum and Lisberger (2011)), but that this readout can also be learned optimally by a stereotypical cortical microcircuit motif, in a process which is entirely autonomous and self-organizing. An interesting difference to previous models for population decoding lies in the output code of the readout circuit, which can represent a whole distribution (in contrast to Deneve et al. (1999)), while being sparse for typical stimuli (as opposed to the predictions of Jazayeri and Movshon (2006)). Hence, the information which is extracted about the inputs is represented and conveyed by the whole network population, rather than by the identity of a single neuron. Strong, unambiguous stimuli will elicit sharp responses, whereas weaker/low-contrast/ambiguous stimuli will lead to a distributed code. Furthermore, our model predicts that neural activity represents *samples* from a probability distribution, a coding scheme which has recently attracted considerable attention, as it appears to be particularly suitable for probabilistic representations subject to learning and adaptation (Fiser et al., 2010). As a consequence, our model is not only consistent with the experimentally observed trial-to-trial variability of neuronal responses, but in fact requires that neurons respond in a stochastic manner. Hence, one can view this article also as a contribution to the growing literature on computational properties of networks of stochastically firing neurons.

The link to EM extends the presented model's generality beyond population coding, to input representations which cannot be easily described in terms of a fixed number of external variables. In this more general context, different authors have proposed neural network models which are able to carry out Bayesian inference (Doya et al., 2007), and some also considered the question how probabilistic representations could emerge autonomously through learning, e.g. Dayan et al. (1995); Rao and Ballard (1999), or Keck et al. (2012) who used a related approach as the one developed here, but required more approximations and did not model spiking neurons (and hence also not the relation between EM and STDP). While these models focused on rather artificial neural networks based on abstract (either binary or continuous-valued rate-based) neural units, recent studies have started to address whether also more realistic, spiking neural networks are capable of acquiring optimized probabilistic representations through autonomous learning. A recent model by Deneve (2008b) focused on a single spiking neuron and showed that such a neuron can in principle learn an efficient code for its inputs in a basic temporal generative model. Whether parameter learning in the model can be scaled up to multiple neurons, which would be required to cope with complex but biologically realistic input distributions, remains an interesting open issue. Finally, the recent model by Nessler et al. (2010) showed that spiking neurons can learn to represent a mixture of multinomial input variables, assuming that exactly one input neuron in a group is active at a time. The generalization and extension of this result to biologically realistic firing statistics and EPSP shapes, required for the decoding of realistic population codes, was described in the present article.

Finally, Law and Gold (2009) showed in a related work that near-optimal population readouts can also be learned autonomously through a simple reinforcement learning (RL) rule. More precisely, they demonstrate that for a given decision problem, e.g. whether the motion direction encoded by the input is $< 180°$ or $> 180°$, near-optimal linear readouts from an input population can be found through RL. The learning scheme relies on a feedback signal which conveys the correctness of the decisions computed from the readout. In principle, for any particular decision problem a different set of readout weights is required. Our work complements these findings insofar as we show that an (external) feedback signal is actually not needed for the emergence of optimal readouts. Instead, optimal readout weights in a downstream population of neurons can emerge through a purely self-supervising process. Furthermore, in principle any decision problem can then be reduced to simply counting spikes emitted by the network population, e.g. checking whether those output neurons with preferred directions $< 180°$ produced more spikes than those with preferred directions $> 180°$. Given the attractive properties of both approaches and the prevalence of both STDP and reward-modulated learning in the brain, it is not unreasonable to assume that cortex employs a combination of these learning mechanisms.

Experimentally testable predictions

Our results predict that abolishment of STDP during a critical period prevents the emergence of sparse codes for frequently occurring sensory stimuli. They also predict, that with intact STDP, the coding properties of pyramidal cells will change in a predictable manner in response to changes in the distribution of external stimuli or their behavioral relevance (see Figure 4.6), since this change will be tracked by the implicit generative model of ensembles of pyramidal cells (probably on layers 2/3 and layers 5/6). In addition, a lesion of some neurons within this ensemble will cause a redistribution of neural codes among the remaining neurons (see Figure 4.5). Our theoretical analysis of optimal versions of STDP predicts a specific dependence of features of STDP on the firing statistics of presynaptic neurons that can in principle be tested experimentally.

Furthermore, our model predicts that in WTA neurons, i.e. pyramidal cells in layers 2/3 and layers 5/6, the tuning of *firing rates* should be sharper than the tuning of *membrane potentials*. In fact, this is the well-known experimentally observed ice-berg effect (Carandini and Ferster, 2000). But in our model the ice-berg effect also has a novel functional interpretation from the perspective of Bayesian inference: the instantaneous firing rate must depend on the current membrane potential via a sharpening exponential activation function in order to ensure that neurons encode probabilities. Hence, the ice-berg effect is a prerequisite for correct learning of optimal decoding weights. Another direct prediction of our model is that the stimulus selectivity in ensembles of pyramidal cells with lateral inhibition is sharper when familiar stimuli are presented compared to novel stimuli. This effect

has been recently reported in monkey Inferior Temporal Cortex (ITC) by Freedman et al. (2006).

Conclusion

In summary, we have shown that in conjunction with STDP, a common network motif of cortical microcircuits acquires a generic computational function: it creates a sparse representation of complex high-dimensional spike inputs to a local microcircuit. Although we have discussed in this article only the application to decoding of information from populations of noisy sensory neurons, the generation of sparse representations for complex high dimensional inputs, which converge onto a microcircuit from many different brain areas, is a candidate for a generic computational operation that is meaningful for microcircuits in any cortical area.

4.8 Acknowledgments

This chapter is based on a joint work with Helmut Puhr (HP) and Wolfgang Maass (WM) which was published 2013 in Neural Computation ("Emergence of Optimal Decoding of Population Codes through STDP"). The idea of the paper was conceived by WM. The theory was developed by SH. The computer experiments were designed by SH, HP and WM. The software framework was written and the simulations were conducted by HP. The figures were designed by SH, HP and WM. The manuscript was written by SH, under supervision of WM.

Chapter 5

Interplay of Synaptic and Homeostatic Plasticity in Stochastic Spiking Networks

Contents

Recent spiking network models of Bayesian inference and unsupervised learning frequently assume either inputs to arrive in a special format or employ complex computations in neuronal activation functions and synaptic plasticity rules. Here we show in a rigorous mathematical treatment how homeostatic processes, which have previously received little attention in this context, can overcome common theoretical limitations and facilitate the neural implementation and performance of existing models. In particular, we show that homeostatic plasticity can be understood as the enforcement of a 'balancing' posterior constraint during probabilistic inference and learning with Expectation Maximization. We link homeostatic dynamics to the theory of variational inference, and show that nontrivial terms, which typically appear during probabilistic inference in a large class of models, drop out. We demonstrate the feasibility of our approach in a spiking Winner-Take-All architecture of Bayesian inference and learning. Finally, we sketch how the mathematical framework can be extended to richer recurrent network architectures. Altogether, our theory provides a novel perspective on the interplay of homeostatic processes and synaptic plasticity in cortical microcircuits, and points to an essential role of homeostasis during inference and learning in spiking networks.

5.1 Introduction

Experimental findings from neuro- and cognitive sciences have led to the hypothesis that humans create and maintain an internal model of their environment in neuronal circuitry of the brain during learning and development (Körding and Wolpert, 2004; Orban et al., 2008; Fiser et al., 2010; Berkes et al., 2011), and employ this model for Bayesian inference in everyday cognition (Griffiths and Tenenbaum, 2006; Angelaki et al., 2009). Yet, how these computations are carried out in the brain remains largely unknown. A number of innovative models has been proposed recently which demonstrate that in principle, spiking networks can carry out quite complex probabilistic inference tasks (Deneve, 2008a; Steimer et al., 2009; Buesing et al., 2011; Pecevski et al., 2011), and even learn to adapt to their inputs near optimally through various forms of plasticity (Deneve, 2008b; Nessler et al., 2010; Brea et al., 2012; Rezende et al., 2012; Keck et al., 2012). Still, in network models for concurrent online inference and learning, most approaches introduce distinct assumptions: Both (Nessler et al., 2010) in a spiking Winner-take-all (WTA) network, and (Keck et al., 2012) in a rate based WTA network, identified the limitation that inputs must be normalized before being presented to the network, in order to circumvent an otherwise nontrivial (and arguably non-local) dependency of the intrinsic excitability on all afferent synapses of a neuron. Nessler et al. (Nessler et al., 2010) relied on population coded input spike trains; Keck et al. (Keck et al., 2012) proposed feed-forward inhibition as a possible neural mechanism to achieve this normalization. A theoretically related issue has been encountered by Deneve (Deneve, 2008a,b), in which inference and learning is realized in a two-state Hidden Markov Model by a single spiking neuron. Although synaptic learning rules are found to be locally computable, the learning update for intrinsic excitabilities remains intricate. In a different approach, Brea et al. (Brea et al., 2012) have recently proposed a promising model for Bayes optimal sequence learning in spiking networks in which a global reward signal, which is computed from the network state and synaptic weights, modulates otherwise purely local learning rules. Also the recent innovative model for variational learning in recurrent spiking networks by Rezende et al. (Rezende et al., 2012) relies on sophisticated updates of variational parameters that complement otherwise local learning rules.

There exists great interest in developing Bayesian spiking models which require minimal non-standard neural mechanisms or additional assumptions on the input distribution: such models are expected to foster the analysis of biological circuits from a Bayesian perspective (Tenenbaum et al., 2011), and to provide a versatile computational framework for novel neuromorphic hardware (Schemmel et al., 2010). With these goals in mind, we introduce here a novel theoretical perspective on homeostatic plasticity in Bayesian spiking networks that complements previous approaches by constraining statistical properties of the *network response* rather than the input distribution. In particular we introduce *'balancing' posterior con-*

straints which can be implemented in a purely local manner by the spiking network through a simple rule that is strongly reminiscent of homeostatic intrinsic plasticity in cortex (Desai et al., 1999; Watt and Desai, 2010). Importantly, it turns out that the emerging network dynamics eliminate a particular class of nontrivial computations that frequently arise in Bayesian spiking networks.

First we develop the mathematical framework for Expectation Maximization (EM) with homeostatic posterior constraints in an instructive Winner-Take-all network model of probabilistic inference and unsupervised learning. Building upon the theoretical results of (Graca et al., 2008), we establish a rigorous link between homeostatic intrinsic plasticity and variational inference. In a second step, we sketch how the framework can be extended to recurrent spiking networks; by introducing posterior constraints on the correlation structure, we recover local plasticity rules for recurrent synaptic weights.

5.2 Theory for balanced autonomous learning in WTA circuits

We first introduce, as an illustrative and representative example, a generative mixture model $p(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{V})$ with hidden causes $\boldsymbol{z}$ and binary observed variables $\boldsymbol{y}$, and a spiking WTA network $\mathcal{N}$ which receives inputs $\boldsymbol{y}(t)$ via synaptic weights $\boldsymbol{V}$. As shown in (Nessler et al., 2010), such a network $\mathcal{N}$ can implement probabilistic inference $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{V})$ through its spiking dynamics, and maximum likelihood learning through local synaptic learning rules (see Figure 1A). The mixture model comprises K binary and mutually exclusive components $z_k \in \{0, 1\}$, $\sum_{k=1}^{K} z_k = 1$, each specialized on a different N-dimensional input pattern:

$$p(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{V}) = \prod_{k=1}^{K} e^{\hat{b}_k z_k} \prod_{i=1}^{N} \left[(\pi_{ki})^{y_i} \cdot (1 - \pi_{ki})^{1-y_i} \right]^{z_k} \tag{5.1}$$

$$\Leftrightarrow \log p(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{V}) = \sum_k z_k \left(\sum_i V_{ki} y_i - A_k + \hat{b}_k \right) \ , \tag{5.2}$$

$$\text{with } \sum_k e^{\hat{b}_k} = 1 \text{ and } \pi_{ki} = \sigma(V_{ki}) \text{ and } A_k = \sum_i \log(1 + e^{V_{ki}}) \ , \tag{5.3}$$

where $\sigma(x) = (1 + \exp(-x))^{-1}$ denotes the logistic function, and π_{ki} the expected activation of input i under the mixture component k. For simplicity and notational convenience, we will treat the prior parameters $\hat{b}_k$ as constants throughout the paper. Probabilistic inference of hidden causes z_k based on an observed input $\boldsymbol{y}$ can be implemented by a spiking WTA network $\mathcal{N}$ of K neurons which fire with the instantaneous spiking probability (for $\delta t \to 0$),

$$p(z_k \text{ spikes in } [t, t + \delta t]) = \delta t \cdot r_{\text{net}} \cdot \frac{e^{u_k(t)}}{\sum_j e^{u_j(t)}} \propto p(z_k = 1|\boldsymbol{y}, \boldsymbol{V}) \ , \tag{5.4}$$

with the input potential $u_k(t) = \sum_i V_{ki} y_i(t) - A_k + \hat{b}_k$. Each WTA neuron k receives spiking inputs y_i via synaptic weights V_{ki} and responds with an instantaneous spiking probability which depends exponentially on its input potential u_k in accordance with biological findings (Jolivet et al., 2006). Stochastic winner-take-all (soft-max) competition between the neurons is modeled via divisive normalization (5.4) (Simoncelli and Heeger, 1998). The input is defined as $y_i(t) = 1$ if input neuron i emitted a spike within the last τ milliseconds, and 0 otherwise, corresponding to a rectangular post-synaptic potential (PSP) of length τ. We define $z_k(t) = 1$ at spike times t of neuron k and $z_k(t) = 0$ otherwise.

In addition to the spiking input, each neuron's potential u_k features an intrinsic excitability $-A_k + \hat{b}_k$. Note that, besides the prior constant $\hat{b}_k$, this excitability depends on the normalizing term A_k, and hence on all afferent synaptic weights through (5.3): WTA neurons which encode strong patterns with high probabilities π_{ki} require lower intrinsic excitabilities, while neurons with weak patterns require larger excitabilities. In the presence of synaptic plasticity, i.e., time-varying V_{ki}, it is unclear how biologically realistic neurons could communicate ongoing changes in synaptic weights from distal synaptic sites to the soma. This critical issue was apparently identified in (Nessler et al., 2010) and (Keck et al., 2012); both papers circumvent the problem (in similar probabilistic models) by constraining the input $\boldsymbol{y}$ (and also the synaptic weights in (Keck et al., 2012)) in order to maintain constant and uniform values A_k across all WTA neurons.

Here, we propose a different approach to cope with the nontrivial computations A_k during inference and learning in the network. Instead of assuming that the inputs $\boldsymbol{y}$ meet a normalization constraint, we constrain the *network response* during inference, by applying homeostatic dynamics to the intrinsic excitabilities. This approach turns out to be beneficial in the presence of time-varying synaptic weights, i.e., during ongoing changes of V_{ki} and A_k. The resulting interplay of intrinsic and synaptic plasticity can be best understood from the standard EM lower bound (Bishop, 2006),

$$F(\boldsymbol{V}, q(\boldsymbol{z}|\boldsymbol{y})) = L(\boldsymbol{V}) - \langle \operatorname{KL}(q(\boldsymbol{z}|\boldsymbol{y}) \,||\, p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{V})) \rangle_{p^*(\boldsymbol{y})} \qquad \rightarrow \text{E-step} \quad , \quad (5.5)$$

$$= \langle \log p(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{V}) \rangle_{p^*(\boldsymbol{y})q(\boldsymbol{z}|\boldsymbol{y})} + \langle H(q(\boldsymbol{z}|\boldsymbol{y})) \rangle_{p^*(\boldsymbol{y})} \qquad \rightarrow \text{M-step} \quad , \quad (5.6)$$

where $L(\boldsymbol{V}) = \langle \log p(\boldsymbol{y}|\boldsymbol{V}) \rangle_{p^*(\boldsymbol{y})}$ denotes the log-likelihood of the input under the model, $\operatorname{KL}(\cdot \,||\, \cdot)$ the Kullback-Leibler divergence, and $H(\cdot)$ the entropy. The decomposition holds for arbitrary distributions q. In hitherto proposed neural implementations of EM (Deneve, 2008b; Nessler et al., 2010; Keck et al., 2012; Sato, 1999), the network implements the current posterior distribution in the E-step, i.e., $q = p$ and $\operatorname{KL}(q \,||\, p) = 0$. In contrast, by applying homeostatic plasticity, the network response will be constrained to implement a variational posterior from a class of "homeostatic" distributions $\mathcal{Q}$: the long-term average activation of each WTA neuron z_k is constrained to an a priori defined target value. Notably, we will see that the resulting network response q^* describes an optimal variational E-Step in the

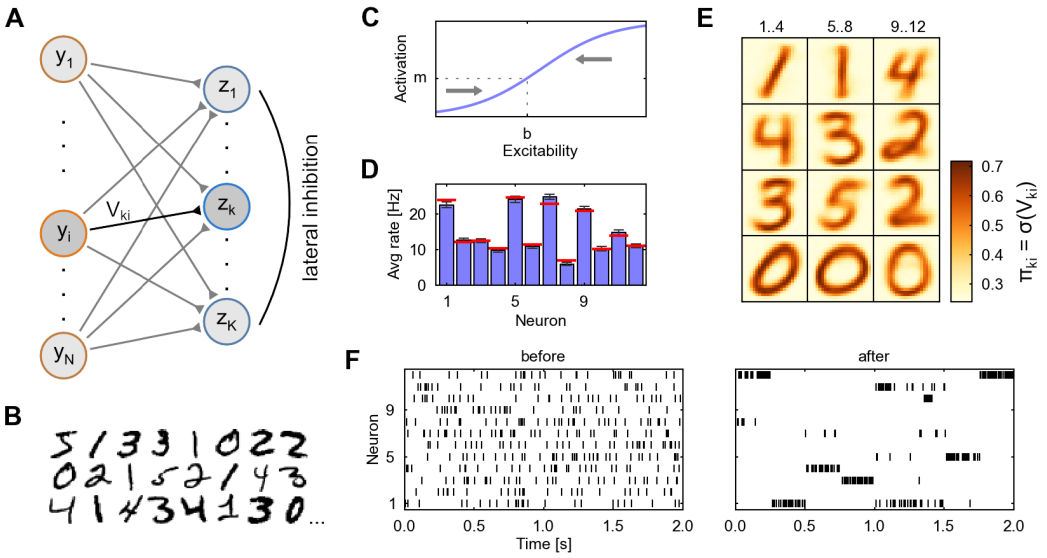

Figure 5.1: **A**. Spiking WTA network model. **B**. Input templates from MNIST database (digits *0-5*) are presented in random order to the network as spike trains (the input template switches after every 250ms, black/white pixels are translated to high/low firing rates between 20 and 90 Hz). **C**. Sketch of intrinsic homeostatic plasticity maintaining a certain target average activation. **D**. Homeostatic plasticity induces average firing rates (blue) close to target values (red). **E**. After a learning period, each WTA neuron has specialized on a particular input motif. **F**. WTA output spikes during a test phase before and after learning. Learning leads to a sparse output code.

sense that $q^*(\boldsymbol{z}|\boldsymbol{y}) = \arg\min_{q\in\mathcal{Q}} \mathrm{KL}\left(q(\boldsymbol{z}|\boldsymbol{y}) \,||\, p(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{V})\right)$. Importantly, homeostatic plasticity fully regulates the intrinsic excitabilities, and as a side effect eliminates the non-local terms A_k in the E-step, while synaptic plasticity of the weights V_{ki} optimizes the underlying probabilistic model $p(\boldsymbol{y},\boldsymbol{z}|\boldsymbol{V})$ in the M-step.

In summary, the network response implements q^* as the variational E-step, the M-Step can be performed via gradient ascent on (5.6) with respect to V_{ki}. As derived in section 5.2, this gives rise to the following temporal dynamics and plasticity rules in the spiking network, which instantiate a stochastic version of the variational EM scheme:

$$u_k(t) = \sum_i V_{ki} y_i(t) + b_k \;, \qquad \dot{b}_k(t) = \eta_b \cdot (r_{\text{net}} \cdot m_k - \delta(z_k(t) - 1)) \;, \tag{5.7}$$

$$\dot{V}_{ki}(t) = \eta_V \cdot \delta(z_k(t) - 1) \cdot (y_j(t) - \sigma(V_{ki})) \;, \tag{5.8}$$

where $\delta(\cdot)$ denotes the Dirac delta function, and η_b, η_V are learning rates (which were kept time-invariant in the simulations with $\eta_b = 10 \cdot \eta_V$). Note that (5.8) is a spike-timing dependent plasticity rule (cf. (Nessler et al., 2010)) and is non-zero only at post-synaptic spike times t, for which $z_k(t) = 1$. The effect of the homeostatic intrinsic plasticity rule (5.7) is illustrated in Figure 5.1C: it aims to

keep the long-term average activation of each WTA neuron k close to a certain target value m_k. More precisely, if r_k is a neuron's long-term average firing rate, then homeostatic plasticity will ensure that $r_k/r_{\text{net}} \approx m_k$. The target activations $m_k \in (0,1)$ can be chosen freely with the obvious constraint that $\sum_k m_k = 1$. Note that (5.7) is strongly reminiscent of homeostatic intrinsic plasticity in cortex (Desai et al., 1999; Watt and Desai, 2010).

We have implemented these dynamics in a computer simulation of a WTA spiking network $\mathcal{N}$. Inputs $\boldsymbol{y}(t)$ were defined by translating handwritten digits *0-5* (Figure 5.1B) from the MNIST dataset (LeCun et al., 1998) into input spike trains. Figure 5.1D shows that, at the end of a 10^4s learning period, homeostatic plasticity has indeed achieved that $r_k \approx r_{\text{net}} \cdot m_k$. Figure 5.1E illustrates the patterns learned by each WTA neuron after this period (shown are the π_{ki}). Apparently, the WTA neurons have specialized on patterns of different intensity which correspond to different values of A_k. Figure 5.1F shows the output spiking behavior of the circuit before and after learning in response to a set of test patterns. The specialization to different patterns has led to a distinct sparse output code, in which any particular test pattern evokes output spikes from only one or two WTA neurons. Note that homeostasis forces all WTA neurons to participate in the competition, and thus prevents neurons from becoming underactive if their synaptic weights decrease, and from becoming overactive if their synaptic weights increase, much like the original A_k terms (which are nontrivial to compute for the network). Indeed, the learned synaptic parameters and the resulting output behavior corresponds to what would be expected from an optimal learning algorithm for the mixture model (5.1)-(5.3).[1]

Theory for the WTA model

In the following, we develop the three theoretical key results for the WTA model (5.1)-(5.3):

- Homeostatic intrinsic plasticity finds the network response distribution $q^*(\boldsymbol{z}|\boldsymbol{y}) \in \mathcal{Q}$ closest to the posterior distribution $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{V})$, from a set of "homeostatic" distributions $\mathcal{Q}$.

- The interplay of homeostatic and synaptic plasticity can be understood from the perspective of variational EM.

- The critical non-local terms A_k defined by (5.3) drop out of the network dynamics.

[1] Without adaptation of intrinsic excitabilities, the network would start performing erroneous inference, learning would reinforce this erroneous behavior, and performance would quickly break down. We have verified this in simulations for the present WTA model: Consistently across trials, a small subset of WTA neurons became dominantly active while most neurons remained silent.

E-step: variational inference with homeostasis

The variational distribution $q(\boldsymbol{z}|\boldsymbol{y})$ we consider for the model (5.1)-(5.3) is a $2^N \cdot K$ dimensional object. Since q describes a conditional probability distribution, it is non-negative and normalized for all $\boldsymbol{y}$. In addition, we constrain q to be a "homeostatic" distribution $q \in \mathcal{Q}$ such that the average activation of each hidden variable (neuron) z_k equals an a-priori specified mean activation m_k under the input statistics $p^*(\boldsymbol{y})$. This is sketched in Figure 5.2. Formally we define the constraint set,

$$\mathcal{Q} = \{q : \langle z_k \rangle_{p^*(\boldsymbol{y})q(\boldsymbol{z}|\boldsymbol{y})} = m_k, \text{ for all } k = 1 \dots K\} \ , \qquad \text{with } \sum_k m_k = 1 \ . \quad (5.9)$$

The constrained maximization problem $q^*(\boldsymbol{z}|\boldsymbol{y}) = \arg\max_{q \in \mathcal{Q}} F(\boldsymbol{V}, q(\boldsymbol{z}|\boldsymbol{y}))$ can be solved with the help of Lagrange multipliers (cf. (Graca et al., 2008)). We find that the q^* which maximizes the objective function F during the E-step (and thus minimizes the KL-divergence to the posterior $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{V})$) has the convenient form $q^*(\boldsymbol{z}|\boldsymbol{y}) \propto p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{V}) \cdot \exp(\sum_k \beta_k^* z_k)$ with some β_k^*. Hence, it suffices to consider distributions of the form,

$$q_{\boldsymbol{\beta}}(\boldsymbol{z}|\boldsymbol{y}) \propto \exp(\sum_k z_k (\sum_i V_{ki} y_i + \underbrace{\hat{b}_k - A_k + \beta_k}_{=: b_k})) \ , \quad (5.10)$$

for the maximization problem. We identify β_k as the variational parameters which remain to be optimized. Note that any distribution of this form can be implemented by the spiking network $\mathcal{N}$ if the intrinsic excitabilities are set to $b_k = -A_k + \hat{b}_k + \beta_k$. The optimal variational distribution $q^*(\boldsymbol{z}|\boldsymbol{y}) = q_{\boldsymbol{\beta}^*}(\boldsymbol{z}|\boldsymbol{y})$ then has $\boldsymbol{\beta}^* = \arg\max_{\boldsymbol{\beta}} \Psi(\boldsymbol{\beta})$, i.e. the variational parameter vector which maximizes the dual (Graca et al., 2008),

$$\Psi(\boldsymbol{\beta}) = \sum_k \beta_k m_k - \langle \log \sum_{\boldsymbol{z}} p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{V}) \exp(\sum_k \beta_k z_k) \rangle_{p^*(\boldsymbol{y})} \ . \quad (5.11)$$

Due to concavity of the dual, a unique global maximizer $\boldsymbol{\beta}^*$ exists, and thus also the corresponding optimal intrinsic excitabilities $b_k^* = -A_k + \hat{b}_k + \beta_k^*$ are unique. Hence, the posterior constraint $q \in \mathcal{Q}$ can be illustrated as in Figure 5.2B: For each synaptic weight configuration $\boldsymbol{V}$ there exists, under a particular input distribution $p^*(\boldsymbol{y})$, a unique configuration of intrinsic excitabilities $\boldsymbol{b}$ such that the resulting network output fulfills the homeostatic constraints. The theoretical relation between the intrinsic excitabilities b_k, the original nontrivial term $-A_k$ and the variational parameters β_k is sketched in Figure 5.2C. Importantly, while b_k is implemented in the network, A_k, β_k and $\hat{b}_k$ are not explicitly represented in the implementation anymore. Finding the optimal $\boldsymbol{b}$ in the dual perspective, i.e. those intrinsic excitabilities which fulfill the homeostatic constraints, amounts to gradient ascent

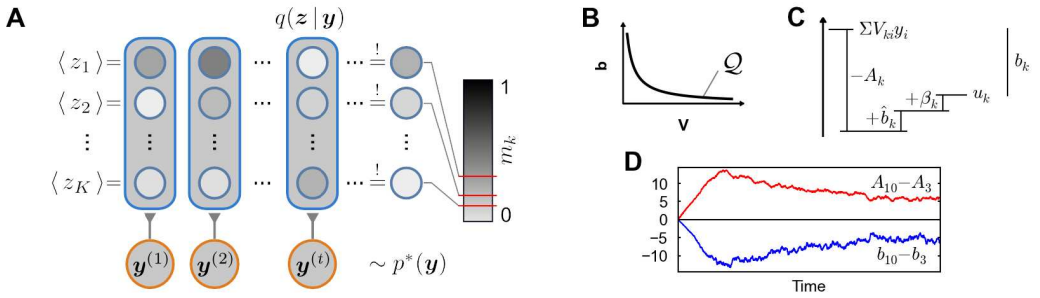

Figure 5.2: **A**. Homeostatic posterior constraints in the WTA model: Under the variational distribution q, the average activation of each variable z_k must equal m_k. **B**. For each set of synaptic weights $\boldsymbol{V}$ there exists a unique assignment of intrinsic excitabilities $\boldsymbol{b}$, such that the constraints are fulfilled. **C**. Theoretical decomposition of the intrinsic excitability b_k into $-A_k$, $\hat{b}_k$ and β_k. **D**. During variational EM the b_k predominantly "track" the dynamically changing non-local terms $-A_k$ (relative comparison between two WTA neurons from Figure 5.1).

$\partial_{\boldsymbol{\beta}} \Psi(\boldsymbol{\beta})$ on the dual, which leads to the following homeostatic learning rule for the intrinsic excitabilities,

$$\Delta b_k \propto \partial_{\beta_k} \Psi(\boldsymbol{\beta}) = m_k - \langle z_k \rangle_{p^*(\boldsymbol{y}) q(\boldsymbol{z}|\boldsymbol{y})} \quad . \tag{5.12}$$

Note that the intrinsic homeostatic plasticity rule (5.7) in the network corresponds to a sample-based stochastic version of this theoretically derived adaptation mechanism (5.12). Hence, given enough time, homeostatic plasticity will automatically install near-optimal intrinsic excitabilities $\boldsymbol{b} \approx \boldsymbol{b}^*$ and implement the correct variational distribution q^* up to stochastic fluctuations in $\boldsymbol{b}$ due to the non-zero learning rate η_b. The non-local terms A_k have entirely dropped out of the network dynamics, since the intrinsic excitabilities b_k can be arbitrarily initialized, and are then fully regulated by the local homeostatic rule, which does not require knowledge of A_k.

As a side remark, note that although the variational parameters β_k are not explicitly present in the implementation, they can be theoretically recovered from the network at any point, via $\beta_k = b_k + A_k - \hat{b}_k$. Notably, in all our simulations we have consistently found small absolute values of β_k, corresponding to a small KL-divergence between q^* and p.[2] Hence, a major effect of the local homeostatic plasticity rule during learning is to dynamically track and effectively implement the non-local terms $-A_k$. This is shown in Figure 5.2D, in which the relative excitabilities of two WTA neurons $b_k - b_j$ are plotted against the corresponding non-local $A_k - A_j$ over the course of learning in the first simulation (Figure 5.1).

[2]This is assuming for simplicity uniform prior parameters $\hat{b}_k$. Note that a small KL-divergence is in fact often observed during variational EM since F, which contains the negative KL-divergence, is being maximized.

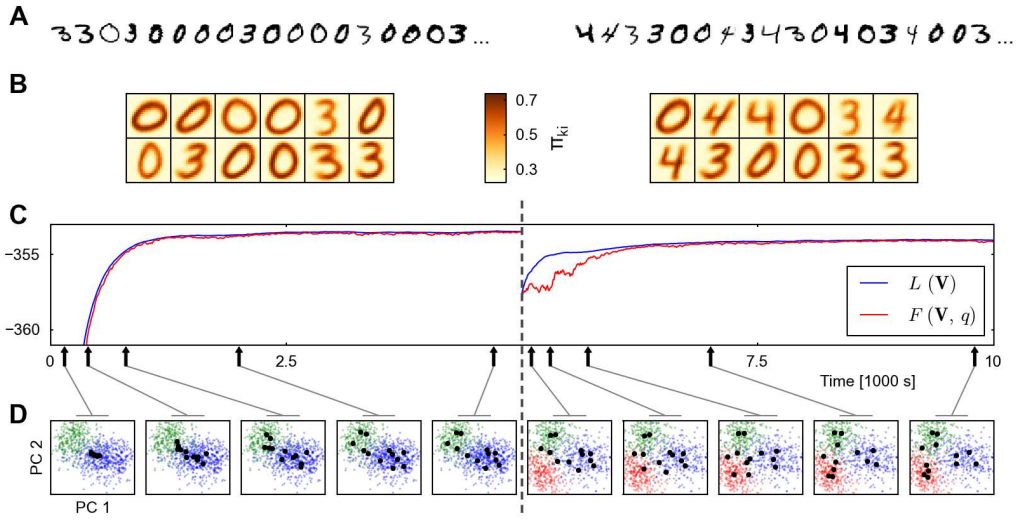

Figure 5.3: **A**. Input templates from MNIST dataset (digits *0,3* at a ratio 2:1, and digits *0,3,4* at a ratio 1:1:1) used during the first and second learning period, respectively. **B**. Learned patterns at the end of each learning period. **C**. Network performance converges in the course of learning. F is a tight lower bound to L. **D**. Illustration of pattern learning and re-learning dynamics in a 2-D projection in the input space. Each black dot corresponds to the pattern π_{ki} of one WTA neuron k. Colored dots are input samples from the training set (blue/green/red $\leftrightarrow$ digits *0/3/4*).

M-step: interplay of synaptic and homeostatic intrinsic plasticity

During the M-step, we aim to increase the EM lower bound F in (5.6) w.r.t. the synaptic parameters $\boldsymbol{V}$. Gradient ascent yields,

$$\partial_{V_{ki}} F(\boldsymbol{V}, q(\boldsymbol{z}|\boldsymbol{y})) = \langle \partial_{V_{ki}} \log p(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{V}) \rangle_{p^*(\boldsymbol{y})q(\boldsymbol{z}|\boldsymbol{y})} \tag{5.13}$$

$$= \langle\, z_k \cdot (y_j - \sigma(V_{ki}))\, \rangle_{p^*(\boldsymbol{y})q(\boldsymbol{z}|\boldsymbol{y})} \quad , \tag{5.14}$$

where q is the variational distribution determined during the E-step, i.e., we can set $q = q^*$. Note the formal correspondence of (5.14) with the network synaptic learning rule (5.8). Indeed, if the network activity implements q^*, it can be shown easily that the expected update of synaptic weights due to the synaptic plasticity (5.8) is proportional to (5.14), and hence implements a stochastic version of the theoretical M-step (cf. (Nessler et al., 2010)).

Dynamical properties of the Bayesian spiking network with homeostasis

To highlight a number of salient dynamical properties emerging from homeostatic plasticity in the considered WTA model, Figure 5.3 shows a simulation of the same network $\mathcal{N}$ with homeostatic dynamics as in Figure 5.1, only with different input

statistics presented to the network, and uniform $m_k = \frac{1}{K}$. During the first 5000s, different writings of *0*'s and *3*'s from the MNIST dataset were presented, with *0*'s occurring twice as often as *3*'s. Then the input distribution $p^*(\boldsymbol{y})$ abruptly switched to include also *4*'s, with each digit occurring equally often. The following observations can be made: Due to the homeostatic constraint, each neuron responds on average to $m_k \cdot T$ out of T presented inputs. As a consequence, the number of neurons which specialize on a particular digit is directly proportional to the frequency of occurrence of that digit, i.e. 8:4 and 4:4:4 after the first and second learning period, respectively (Figure 5.3B). In general, if uniform target activations m_k are chosen, output resources are allocated precisely in proportion to input frequency. Figure 5.3C depicts the time course of the EM lower bound F as well as the average likelihood L (assuming uniform $\hat{b}_k$) under the model during a single simulation run, demonstrating both convergence and tightness of the lower bound. As expected due to the stabilizing dynamics of homeostasis, we found variability in performance among different trials to be small (not shown). Figure 5.3D illustrates the dynamics of learning and re-learning of patterns π_{ki} in a 2D projection of input patterns onto the first two principal components.

5.3 Extended theory for recurrent stochastic spiking networks

The neural model so far was essentially a feed-forward network, in which every postsynaptic spike can directly be interpreted as one sample of the instantaneous posterior distribution (Nessler et al., 2010). The lateral inhibition served only to ensure the normalization of the posterior. We will now extend the concept of homeostatic processes as posterior constraints to the broader class of recurrent networks and sketch the utility of the developed framework beyond the regulation of intrinsic excitabilities.

Recently it was shown in (Buesing et al., 2011; Pecevski et al., 2011) that recurrent networks of stochastically spiking neurons can in principle carry out probabilistic inference through a sampling process. At every point in time, the joint network state $\boldsymbol{z}(t)$ represents one sample of a posterior. However, (Buesing et al., 2011) and (Pecevski et al., 2011) did not consider unsupervised learning on spiking input streams.

For the following considerations, we divide the definition of the probabilistic model in two parts. First, we define a Boltzmann distribution,

$$p(\boldsymbol{z}) = \exp(\sum_k \hat{b}_k z_k + \frac{1}{2}\sum_{j\neq k} \hat{W}_{kj} z_k z_j)/\text{norm.} \ , \tag{5.15}$$

with $\hat{W}_{kj} = \hat{W}_{jk}$ as "prior" for the hidden variables $\boldsymbol{z}$ which will be represented by a recurrently connected network of K spiking neurons. For the purpose of this section,

we treat $\hat{b}_k$ and $\hat{W}_{kj}$ as constants. Secondly, we define a conditional distribution in the exponential-family form (Bishop, 2006),

$$p(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{V}) = \exp(f_0(\boldsymbol{y}) + \sum_{k,i} V_{ki} z_k f_i(\boldsymbol{y}) - A(\boldsymbol{z},\boldsymbol{V})) \ , \tag{5.16}$$

that specifies the likelihood of observable inputs $\boldsymbol{y}$, given a certain network state $\boldsymbol{z}$. This defines the generative model $p(\boldsymbol{y},\boldsymbol{z}|\boldsymbol{V}) = p(\boldsymbol{z})\, p(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{V})$.

We map this probabilistic model to the spiking network and define that for every k and every point in time t the variable $z_k(t)$ has the value 1, if the corresponding neuron has fired within the time window $(t-\tau, t]$. In accordance with the neural sampling theory, in order for a spiking network to sample from the correct posterior $p(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{V}) \propto p(\boldsymbol{z})\, p(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{V})$ given the input $\boldsymbol{y}$, each neuron must compute in its membrane potential the log-odd (Buesing et al., 2011),

$$u_k = \log \frac{p(z_k=1|\boldsymbol{z}_{\setminus k},\boldsymbol{V})}{p(z_k=0|\boldsymbol{z}_{\setminus k},\boldsymbol{V})} = \underbrace{\sum_i V_{ki} f_i(\boldsymbol{y})}_{\text{feedforward drive}} \underbrace{-A_k(\boldsymbol{V}) + \hat{b}_k}_{\text{intr. excitability}} + \sum_{j\neq k} \underbrace{(-A_{kj}(\boldsymbol{V}) + \hat{W}_{kj})}_{\text{recurrent weight}} z_j - \dots \tag{5.17}$$

where $\boldsymbol{z}_{\setminus k} = (z_1,\dots,z_{k-1},z_{k+1},\dots z_K)^{\mathsf{T}}$. The $A_k, A_{kj},\dots$ are given by the decomposition of $A(\boldsymbol{z},\boldsymbol{V})$ along the binary combinations of $\boldsymbol{z}$ as,

$$A(\boldsymbol{z},\boldsymbol{V}) = A_0(\boldsymbol{V}) + \sum_k z_k A_k(\boldsymbol{V}) + \frac{1}{2}\sum_{j\neq k} z_k z_j A_{kj}(\boldsymbol{V}) + \dots \tag{5.18}$$

Note, that we do not aim at this point to give learning rules for the prior parameters $\hat{b}_k$ and $\hat{W}_{kj}$. Instead we proceed as in the last section and specify a-priori desired properties of the average network response under the input distribution $p^*(\boldsymbol{y})$,

$$c_{kj} = \langle z_k z_j \rangle_{p^*(\boldsymbol{y})q(\boldsymbol{z}|\boldsymbol{y})} \qquad \text{and} \qquad m_k = \langle z_k \rangle_{p^*(\boldsymbol{y})q(\boldsymbol{z}|\boldsymbol{y})} \ . \tag{5.19}$$

Let us explore some illustrative configurations for m_k and c_{kj}. One obvious choice is closely related to the goal of maximizing the entropy of the output code by fixing $\langle z_k \rangle$ to $\frac{1}{K}$ and $\langle z_k z_j \rangle$ to $\langle z_k \rangle \langle z_j \rangle = \frac{1}{K^2}$, thus enforcing second order correlations to be zero. Another intuitive choice would be to set all $\langle z_k z_j \rangle$ very close to zero, which excludes that two neurons can be active simultaneously and thus recovers the function of a WTA. It is further conceivable to assign positive correlation targets to groups of neurons, thereby creating populations with redundant codes. Finally, with a topographical organization of neurons in mind, all three basic ideas sketched above might be combined: one could assign positive correlations to neighboring neurons in order to create local cooperative populations, mutual exclusion at intermediate distance, and zero correlation targets between distant neurons.

With this in mind, we can formulate the goal of learning for the network in the context of EM with posterior constraints: we constrain the E-step such that the

average posterior fulfills the chosen targets, and adapt the forward weights $\boldsymbol{V}$ in the M-step according to (5.6). Analogous to the first-order case, the variational solution of the E-step under these constraints takes the form,

$$q_{\boldsymbol{\beta},\boldsymbol{\omega}}(\boldsymbol{z}|\boldsymbol{y}) \propto p(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{V}) \cdot \exp\left(\sum_k \beta_k z_k + \frac{1}{2}\sum_{j\neq k} \omega_{kj} z_k z_j\right) \quad , \tag{5.20}$$

with symmetric $\omega_{kl} = \omega_{lk}$ as variational parameters. A neural sampling network $\mathcal{N}$ with input weights V_{ki} will sample from $q_{\boldsymbol{\beta},\boldsymbol{\omega}}$ if the intrinsic excitabilities are set to $b_k = -A_k + \hat{b}_k + \beta_k$, and the symmetric recurrent synaptic weights to $W_{kj} = -A_{kj} + \hat{W}_{kj} + \omega_{kj}$. The variational parameters $\boldsymbol{\beta}, \boldsymbol{\omega}$ (and hence also $\boldsymbol{b}, \boldsymbol{W}$) which optimize the dual problem $\Psi(\boldsymbol{b}, \boldsymbol{\omega})$ are uniquely defined and can be found iteratively via gradient ascent. Analogous to the last section, this yields the intrinsic plasticity rule (5.12) for b_k. In addition, we obtain for the recurrent synapses W_{kj},

$$\Delta W_{kj} \propto c_{kj} - \langle z_k z_j \rangle_{p^*(\boldsymbol{y})q(\boldsymbol{z}|\boldsymbol{y})} \quad , \tag{5.21}$$

which translates to an anti-Hebbian spike-timing dependent plasticity rule in the network implementation.

For any concrete instantiation of $f_0(\boldsymbol{y})$, $f_i(\boldsymbol{y})$ and $A(\boldsymbol{z}, \boldsymbol{V})$ in (5.16) it is possible to derive learning rules for V_{ki} for the M-step via $\partial_{V_{ki}} F(\boldsymbol{V}, q)$. Of course not all models entail local synaptic learning rules. In particular it might be necessary to assume conditional independence of the inputs $\boldsymbol{y}$ given the network state $\boldsymbol{z}$, i.e., $p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{V}) = \prod_i p(y_i|\boldsymbol{z}, \boldsymbol{V})$. Furthermore, in order to fulfill the neural computability condition (5.17) for neural sampling (Buesing et al., 2011) with a recurrent network of point neurons, it might be necessary to choose $A(\boldsymbol{z}, \boldsymbol{V})$ such that terms of order higher than 2 vanish in the decomposition. This can be shown to hold, for example, in a model with conditionally independent Gaussian distributed inputs y_i. It is ongoing work to find further biologically realistic network models in the sense of this theory and to assess their computational capabilities through computer experiments.

5.4 Discussion

Complex and non-local computations, which appear during probabilistic inference and learning, arguably constitute one of the cardinal challenges in the development of biologically realistic Bayesian spiking network models. In this paper we have introduced homeostatic plasticity, which to the best of our knowledge had not been considered before in the context of EM in spiking networks, as a theoretically grounded approach to stabilize and facilitate learning in a large class of network models. Our theory complements previously proposed neural mechanisms and provides, in particular, a simple and biologically realistic alternative to the assumptions

on the input distribution made in (Nessler et al., 2010) and (Keck et al., 2012). Indeed, our results challenge the hypothesis of (Keck et al., 2012) that feedforward inhibition is critical for correctly learning the structure of the data with biologically plausible plasticity rules. More generally, it turns out that the enforcement of a balancing posterior constraint often simplifies inference in recurrent spiking networks by eliminating nontrivial computations. Our results suggest a crucial role of homeostatic plasticity in the Bayesian brain: to constrain activity patterns in cortex to assist the autonomous optimization of an internal model of the environment.

5.5 Acknowledgments

This chapter is based on a joint work with Johannes Bill (JB) and Bernhard Nessler (BN) which was published and presented at NIPS 2012 ("Homeostatic plasticity in Bayesian spiking networks as Expectation Maximization with posterior constraints"). The theory was developed by SH and JB, with contributions from BN regarding the theoretical extension to recurrent stochastic networks. The computer experiments were designed by SH, JB, and were conducted by JB. The manuscript was written by SH, JB and BN.

Appendix A

List of Publications

Peer-Reviewed Journal and Conference Papers

1. Habenschuss, S., Bill, J., & Nessler, B. (2012) Homeostatic plasticity in Bayesian spiking networks as Expectation Maximization with posterior constraints. *Proceedings of NIPS 2012: Advances in Neural Information Processing Systems 25*, 782–790.

2. Habenschuss, S., Puhr, H., & Maass, W. (2013) Emergence of Optimal Decoding of Population Codes through STDP. *Neural Computation, 25*(6), 1371–1407.

3. Habenschuss, S., Jonke, Z., & Maass, W. (2013) Stochastic Computations in Cortical Microcircuit Models. *PLoS Computational Biology*, 9(11): e1003311. doi:10.1371/journal.pcbi.1003311.

Conference Abstracts

4. Buesing, L., Bill, J., Nessler, B., Habenschuss, S., & Maass, W. (2010) Emergence of Bayesian computation in generic motifs of cortical microcircuits. In *40th Annual Conference of the Society for Neuroscience*.

5. Habenschuss, S., Puhr, H., & Maass, W. (2011) Emergence of Optimal Population Decoding through STDP. In *41st Annual Conference of the Society for Neuroscience*.

6. Jonke, Z., Habenschuss, S., Legenstein, R., & Maass, W. (2012) Improved feature extraction by pyramidal cells through relaxed lateral inhibition. In *42nd Annual Conference of the Society for Neuroscience*.

7. Habenschuss, S., Jonke, Z., & Maass, W. (2013) Networks of spiking neurons with noise have an inherent capability to generate heuristic solutions to complex constraint satisfaction problems. In *43rd Annual Conference of the Society for Neuroscience*.

8. Maass, W., Habenschuss, S., & Jonke, Z. (2013) Neural circuits naturally encode probability distributions over network states, and over trajectories of network states. In *43rd Annual Conference of the Society for Neuroscience*.

9. Bill, J., Buesing, L., Habenschuss, S., Nessler, B., Legenstein, R., & Maass, W. (2013) Local inhibition facilitates synaptic learning in spatially extended Bayesian spiking networks. In *43rd Annual Conference of the Society for Neuroscience*.

Comments and Contributions to Papers

The paper *Stochastic Computations in Cortical Microcircuit Models* is a joint work with Zeno Jonke (ZJ) and Wolfgang Maass (WM). The initial "inverse problem" idea, to consider under what circumstances the concept of a stationary distribution may be applicable in more general networks of spiking neurons than those previously considered, came from WM who also supervised the project. I developed the theory of the paper. The broad strokes of paper organization and design of computer experiments were then conceived in a series of animated discussions, in which all three authors SH, ZJ and WM, were involved equally. ZJ developed the software simulation framework and the analysis tools and conducted the computer experiments. The simulation framework was based on NEVESIM, an event based simulator for networks of spiking neurons written by Dejan Pecevski. The figures were developed and designed by SH, ZJ and WM. The main text was written by SH and WM.

The paper manuscript *Networks of Spiking Neurons with Noise can Solve Hard Computational Problems* (not listed above) is currently in preparation for a journal submission and is a joint work with Zeno Jonke and Wolfgang Maass. The main idea and structure of this work as well as the design of computer experiments were conceived in close collaboration of all three authors, SH, ZJ, and WM. I developed the theory of the paper. ZJ contributed the development of the software framework and conducted the computer experiments. The simulation framework was based on the NEVESIM simulator by Dejan Pecevski. The main text was written by SH and WM. The figures were designed by SH, ZJ and WM.

The paper *Emergence of Optimal Decoding of Population Codes through STDP* is a joint work with Helmut Puhr (HP) and Wolfgang Maass. The idea of the paper was conceived by WM. The theory was developed by SH. The computer experiments were designed by SH, HP and WM. The software framework was written and the simulations were conducted by HP. The figures were designed by SH, HP and WM. The manuscript was written by SH, under supervision of WM.

The paper *Homeostatic plasticity in Bayesian spiking networks as Expectation Maximization with posterior constraints* is a joint work with Johannes Bill (JB) and Bernhard Nessler (BN). The theory was developed by SH and JB, with contributions from BN regarding the theoretical extension to recurrent stochastic networks. The computer experiments were designed by SH, JB, and were conducted by JB. The manuscript was written by SH, JB and BN.

Appendix B

Appendix to Chapter 2: Stochastic Computations in Cortical Microcircuit Models

Contents

B.1 Network states and distributions of network states

Markov states

The Markov state $y_M(t)$ (or more explicitly, $y_{M:\Theta}(t)$) of a network at time t is defined here as the recent history of spike times of all neurons in the network within the period $(t-\Theta, t]$. The term "Markov" refers to the fact that, under mild conditions and for a sufficiently long window Θ, the network dynamics of a neural circuit after time t becomes independent of the network activity at times $\leq t-\Theta$, given the Markov state $y_M(t)$ and the external input x. Hence, the network dynamics has the Markov property with respect to this state definition.

For each neuron $k \in 1 \dots K$ in a neural circuit a spike history of length Θ is defined as the list of spike times emitted by neuron k within the window $(t-\Theta, t]$.

Spike times are counted relative to the beginning of the window at $t - \Theta$. If m is the number of spikes within $(t - \Theta, t]$ for neuron k, then the list takes the form,

$$y_M^k(t) \equiv (y^{k,1}(t), \ldots, y^{k,m}(t)) \in \mathbb{R}^m \ , \tag{B.1}$$

where $0 < y^{k,1}(t) < \cdots < y^{k,m}(t) \leq \Theta$.

We denote the space of all possible network states of length Θ by S_Θ or, when unambiguous, simply by S. Note that this definition is equivalent to the state definition in (Borovkov et al., 2012), to which the interested reader is referred for further formal details (e.g. the associated σ-algebra $\mathcal{S}$ of the state space S).

Scope of theoretical results: Required properties of the network and neuronal noise models

We study general theoretical properties of stochastic spiking circuit models, driven by some external, possibly vector-valued, input $x(t)$, which could represent for example input rates in a set of input neurons or injected input currents. Formally, the input sequence can assume values from any state space Q; a concrete example is vector-valued input with $Q = \mathbb{R}^N$, where N is the number of input dimensions.

We consider in this article two different noise models for a neuron: In noise model I, the spike generation is directly modeled as a stochastic process. All network dynamics, including axonal delays, synaptic transmission, short-term synaptic dynamics, dendritic interactions, integration of input at the soma, etc. can be modeled by a function which maps the Markov state (which includes the recent spike history of the neuron itself) onto an instantaneous spiking probability. This model is highly flexible and may account for various types of neuronal noise. In the more specific noise model II, the firing mechanism of the neuron is assumed to be deterministic, and noise enters its dynamics through stochastic vesicle release at afferent synaptic inputs. Also combinations of noise models I and II in the same neuron and circuit can be assumed for our theoretical results, for example neurons with a generic stochastic spiking mechanism which possess in addition stochastic synapses, or mixtures of neurons from model I and II in the same circuit.

In noise model I, the instantaneous spiking probability of neuron k at time t is given by,

$$\lim_{\delta t \to 0} \frac{1}{\delta t} \cdot p(\text{neuron } k \text{ fires in } (t, t + \delta t)) = \rho_k(t) \ . \tag{B.2}$$

This instantaneous firing rate $\rho_k(t) = f(y_M(t))$ at time t is assumed to be bounded and completely determined by the network's current Markov state $y_{M:\Theta}(t)$, for some sufficiently large Θ. More precisely, the following four assumptions are made for noise model I:

*A*1 **Spikes are individual events:** We assume that,

$$\lim_{\delta t \to 0} \frac{1}{\delta t} \cdot p(\text{more than one neuron fires in } (t, t + \delta t)) = 0 \ , \tag{B.3}$$

which is, for example, fulfilled if each neuron has some independent source of stochasticity.

*A*2 **Bounded rates:** The instantaneous firing rates are bounded from above: $0 \leq \rho_k(t) \leq \hat{\rho}_k$ for some $\hat{\rho}_k < \infty$. The ensuing upper bound on the total network firing rate is denoted by $\hat{\rho}$, i.e. $0 \leq \sum_{k=1}^{K} \rho_k(t) \leq \hat{\rho}$. It is assumed that instantaneous rates are bounded at any time, and in the presence of any input $x(t)$.

*A*3 **Bounded memory:** The firing rates $\rho_k(t)$ at time t depend on the network's past activity only through the history of recent spikes in a finite window $(t - \underline{\Theta}, t]$ of length $\underline{\Theta}$. Hence, the *direct* effect of a spike at time t on future firing rates of all neurons is limited to a bounded "memory period", $[t, t + \underline{\Theta})$. This bounded memory period $\underline{\Theta}$ can be understood as a lower bound for Θ during the subsequent convergence proofs (since smaller Θ would violate the Markov property). In addition to this bounded-memory dependence on network spikes, $\rho_k(t)$ may depend on the current input $x(t)$ in any manner consistent with *A*2.

*A*4 **Time-homogeneity:** The functional mapping from recent spikes and/or input signals $x(t)$ to instantaneous firing rates $\rho_k(t)$ does not change over time. In particular, we do not consider long-term plasticity of synaptic weights and/or excitabilities in this work.

Assumptions $A2-A4$ can be summarized as follows: Let $x(t) \in Q$ and $y_{M:\underline{\Theta}}(t) \in S$ be the trajectories of input and network states as defined above. Then there exists a memory constant $\underline{\Theta}$ and rate bounds $0 \leq \hat{\rho}_k < \infty$, such that for each neuron k there exists a function $f_k : Q \times S \to [0, \hat{\rho}_k]$, where $\rho_k(t) = f_k(x(t), y_{M:\underline{\Theta}}(t))$ for all t. The function f_k is time-invariant but otherwise unconstrained, and can capture complex dynamical effects such as non-linear dendritic interactions between synaptic inputs or short-term plasticity of synapses.

The input signal $x(t)$ can formally represent any variable which exerts some arbitrary influence on the instantaneous network dynamics (the neuronal firing functions f_k). In the simplest case, $x(t)$ could be a vector of firing rates controlling the spiking behavior of a set of N input neurons i, such that $f_i(x(t), y_{M:\underline{\Theta}}(t)) = x_i(t)$ in these neurons. In this case (which we focused on in the main text), input neurons are formally considered part of the circuit C. Note that in principle, $x(t)$ could also represent the strength of currents which are injected into a subset of neurons in the network C, or the recent spiking history of a set of external input neurons ("input Markov states"). If the input comprises rates or currents, these can be either fixed (e.g. fixed input firing rates) or dynamically changing (in particular rates which are either subject to stochastic ergodic dynamics, or periodically changing rates). Below convergence proofs will be provided for both fixed and dynamic input conditions. If the input is defined in terms of input Markov states, the dynamic input analysis is applicable under conditions described further below.

In noise model II the basic stochastic event is a synaptic vesicle release (in noise model I it is a spike). Accordingly, the Markov state $y_M(t)$ of a network in noise model II is defined as the list of vesicle release times for each synaptic release site in the network (instead of spike timings for each neuron). We assume here that each synaptic release site releases at a given instance t at most one vesicle filled with neurotransmitters. But a synaptic connection between two neurons may consist of multiple synaptic release sites (see (Lisman et al., 2007; Branco and Staras, 2009) and (Borst, 2010) for reviews). Instead of expressing the network dynamics through an instantaneous firing probability function for each neuron k, $\rho_k(t) = f_k(x(t), y_{M:\underline{\Theta}}(t))$ (noise model I), for noise model II the network dynamics is expressed in terms of instantaneous release probabilities for each synapse k: $\psi_k(t) = g_k(x(t), y_{M:\underline{\Theta}}(t))$. Similar to noise model I, it is assumed that there exists a window length $\underline{\Theta}$, such that the dynamics of vesicle release at time t is fully determined by the timing of previous vesicle releases within $(t - \underline{\Theta}, t]$, and hence can be expressed in terms of a corresponding variation of the definition of a Markov state $y_{M:\underline{\Theta}}(t)$. The same framework of assumptions applies as in noise model I: vesicle releases are individual events, and the functions g_k are assumed to be bounded from above by rate constants $\hat{\psi}_k < \infty$.

Combinations of noise model I and II are also possible. In this case, the Markov state $y_M(t)$ may contain both spike times and vesicle release times. The assumptions of noise model I/II described above apply to the corresponding stochastic neurons and vesicle releases, respectively. Altogether, note that all three types of networks (based on model I, II and mixtures of the two) are based on a common framework of definitions and assumptions: in all cases the dynamics is described in terms of stochastic components (neurons, synapses) which generate point events (spikes/vesicle releases) according to instantaneous probabilities which depend on the recent event history of the network.

Convergence of state distributions

Below, proofs for the existence and uniqueness of stationary distributions of network states for the considered network models are given. Furthermore, bounds on the convergence speed to this stationary distribution are provided. To obtain a comprehensive picture, convergence is studied under three different input conditions: constant, stochastic, and periodic input. All proofs are described in detail for noise model I. The results transfer in a straightforward manner to noise model II and mixtures of these two models, since the same framework of assumptions applies to all cases.

Network dynamics as a Markov process

We view the simulation of a cortical microcircuit model, under a given input condition and starting from a given initial network state, as a random experiment.

Formally, we denote the set of all possible outcomes in this random experiment by Ω, the set of all considered *events* by $\mathcal{F}$ (i.e. a σ-algebra on Ω), and the probability measure which assigns a probability to each event in $\mathcal{F}$ by $\mathbb{P}$. An outcome is the result of a single run of the network. An outcome is associated with an assignment of particular values to all defined random variables. An event is a set of outcomes, for example the set of all outcomes in which neuron 7 spikes within the first 200 milliseconds of the experiment. Suppose X is a random variable with some state space $(R, \mathcal{R})$, i.e. X assumes values in R, and $\mathcal{R}$ is a set of events on the space R. Formally, such a random variable X is defined as a map $X : \Omega \to R$, which assigns a value $x \in R$ to every possible outcome $\omega \in \Omega$. To denote the probability that the random variable X assumes some value in the set $B \in \mathcal{R}$, we define the short-hand $\mathbb{P}_X(B) := \mathbb{P}(X \in B)$. Furthermore, if Y is another random variable we use the notation $\mathbb{P}_{X|Y=y}(B) := \mathbb{P}(X \in B|Y = y)$ for conditional probabilities, and write even shorter, when unambiguous, $\mathbb{P}_{X|y}(B)$. The base probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is assumed to be rich enough such that all random variables which are needed in the following exist.

We define the index set of time $T = \{t \in \mathbb{R} : t \geq 0\}$, and the stochastic process $(Y_t,\ t \in T)$, as a description of the stochastic evolution of Markov states of a network C for $t \geq 0$. For each time $t \in T$ we define a random variable Y_t (also written $Y(t)$) representing the Markov state of the network at time t. Y_t takes values on the state space S of all possible Markov states of some fixed duration Θ. We denote by $\mathcal{S}$ the σ-algebra associated with S. The assumptions on the network described in the previous section imply that the process has the Markov property for Markov states of any length $\Theta \geq \underline{\Theta}$, since the future evolution of the process is then entirely independent of the past, given the current Markov state. For the subsequent proofs, we therefore assume some $\Theta \geq \underline{\Theta}$. We also define a random variable Y of *entire sample paths* on the measurable space $(S^T, \mathcal{S}^T)$, i.e. a map $Y : \Omega \to S^T$. Realizations of Y are sample paths (or trajectories), i.e. functions $y_M(t),\ \ t \in T$, taking values in S. Since realizations of Y are functions, Y can be thought of as a random function.

For subsequent proofs the following definition of a *transition probability kernel* is essential: A transition probability kernel $\mathcal{P}$ on a measurable state space $(S, \mathcal{S})$ is a function $\mathcal{P} : S \times \mathcal{S} \to [0, 1]$, which assigns a probability to the transition from any point $x \in S$ to any set $B \in \mathcal{S}$. More precisely, if one fixes a particular "initial state" $x \in S$, then $\mathcal{P}(x, B) \equiv \mathcal{P}_x(B)$ is a probability measure in its target argument B, corresponding to the result of applying the transition kernel $\mathcal{P}$ to x (in addition, for each event $B \in \mathcal{S}$ in the target space, $\mathcal{P}(x, B)$ is $\mathcal{S}$-measurable in its source argument). Stochastic transition matrices of Markov chains are, e.g., transition probability kernels.

Here we write $\mathcal{P}^{s:t}$ for the transition probability kernel corresponding to progression of the state of the network C from time s to $s + t$, i.e.,

$$\mathcal{P}^{s:t}(y_0, B) := \mathbb{P}(Y(s + t) \in B \mid Y(s) = y_0) \ . \tag{B.4}$$

We further define the shorthand $\mathcal{P}^t = \mathcal{P}^{0:t}$ for the progression of duration t starting from initial time $s = 0$. Transition kernels can also be applied to probability measures ϕ of initial states y_0 (as opposed to single initial states y_0). We will write $\mathcal{P}^{s:t}\phi$ to denote the result of applying the kernel $\mathcal{P}^{s:t}$ to an initial probability measure ϕ. The result $\mathcal{P}^{s:t}\phi$ is again a probability measure, assigning a probability to any event $B \in \mathcal{S}$ on the state space according to:

$$(\mathcal{P}^{s:t}\phi)(B) := \int_S \mathcal{P}^{s:t}(y_0, B)\, d\phi(y_0) \quad , \tag{B.5}$$

Since $\mathcal{P}^{s:t}\phi$ is again a probability measure on the state space $(S, \mathcal{S})$, transition kernels can be applied sequentially. Note that due to the Markov property one has, $\mathcal{P}^{r:(t_1+t_2)}\,\phi = \mathcal{P}^{(r+t_1):t_2}\,\mathcal{P}^{r:t_1}\,\phi$ for $s \geq 0,\ \ t_1, t_2 > 0$.

Stochastic network dynamics is contracting

Before studying specific input conditions, a few basic key properties of the network dynamics Y are developed. Let $\mathcal{P}^{s:t}$ be the transition probability kernel corresponding to progression of the network C from time s to $s + t$. For the proofs below, transitions to the *resting state*, $Y(s+t) = \mathbf{0}$, will be of particular importance. The resting state $\mathbf{0}$ is defined as the "empty" Markov state in which no spikes occurred within the last Θ time units. The first key observation is the following Proposition:

Proposition 1 *Consider the probability $\mathcal{P}^{s:\Theta}(y_0, \mathbf{0})$, that the process Y will be in the resting state $\mathbf{0}$ at time $s + \Theta$, starting from some initial state $y_0 \in S$ at time s. This "return probability" to the resting state is bounded from below by,*

$$\mathcal{P}^{s:\Theta}(y_0, \mathbf{0}) \geq \epsilon^{\Theta} \quad , \tag{B.6}$$

where $\epsilon := e^{-\hat{\rho}}$. This holds regardless of the input trajectory $x(t)$ driving the network.

The proposition follows directly from the fact that $\hat{\rho}$ bounds the sum of all instantaneous firing rates in the network. Hence with at least probability $e^{-\hat{\rho}\Theta} = \epsilon^{\Theta}$ no neuron fires within Θ time units (cf. (Borovkov et al., 2012)). In technical terms, this implies that the stochastic kernel corresponding to a duration of length Θ fulfills the Doeblin condition (Doeblin, 1937) – a property which is highly useful for proving convergence and ergodicity results.

Proposition 1 entails a central contraction property of stochastic networks of spiking neurons C, which holds in the presence of any input trajectory $x(t)$, and forms the basis for several subsequent proofs. The following definitions are essential: We will measure below the difference between any two probability distributions ϕ_1 and ϕ_2 in terms of the total variation $\|\cdot\|$ of the signed measure $\mu = \phi_1 - \phi_2$. Any such signed measure μ can be expressed in terms of its non-negative and non-positive components, $\mu = \mu^+ - \mu^-$, where μ^+ and μ^- are both non-negative

measures (but in general no probability measures). The total variation of a signed measure μ on a measurable space $(X, \mathcal{X})$ is defined as $\|\mu\| = \mu^+(X) + \mu^-(X)$, i.e. the total mass of its positive and negative components. According to this definition, $\|\mu\| = \|\mu^+\| + \|\mu^-\|$.

Lemma 1 (Contraction Lemma) *The following strict contraction property holds for the Markov process Y, for any $\Theta \geq \underline{\Theta}$, and for any initial probability measures ϕ_1 and ϕ_2 at any time $s \geq 0$:*

$$\|\mathcal{P}^{s:\Theta}\phi_1 - \mathcal{P}^{s:\Theta}\phi_2\| \leq (1 - \epsilon^\Theta) \cdot \|\phi_1 - \phi_2\| \ . \tag{B.7}$$

In words: applying the dynamics of the network C for Θ time units is guaranteed to reduce the distance between any two initial distributions ϕ_1 and ϕ_2 of network states by a factor $1 - \epsilon^\Theta$.

Proof: Define the auxiliary measure ν_0 as zero everywhere outside $\mathbf{0}$, and $\nu_0(\mathbf{0}) = \epsilon^\Theta$. Rewrite $\phi_1 - \phi_2 = \mu = \mu^+ - \mu^-$ in terms of the non-negative measures μ^+ and μ^-, such that

$$\|\phi_1 - \phi_2\| = \|\mu^+\| + \|\mu^-\| \ , \tag{B.8}$$

and note that $\|\phi_1\| = \|\phi_2\| = 1$ implies that $\|\mu^+\| = \|\mu^-\|$. Then

$$\begin{aligned}
\|\mathcal{P}^{s:\Theta}\phi_1 - \mathcal{P}^{s:\Theta}\phi_2\| &= \|\mathcal{P}^{s:\Theta}\mu^+ - \mathcal{P}^{s:\Theta}\mu^-\| && \text{(B.9)}\\
&= \|(\mathcal{P}^{s:\Theta}\mu^+ - \|\mu^+\| \cdot \nu_0) - (\mathcal{P}^{s:\Theta}\mu^- - \|\mu^-\| \cdot \nu_0)\| && \text{(B.10)}\\
&\leq \|\underbrace{\mathcal{P}^{s:\Theta}\mu^+ - \|\mu^+\| \cdot \nu_0}_{\geq 0 \text{ for all events } B \in \mathcal{S}}\| \ + \ \|\underbrace{\mathcal{P}^{s:\Theta}\mu^- - \|\mu^-\| \cdot \nu_0}_{\geq 0 \text{ for all events } B \in \mathcal{S}}\| && \text{(B.11)}\\
&= \|\mathcal{P}^{s:\Theta}\mu^+\| - \|\mu^+\| \cdot \|\nu_0\| \ + \ \|\mathcal{P}^{s:\Theta}\mu^-\| - \|\mu^-\| \cdot \|\nu_0\| && \text{(B.12)}\\
&= (1 - \|\nu_0\|) \cdot \|\mu^+\| \ + \ (1 - \|\nu_0\|) \cdot \|\mu^-\| && \text{(B.13)}\\
&= (1 - \epsilon^\Theta) \cdot \big(\|\mu^+\| + \|\mu^-\|\big) && \text{(B.14)}\\
&= (1 - \epsilon^\Theta) \cdot \|\phi_1 - \phi_2\| \ . && \text{(B.15)}
\end{aligned}$$

The equality in (B.9) follows from linearity of transition probability kernels. The transition to (B.11) is an application of the triangle inequality. The transition to (B.12) uses the fact that both $\mathcal{P}^{s:\Theta}\mu^+ - \|\mu^+\| \cdot \nu_0$ and $\mathcal{P}^{s:\Theta}\mu^- - \|\mu^-\| \cdot \nu_0$ are non-negative: this follows from Proposition 1, which ensures that the measure $\mathcal{P}^{s:\Theta}\mu^+$ has at least mass $\|\mu^+\| \cdot \epsilon^\Theta$ at the resting state $\mathbf{0}$ and, hence, for any (non-negative) measure ν,

$$\mathcal{P}^{s:\Theta}\nu \geq \|\nu\| \cdot \nu_0 \ . \tag{B.16}$$

Finally, note that (B.13) uses a general property of transition probability kernels $\mathcal{P}$, which ensures that $\|\mathcal{P}\nu\| = \|\nu\|$, for any non-negative measure ν. □

Note that the above Contraction Lemma which holds for spiking neural networks has some similarities to Lemma 1 in (Maass and Sontag, 1999) who analyzed artificial analog neural networks in discrete time.

B.2 Proof of Theorem 1

We divided the precise formulation of Theorem 1 into two Lemmata: Lemma 2 is a precise formulation for the case where inputs are fixed (e.g. fixed input rates). Lemma 3 in the next section corresponds to the case where input rates are controlled by a Markov process. The precise assumptions on the network model required for both Lemmata are described above (see "Scope of theoretical results").

Proof of Theorem 1 for fixed input rates

Here we assume that the vector of inputs $x(t)$ provided to the network is kept fixed during a trial. Concretely, this is for example the case if there is a set of input neurons whose rates are fixed. In this case, $x(t)$ is a vector of input rates, which remains constant over time. The input neurons are formally considered part of the network in this case. Alternatively, a constant x could correspond to constant currents which are injected into a subset of neurons.

Under constant input conditions, $x(t) \equiv x$, the dynamics of the process is time-homogeneous: the transition probability kernels are invariant to time-shifts, i.e.

$$\mathcal{P}^{s_1:t}\phi = \mathcal{P}^{s_2:t}\phi, \quad s_1, s_2 \geq 0,\ t > 0 \ . \tag{B.17}$$

Lemma 2 *Let $x(t) \equiv x$. Then the Markov process Y has a unique stationary distribution π, to which it converges exponentially fast,*

$$\|\mathcal{P}^t(y_0, \cdot) - \pi\| \leq 2 \cdot (1 - \epsilon^{\Theta})^{t-1} \ , \quad t \geq 0 \ , \tag{B.18}$$

from any initial Markov state $y_0 \in S$.

Proof: Y is clearly non-explosive, aperiodic and stochastically continuous (cf. (Borovkov et al., 2012)). To prove exponential ergodicity it thus suffices to show that some skeleton chain is geometrically ergodic (see for example Theorem 18.1 in (Borovkov, 1998)). The skeleton chain $Y_{\Theta n},\ n \in \mathbb{N}$, with transition probability kernel $\mathcal{P}^{\Theta}$ is aperiodic and irreducible and hence has a unique stationary distribution π. Then, through recursive application of Lemma 1 with $\phi_2 = \pi$,

$$\|\mathcal{P}^{\Theta n}\phi_1 - \mathcal{P}^{\Theta n}\pi\| \leq (1 - \epsilon^{\Theta})^n \cdot \|\phi_1 - \pi\| \ , \tag{B.19}$$

$$\|\mathcal{P}^{\Theta n}\phi_1 - \pi\| \leq 2 \cdot (1 - \epsilon^{\Theta})^n \ , \tag{B.20}$$

proving geometric ergodicity of the skeleton chain, and thus exponential ergodicity of Y. The quantitative convergence bound follows from (B.20) by choosing a singleton y_0 as initial distribution, and using the general fact that for any transition probability kernel $\mathcal{P}$ and distributions ϕ_1 and ϕ_2,

$$\|\mathcal{P}\phi_1 - \mathcal{P}\phi_2\| \leq \|\phi_1 - \phi_2\| \ , \tag{B.21}$$

thus guaranteeing that the total variation distance does not (temporarily) grow between Θn and $\Theta(n+1)$. □

Lemma 2 provides a general ergodicity result for the considered class of stochastic spiking networks in the presence of fixed input rates x. The proof relies on two key properties of stochastic spiking networks: aperiodicity and irreducibility. These properties can be understood intuitively in the context of Figure 2.1H. If the intrinsic network dynamics was not aperiodic, for example, then one might be able to observe oscillating pattern frequencies over time (as in Figure 2.4C). Lemma 2 proves that this cannot occur in stochastic spiking networks as long as input rates are fixed. Oscillating pattern frequencies can indeed only emerge when input rates are themselves periodically changing (see Theorem 2 and Figure 2.4). If the network dynamics was not irreducible on the other hand, i.e. if there were network states which are unreachable from some other network states, then pattern frequencies could potentially be observed to converge to different fixed points for different initial states (e.g. the two lines in Figure 2.1H settling at different values). This cannot occur in stochastic spiking networks due to Proposition 1 which guarantees that the state space is connected through the resting state $\mathbf{0}$.

Note that, although aperiodicity and irreducibility are well known necessary and sufficient conditions for ergodicity in discrete time Markov chains on finite state spaces, they are not sufficient for exponential ergodicity in continuous time Markov processes on general state spaces (see (Down et al., 1995) for precise definitions of ϕ-irreducibility and aperiodicity for such processes). Additional conditions in this more complex case which ensure exponential ergodicity, such as nonexplosivity, stochastic continuity and geometric ergodicity of a skeleton chain, have also been taken into account in the proof for Lemma 2 (i.e. stochastic spiking networks also meet these additional criteria).

Lemma 2 constitutes a proof for Theorem 1 for fixed input rates x. In the main text we refer to the stationary distribution of the circuit C under fixed input x as $p_C(y|x)$. The proof above guarantees a stationary distribution for both Markov and simple states. In the main text y refers to the simple network state y_S if not stated otherwise.

Proof of Theorem 1 for input rates controlled by a Markov process

Fixed input assumptions may often hold for the external input $x(t)$, driving a stochastic computation in a neural system C, only approximately. Stochastic fluctuations on various spatial and temporal scales may be present in the input. In addition, inputs may have their own short-term stochastic dynamics: Imagine, for example, a visual scene of randomly moving dots. Despite the presence of such short-term dynamical features in the input, in many cases one may still suspect that network state distributions converge. Indeed, below we generalize the convergence results from the constant case to the quite large class of stochastic (and stochastically changing) inputs which are generated by a uniformly ergodic Markov

process. Uniform ergodicity is defined as exponential ergodicity (exponentially fast convergence to a unique stationary distribution) with convergence constants which apply uniformly to all initial states (Down et al., 1995) (this holds for example for the convergence constants in Lemma 2).

Let X be a time-homogeneous *input Markov process*, in the sense that the input trajectory $x(t)$ provided to the network C is itself generated randomly from a Markov process X. Let $(Q, \mathcal{Q})$ be the (measurable) state space of X. Then define a joint input/network Markov process Z on the state space $(Q \times S,\ \sigma(\mathcal{Q} \times \mathcal{S}))$, where $\sigma(\cdot)$ denotes the σ-algebra generated by $\cdot$. Further definitions for Z are analogous to those introduced for Y.

Lemma 3 *If the input process X is uniformly ergodic, then the joint Markov process Z has a unique stationary distribution $\hat{\pi}$ on the joint input/network state space, to which convergence occurs exponentially fast, i.e. there exist constants $C < \infty, \rho < 1$, such that*

$$\|\mathcal{P}^t(z_0, \cdot) - \hat{\pi}\| \leq C \cdot \rho^t \ , t \geq 0 \ , \tag{B.22}$$

for any initial state z_0 of the joint Markov process Z.

Proof: If X and Y were entirely independent processes (if X did not influence Y) then the joint process Z would automatically be exponentially ergodic if both X and Y are. Although in the present case Y is not independent of X, a weaker version of independence applies: the return probability to the resting state $Y(t) = \mathbf{0}$ during $(t - \Theta, t]$ is at least ϵ^Θ *regardless* of the input trajectory of X during that time. This property can be exploited to show that the distribution of hitting times to a joint resting state has an exponential bound. It follows that the joint process is exponentially ergodic. A detailed proof is given in the next section. □

The second part of Theorem 1 (exponentially fast convergence for the case of external input generated by an ergodic Markov process) follows from Lemma 3. Note that in the main text we slightly abuse the notation $p_C(y|x)$ for the dynamic case to indicate the stationary distribution over network states y, where x denotes a specific Markov process controlling the inputs.

Detailed proof of Lemma 3

We have split the proof of Lemma 3 into proofs of four auxiliary claims (Propositions 2-5). Consider the following variations of Proposition 1, which hold for the Markov process Z describing the joint dynamics of input and network states. Let $\{x(t)\}$ denote a particular input sequence defined for $t \geq 0$ (a realization of the input process X) and $y_0 \in S$ an initial network Markov state (with $\Theta \geq \underline{\Theta}$) at time $s \geq 0$. Then

$$\mathbb{P}(Y(s + \Theta) = \mathbf{0} \mid Y(s) = y_0, X = \{x(t)\}) \geq \epsilon^\Theta \ , \tag{B.23}$$

$$\mathbb{P}(Y(s + \Theta) = \mathbf{0} \mid X = \{x(t)\}) \geq \epsilon^\Theta \ . \tag{B.24}$$

It is easy to show that these properties, together with the fact that X is uniformly ergodic, ensure that Z is irreducible and aperiodic. Hence, to prove exponential ergodicity of Z it suffices to show that some skeleton chain is geometrically ergodic (Down et al., 1995). To that end, we will consider the skeleton chain $Z_{\Theta n},\ n \in \mathbb{N}$ and prove geometric ergodicity by showing that the hitting time distribution $\mathbb{P}_{\tau_C}(\tau_C)$ to a *small set* C on the joint state space $Q \times S$ of input and network states admits an exponential bound.

The *hitting time* τ_D to some set D on the input state space Q is defined as

$$\tau_D = \min\{n \in \mathbb{N}^+ : X_{\Theta n} \in D\} \ . \tag{B.25}$$

For notational convenience we abbreviate in the following $\tau = \tau_D$. Due to uniform ergodicity of X (which implies Harris recurrence (Down et al., 1995)), there exists some set D to which the hitting time τ is finite ($< \infty$) from any initial state, with probability one (Meyn and Tweedie, 1993). Furthermore, there exists according to (Down et al., 1995) a *small set* D and constants $\kappa > 1$ and $1 \leq V < \infty$, such that

$$\forall x_0 \in Q: \quad \mathbb{E}\left[\kappa^{\tau} \mid X(0) = x_0\right] < V \ . \tag{B.26}$$

This implies that there exists a small set D on the input state space Q which can not only be reached in finite time from any initial input state x_0, but for which the hitting time distribution to D has also finite mean and variance (and finite higher-order moments). At least one pair of constants κ and V which fulfills (B.26) is guaranteed to exist, but in fact the following Proposition shows that one can specify a particular desired bound on the right-hand side (for reasons which will become clear later), and find a matching λ on the left-hand side.

Proposition 2 *There exists a $\lambda > 1$, such that*

$$\forall x_0 \in Q: \quad \mathbb{E}\left[\lambda^{\tau} \mid X(0) = x_0\right] < (1 - \epsilon^{\Theta})^{-1/2}. \tag{B.27}$$

Proof: Define $v(\lambda) := \mathbb{E}\left[\lambda^{\tau} \mid X(0) = x_0\right]$. Let κ and V be any valid pair of constants which fulfills (B.26). The trivial case is $v(\kappa) < (1 - \epsilon^{\Theta})^{-1/2}$. In the remainder of the proof it is assumed that κ is "too large", such that $v(\kappa) \geq (1 - \epsilon^{\Theta})^{-1/2}$. By definition of the exponential function, for any $\lambda > 0$,

$$v(\lambda) = \mathbb{E}\left[\lambda^{\tau} | x_0\right] = \mathbb{E}\left[\sum_{n=0}^{\infty} \frac{(\log \lambda)^n \tau^n}{n!} | x_0\right] \tag{B.28}$$

$$= \sum_{\tau=0}^{\infty} \sum_{n=0}^{\infty} \mathbb{P}_{\tau | x_0}(\tau) \frac{(\log \lambda)^n \tau^n}{n!} \ . \tag{B.29}$$

By Tonelli's theorem, since all summands are non-negative, the order of the double sum can be exchanged:

$$v(\lambda) = \sum_{n=0}^{\infty} \sum_{\tau=0}^{\infty} \mathbb{P}_{\tau|x_0}(\tau) \frac{(\log \lambda)^n \tau^n}{n!} \tag{B.30}$$

$$= \sum_{n=0}^{\infty} \frac{(\log \lambda)^n}{n!} \mathbb{E}[\tau^n | x_0] \ . \tag{B.31}$$

Note that $\mathbb{E}[\tau^n|x_0]$ are the moments of the distribution $\mathbb{P}_{\tau|x_0}[\tau]$. By uniform ergodicity of X, all moments must exist, and in addition there exists a $\kappa > 1$ such that $v(\kappa) < \infty$. It is straightforward to see that the series then converges for all $1 \leq \lambda \leq \kappa$, such that $v(\lambda)$ is continuous on $[1, \kappa]$. Finally, since $v(1) = 1$ and $v(\kappa) \geq (1-\epsilon^{\Theta})^{-1/2}$, by the intermediate value theorem there exists some $1 < \lambda < \kappa$ such that $v(\lambda) = (1 + (1-\epsilon^{\Theta})^{-1/2})/2$. □

Denote by $\tau^{(m)}$ the time at which the skeleton chain $X_{\Theta n}$ visits the small set D for the m-th time:

$$\tau^{(m)} = \min \{n \in \mathbb{N}^+ : \exists\, n_1 < n_2 < \cdots < n_m \leq n \in \mathbb{N} : X_{\Theta \cdot n_k} \in D, k \in \{1, \ldots, m\}\}. \tag{B.32}$$

Furthermore, denote by $\delta^{(m)}$ the time between the $(m-1)$-th and m-th visit:

$$\delta^{(1)} := \tau^{(1)}, \tag{B.33}$$

$$\delta^{(m)} := \tau^{(m)} - \tau^{(m-1)}, \quad m > 1. \tag{B.34}$$

According to this definition, one can express the hitting time of degree m as $\tau^{(m)} = \sum_{k=1}^{m} \delta^{(k)}$. The following Proposition extends the exponential bound on the first hitting time to hitting times of higher degrees.

Proposition 3 *There exists a $\lambda > 1$, such that,*

$$\forall x_0 \in Q: \quad \mathbb{E}\left[\lambda^{\tau^{(m)}} \mid X(0) = x_0\right] < (1-\epsilon^{\Theta})^{-m/2} \ . \tag{B.35}$$

Proof:

$$\mathbb{E}\left[\lambda^{\tau^{(m)}} | y_0\right] = \int d\mathbb{P}_{\delta^{(1\ldots m)}|y_0}(\delta^{(1\ldots m)}) \cdot \lambda^{\sum_{k=1}^{m} \delta^{(k)}} \tag{B.36}$$

$$= \int d\mathbb{P}_{\delta^{(1)}|y_0}(\delta^{(1)}) \cdot \lambda^{\delta^{(1)}} \int d\mathbb{P}_{\delta^{(2)}|y_0,\delta^{(1)}}(\delta^{(2)}) \cdot \lambda^{\delta^{(2)}} \cdots$$

$$\cdots \int d\mathbb{P}_{\delta^{(m)}|y_0,\delta^{(1\ldots m-1)}}(\delta^{(m)}) \cdot \lambda^{\delta^{(m)}} \tag{B.37}$$

$$< \left[(1-\epsilon^{\Theta})^{-1/2}\right]^m \ . \tag{B.38}$$

□

Let τ_C be the hitting time to the small set $C = D \times \mathbf{0}$ on the joint state space $Q \times S$ of input and network states,

$$\tau_C = \min\{n \in \mathbb{N}^+ : X_{\Theta n} \in D, Y_{\Theta n} = \mathbf{0}\} \ . \tag{B.39}$$

Furthermore, let R be the number of visits to the small set D prior to and including time τ_C,

$$R = \max\{m \in \mathbb{N}^+ : \exists\, n_1 < n_2 < \cdots < n_m \leq \tau_C \in \mathbb{N} : X_{\Theta n_k} \in D,\ k \in \{1, \ldots, m\}\}. \tag{B.40}$$

Proposition 4 *For any input trajectory $x(t)$ and any initial network state $y_0 \in S$,*

$$\mathbb{P}(R = m \mid Y(0) = y_0, X = \{x(t)\}) \leq (1 - \epsilon^{\Theta})^{m-1} \ . \tag{B.41}$$

This follows from (B.23) and (B.24) which ensure that whenever the input process visits the small set D, there is also a small probability that the network is in the resting state.

Proposition 5 *There exists a $\lambda > 1$ and a constant $W < \infty$ such that,*

$$\forall z_0 \in (Q \times S): \quad \mathbb{E}\left[\lambda^{\tau_C} \mid Z(0) = z_0\right] < W \ . \tag{B.42}$$

Proof: Let $\boldsymbol{\tau} = (\tau^{(m)},\ m \in \mathbb{N}^+)$. Choose some λ which fulfills Proposition 3.

$$\mathbb{E}\left[\lambda^{\tau_C} | z_0\right] = \int d\mathbb{P}_{\boldsymbol{\tau}, R|z_0}(\boldsymbol{\tau}, m)\, \lambda^{\tau^{(m)}} \tag{B.43}$$

$$= \int d\mathbb{P}_{\boldsymbol{\tau}|z_0}(\boldsymbol{\tau}) \int d\mathbb{P}_{X|z_0,\boldsymbol{\tau}}(\{x(t)\}) \sum_{m=1}^{\infty} \mathbb{P}(R = m | y_0, \{x(t)\})\, \lambda^{\tau^{(m)}} \tag{B.44}$$

$$\leq \int d\mathbb{P}_{\boldsymbol{\tau}|z_0}(\boldsymbol{\tau}) \sum_{m=1}^{\infty} (1 - \epsilon^{\Theta})^{m-1} \lambda^{\tau^{(m)}} \tag{B.45}$$

$$= \int d\mathbb{P}_{\boldsymbol{\tau}|x_0}(\boldsymbol{\tau}) \sum_{m=1}^{\infty} (1 - \epsilon^{\Theta})^{m-1} \lambda^{\tau^{(m)}} \tag{B.46}$$

$$= \sum_{m=1}^{\infty} (1 - \epsilon^{\Theta})^{m-1} \int d\mathbb{P}_{\tau^{(m)}|x_0}(\tau^{(m)}) \cdot \lambda^{\tau^{(m)}} \tag{B.47}$$

$$< \sum_{m=1}^{\infty} (1 - \epsilon^{\Theta})^{m-1} (1 - \epsilon^{\Theta})^{-m/2} \tag{B.48}$$

$$= \sum_{m=1}^{\infty} (1 - \epsilon^{\Theta})^{(m/2)-1} =: W < \infty \ . \tag{B.49}$$

□

By Proposition 5, Z is exponentially ergodic (Down et al., 1995). This completes the proof of Lemma 3.

Distribution of trajectories of network states

The Markov states $y_{M:\Theta}(t)$ are segments of spiking trajectories of length Θ. Hence, all statements developed above apply to convergence of the distribution over these (short) spiking trajectories. If one is interested in the convergence of longer trajectories, the simplest option is to choose a larger Θ, since any finite $\Theta \geq \underline{\Theta}$ is admissible, and all convergence results readily extend to trajectories of any finite length. A limitation of this approach is that the quantitative convergence statements will suffer from making Θ too large, since convergence rates scale approximately with ϵ^{Θ} (and $\epsilon \ll 1$). Hence, in practice, empirical convergence tests are required to make statements about specific circuits.

B.3 Proof of Theorem 2

If the input sequence is periodic with period L, i.e. $x(t) = x(t+L)$ for all $t \geq 0$, then the Markov process Y will be time-periodic, in the sense that transition kernels are invariant to shifts which are multiples of the period L:

$$\mathcal{P}^{s:t}\phi = \mathcal{P}^{s+kL:t}\phi, \quad s \geq 0,\ t > 0,\ k \in \mathbb{N}\ . \tag{B.50}$$

This implies the following result, which is a more precise version of Theorem 2:

Lemma 4 *Under periodic input, i.e. $x(t) = x(t+L)$ for all $t \geq 0$ with some $L \geq \Theta$, the time-periodic Markov process Y with period L has a periodically stationary distribution $\tilde{\pi}_l$, to which convergence occurs exponentially fast from any initial state. In particular, for every $0 \leq l < L$ there exists a unique stationary distribution $\tilde{\pi}_l$ such that,*

$$\|\mathcal{P}^{l+Ln}(y_0, \cdot) - \tilde{\pi}_l\| \leq 2 \cdot (1 - \epsilon^{\Theta})^{\lfloor \frac{L}{\Theta} \rfloor \cdot n}\ , \quad n \in \mathbb{N}\ , \tag{B.51}$$

from any initial Markov state y_0.

Proof: For each $0 \leq l < L$ there exists a skeleton chain $Y_{l+Ln},\ n \in \mathbb{N}$, with transition probability kernel $\mathcal{P}^{l:L} = \mathcal{P}^{l+L:L} = \mathcal{P}^{l+2L:L} = \ldots$, which is time-homogeneous, irreducible, and aperiodic and thus has a unique stationary distribution $\tilde{\pi}_l$. An application of $\mathcal{P}^{l:L}$, which corresponds to a full period, decreases the total variation distance to $\tilde{\pi}_l$ by at least $(1 - \epsilon^{\Theta})^{\lfloor \frac{L}{\Theta} \rfloor}$:

$$\|\mathcal{P}^{l:L}\phi_1 - \tilde{\pi}_l\| = \|\mathcal{P}^{l:L}\phi_1 - \mathcal{P}^{l:L}\tilde{\pi}_l\| \tag{B.52}$$

$$\leq \|\mathcal{P}^{l:\Theta\lfloor \frac{L}{\Theta} \rfloor}\phi_1 - \mathcal{P}^{l:\Theta\lfloor \frac{L}{\Theta} \rfloor}\tilde{\pi}_l\| \tag{B.53}$$

$$\leq (1 - \epsilon^{\Theta})^{\lfloor \frac{L}{\Theta} \rfloor} \cdot \|\phi_1 - \tilde{\pi}_l\|\ . \tag{B.54}$$

The first inequality follows from the fact that applying the remaining $\mathcal{P}^{l+\Theta\lfloor\frac{L}{\Theta}\rfloor:L-\Theta\lfloor\frac{L}{\Theta}\rfloor}$ can only further decrease the total variation distance between the two distributions, according to (B.21). The second inequality is due to Lemma 1.

Lemma 4 then follows from recursive application of (B.52)-(B.54) for multiple periods, and choosing a singleton y_0 as initial distribution. □

In the main text, we use the notation $p_{C,l}(y|x)$ for a phase-specific stationary distribution, where x denotes a specific periodic input sequence.

B.4 Relation to previous theoretical work

Previous work on the question whether states of spiking neural networks might converge to a unique stationary distribution had focused on the case where neuronal integration of incoming spikes occurs in a linear fashion, i.e., linear subthreshold dynamics followed by a single output non-linearity (Brémaud and Massoulié, 1996; Borovkov et al., 2012). In addition these earlier publications did not allow for the experimentally observed short term dynamics of synapses. The earlier publication (Brémaud and Massoulié, 1996) had studied this question as a special case of the mathematical framework of non-linear Hawkes processes, a class of mutually exciting point processes (see also (Massoulié, 1998)). The authors had arrived for the more restricted type of neurons which they considered at exponential convergence guarantees under a similar set of assumptions as in this article, in particular bounded memory and bounded instantaneous firing rates (and these results can thus be seen as a special case of Theorem 1, for the case of constant external input). (Brémaud and Massoulié, 1996) also derived convergence results for linearly integrating neurons with unbounded memory dynamics under a different set of assumptions, in particular Lipschitz conditions on the output non-linearity and constraints on the effective connectivity matrix of the network. Whether such alternative set of assumptions can be found also in the context of non-linear integration of incoming spikes (needed e.g. for synaptic short-time dynamics or dendritic non-linearities) remains an open question.

The recent publication (Borovkov et al., 2012) also focused on neurons with linear sub-threshold dynamics followed by an output non-linearity (termed there nonlinear Poisson neurons) with static synapses, and extended the convergence results of (Brémaud and Massoulié, 1996) to networks with Hebbian learning mechanisms. In addition, an important methodological innovation by (Borovkov et al., 2012) was the introduction of spike history states (which are equivalent to the Markov states $y_M(t)$ in this article) which allowed them to study convergence in the framework of general Markov processes (in contrast to point processes in (Brémaud and Massoulié, 1996)). Theorem 1 in this article contains as a special case the convergence results of (Borovkov et al., 2012) for their Model I (non-linear Poisson neurons in the absence of Hebbian learning). We note that although (Borovkov et al., 2012) focused on neurons with linear sub-threshold dynamics (and required that firing

rates are strictly greater than 0), their method of proof for Model I could be readily extended to cover also non-linear sub-threshold dynamics to yield the first part of our Theorem 1 (the case where inputs have constant firing rates).

We are not aware of previous work that studied convergence in spiking networks with dynamic synapses, or in the presence of stochastic or periodic inputs (see the second part of Theorem 1 concerning Markov processes as input, and Theorem 2). We further note that our method of proof builds on a new and rather intuitive intermediate result, Lemma 1 (Contraction Lemma), which may be useful in its own right for two reasons. On the one hand it provides more direct insight into the mechanisms responsible for convergence (the contraction between any two distributions). On the other hand, it holds regardless of the input trajectory $x(t)$, and hence has in fact an even larger scope of applicability than Theorem 1 and 2. Hence, Lemma 1 could be, for example, applied to study non-stationary evolutions of state distributions in response to arbitrary input trajectories.

B.5 Extracting knowledge from internally stored distributions

A key advantage of sample-based representations of probability distributions is that probabilities and expected values are in principle straightforward to estimate: To estimate the expected value $\mathbb{E}_{p(y)}[g(y)]$ of a function $g(y)$ under a distribution $p(y)$ from a number of samples $y^1, \ldots, y^T$, simply apply the function to each sample and compute the time average $\frac{1}{T}\sum_{t=1}^{T} g(y^t)$. As long as the samples y^t are distributed according to $p(y)$, either independently drawn, or as the result of an ergodic Markov chain/process with stationary distribution $p(y)$, this estimate is guaranteed to converge to the correct value as one increases the number of samples (Gray, 2009), i.e. $\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} g(y^t) = \mathbb{E}_{p(y)}[g(y)]$. Estimates based on a finite observation window represent an approximation to this exact value.

Under the mild assumptions of Theorem 1 the dynamics of a stochastic spiking network in response to an input x are exponentially ergodic and there exists a unique stationary distribution $p_C(y|x)$, according to which network states $y(t)$ are distributed. Hence, the expected value $\mathbb{E}_{p_C(y|x)}[g(y)]$ of any function $g(y)$ under the stationary distribution $p_C(y|x)$ can be estimated by computing the sample-based time average

$$\frac{1}{T}\int_0^T g(y(t))\,dt \quad . \tag{B.55}$$

This approach can also be used to estimate marginal probabilities, since probabilities can be expressed as expected values, for example,

$$p_C(y_1 = 1|x) = \mathbb{E}_{p_C(y|x)}[\delta(y_1, 1)] \quad \text{, or} \tag{B.56}$$

$$p_C(y_1 = 1, y_2 = 0, y_3 = 1|x) = \mathbb{E}_{p_C(y|x)}[\delta(y_1, 1) \cdot \delta(y_2, 0) \cdot \delta(y_3, 1)] \quad , \tag{B.57}$$

where $\delta(a,b) = 1$ if $a = b$ and 0 otherwise. Hence, in order to estimate the probability $p_C(y_1 = 1|x)$ it suffices to measure the relative time the neuron spends in its active state, i.e. $\frac{1}{T}\int_0^T \delta(y_1, 1)\, dt$. Similarly, to estimate the probability $p_C(y_1 = 1, y_2 = 0, y_3 = 1|x)$ it is sufficient to keep track of the relative frequency of the pattern $(1, 0, 1)$, by computing $\frac{1}{T}\int_0^T \delta(y_1, 1) \cdot \delta(y_2, 0) \cdot \delta(y_3, 1)\, dt$.

B.6 Simulations of data-based cortical microcircuit models

All simulations of microcircuit models for Figures 2.1-2.4 were carried out in PCSIM (Pecevski et al., 2009). A time step of 1 ms was chosen throughout. Further analysis of spike trains was performed in Python (van Rossum and Drake, 2001).

Stochastic neuron model

A stochastic variation of the leaky integrate-and-fire model with conductance-based integration of synaptic inputs was used, for both excitatory and inhibitory neurons. Sub-threshold dynamics of the membrane potential $u(t)$ was defined according to a standard leaky integration model with conductance-based synapses (Gerstner and Kistler, 2002), using passive membrane parameters $R = 60$ MΩ, $C = 0.35$ nF and a resting potential $V_{\text{resting}} = -60$ mV. At simulation start, initial potentials were randomly chosen from $[-65, -55]$ mV. Reversal potentials for excitatory synapses and inhibitory synapses were set to 0 mV and -75 mV, respectively. Neuronal noise was modeled by a voltage-dependent instantaneous probability of firing (instead of a fixed threshold) (Jolivet et al., 2006),

$$\frac{p(\text{neuron spikes in } [t, t+\delta t))}{\delta t} = \frac{1}{\tau_s} e^{(u(t)-\vartheta)/\delta u}, \tag{B.58}$$

for $\delta t \to 0$, with parameters $\tau_s = 19$ ms, $\delta u = 4$ mV taken from (Jolivet et al., 2006). In contrast to (Jolivet et al., 2006) we used a non-adaptive threshold, $\vartheta = -45$ mV. After a spike, a neuron enters an absolute refractory period of 3 ms. Thereafter, the membrane is reset to the resting potential and leaky integration is continued. Altogether, the resulting neuronal spiking mechanism is consistent with the theoretical noise model I described in equation (B.2).

Note that Theorem 1 also holds for substantially more complex multi-compartment neuron models incorporating, for example, data on signal integration in the dendritic tuft of pyramidal cells (Larkum, 2013; Jiang et al., 2013), and data on Ca-spikes in pyramidal cells on layer 5 (Larkum, 2012), but we have not yet integrated these into the simulated microcircuit model because of a lack of coherent quantitative data for all the neuron types involved.

Synaptic short-term plasticity

The short-term dynamics of synapses in all data-based simulations was modeled according to (Maass and Markram, 2002; Markram et al., 1998). The model predicts that at a synapse with "weight" w the amplitude A_k of the k^{th} spike in a spike train with interspike intervals $\Delta_1, \Delta_2, .., \Delta_{k-1}$ is given by,

$$\begin{aligned} A_k &= w \cdot u_k \cdot R_k \ , \\ u_k &= U + u_{k-1}(1-U)\exp^{-\Delta_{k-1}/F} \ , \\ R_k &= 1 + (R_{k-1} - u_{k-1}R_{k-1} - 1)\exp^{-\Delta_{k-1}/D} \ , \end{aligned} \tag{B.59}$$

where the hidden dynamic variables $u_k \in [0,1]$ and $R_k \in [0,1]$ are initialized for the first spike to $u_1 = U$ and $R_1 = 1$. The parameters U,D and F represent the utilization of the synaptic efficacy of the first spike after a resting state, the recovery and the facilitation time constants, respectively. These parameters were set based on experimental data on short-term plasticity in dependence of pre- and post-synaptic neuron (excitatory or inhibitory) as in (Haeusler and Maass, 2007) (see in particular Table 1 in this reference), by randomly drawing for each neuron values for U, D, and F from corresponding data-based Gaussian distributions.

Connectivity and synaptic parameters

Synaptic parameters and connectivity rules for the data-based cortical column model were taken from (Haeusler and Maass, 2007), see Figure 2.1A. In particular, we adopted from (Haeusler and Maass, 2007) the connection probabilities and transmission delays for each type of connection (EE, EI, IE, II) and each cortical layer ((Haeusler and Maass, 2007), Figure 1), as well as short-term plasticity parameters. Furthermore, synaptic efficacies of individual synapses were drawn from Gamma distributions with data-based means and variances for each type of connection (EE, EI, IE, II) taken from (Haeusler and Maass, 2007). Two input streams were connected to the microcircuit, each consisting of 40 input neurons. In contrast to (Haeusler and Maass, 2007) we used rate-based Poisson input neurons instead of injecting "frozen" spike patterns. Background synaptic inputs were emulated as in (Haeusler and Maass, 2007) via background input currents to each neuron, with conductances modeled according to (Destexhe et al., 2001). To adjust connectivity for cortical microcircuit models of different sizes, we also adopted the method proposed by (Haeusler and Maass, 2007), in which recurrent weights are scaled inversely proportional to network size.

We tested the validity of our cortical microcircuit model by comparing the average activity of different layers (see Figure 2.2A) under various conditions against the values reported by (Haeusler and Maass, 2007). We confirmed that all layers exhibited very similar average activity to (Haeusler and Maass, 2007) under all considered conditions.

B.7 Details to small microcircuit model in Figure 2.1

The small cortical microcircuit model of Figure 2.1B was constructed based on the cortical column template of (Haeusler and Maass, 2007): Synaptic connections between neurons and their weights were chosen to approximately reflect connection probabilities and mean synaptic strengths of the cortical column template (Haeusler and Maass, 2007). Due to the very small size of this network, the resulting dynamics was not immediately satisfactory (for example, the influence of inputs on Layer 5 neurons was too weak). To shift the circuit into a more responsive regime, we manually adjusted a few synaptic weights and neuronal excitabilities. In particular, we injected small constant currents into some of the neurons to modulate their intrinsic excitability. Furthermore, to increase activity and correlations between highlighted neurons 2, 7 and 8, we increased synaptic weights $8 \rightarrow 2$ and $8 \rightarrow 7$ by factors 5 and 10, respectively. To set the initial Markov state of the network, preparatory input was shown for 1 s before the actual start of the simulation. Two different preparatory inputs were injected to set the two initial states considered in Figure 2.1F-H (first: i_1 at 100 Hz, i_2 at 100 Hz, second: both i_1 and i_2 at 0 Hz). To reproduce the same initial Markov state in multiple trials (for example the two trials shown in Figure 2.1F), the same random seed was used during the preparatory phase for these trials. The random seed was then reinitialized at $t = 0$ to different values for each trial.

B.8 Estimates of required computation time

Gelman-Rubin univariate and multivariate analysis

Various methods have been developed for measuring convergence speed to a stationary distribution in the context of Markov chain Monte Carlo sampling (Cowles and Carlin, 1996; Brooks and Roberts, 1998; El Adlouni et al., 2006). The Gelman Rubin diagnostic, which we adopted in this article, is one of the most widely used methods (Gelman and Rubin, 1992; Brooks and Roberts, 1998; Brooks et al., 2010; Gjoka et al., 2010), besides other popular methods such as the diagnostics by Raftery and Lewis (Raftery et al., 1992) and by Geweke (Geweke, 1991). We remark that the consensus in the literature is that no single method is perfect in general. Some attractive properties of the Gelman Rubin method are general applicability to any MCMC system (some other methods only work, for example, in the context of Gibbs sampling), ease of use, ease of implementation, computational efficiency, and the fact that results are quantitative (in contrast to graphical diagnostics) (Cowles and Carlin, 1996; Brooks and Roberts, 1998).

The Gelman-Rubin convergence diagnostic (Gelman and Rubin, 1992) takes as input samples from m different runs (trials/chains/sequences) produced by the same system, started from different initial states. The method was originally developed for discrete-time systems in the context of Markov Chain Monte Carlo sampling.

Our simulations use a time step of 1 ms, so we simply treat each simulation step as one discrete time step in a Markov chain. The Gelman-Rubin method produces as output the potential scale reduction factor $\hat{R}(t)$ as a function of time t. The scale reduction factor $\hat{R}(t)$ is an indicator for whether or not the system has converged at time t. High values $\gg 1$ indicate that more time is needed until convergence, while values close to 1 suggest that convergence has (almost) taken place.

For computing the scale reduction factor $\hat{R}(t)$ at time t, samples from the period $[t, 2t]$ from each run of the network are taken into account. In the univariate case one focuses on a particular single variable (such as the marginal simple state of a single neuron, or the simple state of a "random readout" neuron as in the solid lines of Figure 2.2G). Let n be the number of samples obtained from the period $[t, 2t]$ from each of the simulations. Then one defines

$$\hat{R}(t) = \frac{n-1}{n} + \frac{m+1}{mn}\frac{B(t)}{W(t)} \ , \tag{B.60}$$

where $B(t)$ and $W(t)$ are between and within-sequence variances, respectively, which can be computed as described in (Gelman and Rubin, 1992), based on samples taken from the time period $[t, 2t]$. In the rare event of $W = 0$, which happens for example if a neuron never fires and hence its state is constant across all runs, we set $\hat{R}$ to 1.

An unfortunate source of confusion is the fact that Gelman and Rubin (Gelman and Rubin, 1992) originally introduced $\hat{R}$ in its "variance" form equivalent to equation (B.60), but later in (Gelman et al., 2004; Brooks et al., 2010) altered this definition and defined $\hat{R}$ as the square root of (B.60). This issue is particularly critical when considering threshold values for $\hat{R}$: a threshold of 1.2 was suggested in the context of the original definition (Kass et al., 1998). Later, a typical threshold of 1.1 was suggested, but this lower threshold applied to the modified definition (Gelman et al., 2004; Brooks et al., 2010). Squaring this apparently lower threshold yields again a typical threshold of approximately 1.2.

In the multivariate case (e.g. when analyzing convergence of the vector-valued simple state of a small subset of neurons as in the dotted lines of Figure 2.2G) one takes vector-valued (d-dimensional) samples, and computes the multivariate potential scale reduction factor $\hat{R}^d(t)$ according to:

$$\hat{R}^d(t) = \frac{n-1}{n} + \frac{m+1}{m}\lambda_1(t), \tag{B.61}$$

where $\lambda_1(t)$ is the largest eigenvalue of $W(t)^{-1}B(t)/n$, and $W(t)$ and $B(t)$ denote within and between sequence covariance matrix estimates (see (Brooks and Gelman, 1998) for details).

Convergence analysis for cortical microcircuit models

Gelman-Rubin values were calculated based on 100 runs, where the duration of each run was 10 s of biological time. We tried also much longer simulations of 100 s

but did not notice any sign of non-convergent behavior. A random initial state was set in each run by showing random input for 1 s before the start of the actual simulation. This initial random input was fed into the network via the two regular input streams (40 neurons each), by assigning to each input neuron a random rate drawn uniformly from a $0 - 40$ Hz range. Convergence analysis of marginals was performed by applying univariate analysis to single components of the simple state y_S, with $\tau = 10$ ms. From individual marginal convergence values, mean and worst marginal convergence (as in Figure 2.2E,F) were derived by taking at time t the mean/max over all individual $\hat{R}$-values at time t. For pairwise spike coincidences (see Figure 2.2D), we analyzed samples of the product of simple states of two neurons (the product equals 1 only if both neurons spiked within the last 10 ms).

Random readouts for Figure 2.2G were implemented by adding an additional excitatory observer neuron to the network which receives synaptic inputs from a random subset of 500 network neurons (we kept this number 500 fixed across simulations with different network sizes to allow a fair comparison). The number of randomly chosen neurons from each of the pools is given in Table B.1.

	E	I
L2/3	120	30
L4	80	20
L5	200	50

Table B.1: Number of randomly chosen neurons per pool for readout neuron in Figure 2.2G

Synapses onto the readout neuron were created in a similar manner as connections within the cortical column model: short-term plasticity parameters were set depending on the type of connection (EE or IE) according to (Haeusler and Maass, 2007). The weights for EE and IE connections were randomly chosen from a Gamma distribution with mean 2 nS and scale parameter 0.7, and mean 5 nS and scale parameter 0.7, respectively. Gelman-Rubin convergence of readouts was then computed as for the marginal case.

Convergence analysis of vector-valued simple states of subsets of neurons (see Figure 2.2G) was performed by applying multivariate analysis to randomly chosen subnetworks of the cortical column. In particular, we randomly drew 5 neurons from each of the 6 pools, yielding a subnetwork of 30 neurons, and calculated $\hat{R}^{30}(t)$.

B.9 Impact of different dynamic regimes on the convergence time

In Figure 2.3 we compared convergence times in four different neural circuits. The first circuit was identical to the *small cortical microcircuit* from Figure 2.1. For the

remaining three circuits, the same stochastic point neurons and conductance-based dynamic synapses with delays were used as for the data-based cortical microcircuit model. Dynamic synaptic parameters were set to the corresponding mean values of parameters used in the cortical column model. Synaptic delays of 1 ms were used for all networks, except for the network with sequential structure (Figure 2.3C) where delays were 3 ms. To modulate the intrinsic excitability of neurons we injected small currents to each neuron. The strengths of injected currents and connections were tuned for each network until the desired network activity was achieved. Synaptic background inputs were injected as in the cortical microcircuit model. To set different initial states (needed for Gelman Rubin analysis), during a preparatory phase of 1 s we injected into each neuron a random current chosen from $[-2, 2]$ nA. These small random input currents were strong enough to yield sufficiently diverse initial states. Gelman-Rubin values were then calculated based on 100 runs, where the duration of each run (after the preparatory phase) was 20 s of biological time. Convergence analysis was performed on marginals (individual simple states with $\tau = 10$ ms). Mean and worst marginals were computed as described in the previous section.

Below are additional details to the circuits used for Figure 2.3B-D: *The sparsely active network* of Figure 2.3B comprises one excitatory (E) and one inhibitory (I) population (each 10 neurons). Connections between neurons were drawn randomly according to the following set of connection probabilities: EE=0.1, EI=0.1, II=0.9, IE=0.9. *The network with sequential structure* of Figure 2.3C consists of two interconnected subnetworks where each one of them produces a stereotypical trajectory. Each subnetwork consists of a trigger neuron, a subsequent chain of neurons, and a pool of inhibitory neurons. Shown in Figure 2.3C are only the excitatory chain neurons from each subnetwork (neurons 1-15: first subnetwork; neurons 16-30: second subnetwork). Each excitatory neuron in the chain projects to all other neurons in the same chain with synaptic strengths decreasing with distance according to $\exp(-\text{distance}/\tau_d)$ where $\tau_d = 0.01$ applies to the forward direction in the chain and $\tau_d = 0.1$ to the backward direction. The trigger neuron projects (forward) to the chain in the same fashion with $\tau_d = 1$. All neurons in the chain project to the inhibitory pool, and all neurons in the inhibitory pool project back to the trigger neuron and to the chain. Finally, the two subnetworks are combined such that the inhibitory pool of one subnetwork projects to the trigger neuron and the chain of the other subnetwork, and vice versa. This ensures that only one of the two subnetworks can be active at a time (competition between two trajectories). *The bistable network* of Figure 2.3D consists of two populations which strongly inhibit each other (each population comprising 10 neurons).

B.10 Phase-specific distributions in the presence of periodic inputs

The theoretical proof for Theorem 2 can be found after the proof of Theorem 1 above. For Figure 2.4F, a single long simulation (100.000 s) of the bi-stable network in Figure 2.4E was carried out. Each of the two pools was defined active at time t if more than two neurons from the pool had an active simple state at time t (with $\tau = 10$ ms). A transition was defined as the succession of a period in which one pool was active and the other pool inactive by a period in which the other became active and the first pool turned inactive. Between those two periods it typically occurs that either both pools are active or both are inactive for some short time. The exact time (and phase within the current cycle) of each transition was defined as the point in the middle of this intermediate period.

B.11 Generation of heuristic solutions to a constraint satisfaction problem

Formulation of Sudoku as a constraint satisfaction problem

A constraint satisfaction problem consists of a set of variables defined on some domain and a set of constraints, which limit the space of admissible variable assignments. A solution to a problem consists of an assignment to each variable such that all constraints are met. To formulate Sudoku as a constraint satisfaction problem, we define for each of the 81 fields (from a standard 9x9 grid), which has to be filled with a digit from 1 to 9, a set of 9 binary variables (taking values in $\{0, 1\}$) (Ercsey-Ravasz and Toroczkai, 2012). Each of these binary variables votes for exactly one digit in a field. The rules of the Sudoku game impose constraints on groups of these variables, which can be classified into the following three types.

Given number constraints: The given numbers of a puzzle are fixed. Hence, the binary variables for the given fields are constrained to fixed values, for example, a given value 2 corresponds to fixed binary values $(0, 1, 0, \ldots, 0)$.

Unique field constraints: In a correct solution, there must be only one digit active in each field. Hence in each field, exactly one of the 9 associated binary variables must be 1, and all others must be 0 (equivalent to stating that the sum over these binary variables must equal 1).

Unique group constraints: There are three types of groups: rows, columns and 3x3 subgrids. There are 9 row groups, 9 column groups, and 9 subgrid groups. In any of these groups, each digit $1, \ldots, 9$ must appear only once. Hence, in each group, all binary variables voting for the same digit i must sum to 1.

Network architecture for solving Sudoku

Sudoku can be implemented in a spiking neural network by creating for each of the 9 binary variables in each Sudoku field a local group of n_{group} pyramidal cells. Whenever one of these pyramidal cells fires, the corresponding binary variable is set to 1 for a short period $\tau = 20$ ms. The binary variable is defined 0 only if no neuron in its associated group fired within the last $\tau = 20$ ms. This mapping allows one to readout the current (tentative) solution represented by the network at any time t. The tentative solution is correct only if all constraints are met. For all simulations we used $n_{\text{group}} = 4$, resulting in a total $81 * 9 * 4 = 2914$ pyramidal cells. Constraints among Sudoku variables can be implemented via di-synaptic inhibition between the groups of pyramidal cells as detailed below.

Given number constraints are implemented by providing strong positive input currents selectively to those neurons which code for the given numbers, and negative currents to neurons coding for wrong digits in a given field. *Unique field constraints* are implemented by forming a winner-take-all (WTA) circuit among all $9 * 4 = 36$ neurons associated with the same Sudoku field. A WTA circuit is modeled by a single inhibitory neuron which is reciprocally connected to all 36 pyramidal cells. To reduce the probability that no pyramidal cell fires (which would violate the unique field constraint), thresholds of pyramidal cells are set to low values (see next section for details). *Unique group constraints* are implemented by a WTA circuit in which all neurons in a group which code for the same digit participate. In summary, there are 81 unique field constraints and $27 * 9 = 243$ unique group constraints (in each group there is a constraint for each digit), yielding a total of 324 WTA circuits. These WTA circuits are partially overlapping, in the sense that each pyramidal cell participates in 4 of these WTA circuits (one for the unique value constraint in its field, and three for the unique group constraints in its row/column/subgrid).

Stochastic spike generation in both excitatory and inhibitory neurons is implemented consistent with the theoretical noise model I (see next section for details). The network thus fulfills all theoretical conditions for Theorem 1, and is guaranteed to have a unique stationary distribution $p_C(y|x)$ of network states, to which it converges exponentially fast. This landscape will have automatically peaks at those states of the network which fulfill most of the game constraints, since each of the WTA circuits ensures that invalid configurations with respect to that constraint are unlikely to occur. Any specific Sudoku problem can be set by providing input x to the network in the form of strong currents to those neurons which correspond to the given values. This automatically modifies the landscape of the stationary distribution $p_C(y|x)$ such that only (or predominantly) solutions consistent with the givens are generated. Finally, due to neuronal noise the network can quickly probe different peaks in the landscape (different promising solution candidates) and escape them equally fast. Importantly, this process may occur at different places in the Sudoku puzzle simultaneously. Hence, one can interpret the network dynamics also as a highly parallel stochastic search algorithm.

Details to implementation and simulations for Figure 2.5

Simulations for Figure 2.5 were performed in NEVESIM, an event-based simulator for networks of spiking neurons developed in C++ with a Python Interface(Pecevski, 2013). The puzzle in Figure 2.5A was generated and rated "hard" by "Sudoku Solutions" (Aire Technologies, 2013). Spike generation is modeled according to equation (B.58), with parameters $\delta u = 0.5$, $\tau_s = 20$ ms. The stochastic threshold ϑ was set to -1 and 10 for excitatory and inhibitory neurons, respectively. An absolute refractory period of 3 ms was chosen for pyramidal cells. To maximize the speed up of event-based simulations, PSPs were modeled in a simplified manner as current-based rectangular pulses of length 20 ms (in contrast to the more complex conductance based integration of synaptic inputs used for cortical microcircuit models).

WTA circuits were formed by reciprocally connecting a single inhibitory neuron to all participating pyramidal cells. The single inhibitory neuron was modeled to mimic the response of a population of inhibitory neurons (i.e. strong inhibition for a prolonged amount of time), using an absolute refractory period of 20 ms, and strong bidirectional connections from and to excitatory neurons (synaptic weights 100 and -100, respectively).

To set a particular puzzle, given numbers were fixed by providing strong input currents to the corresponding pyramidal cells. In particular, neurons coding for the given numbers in a Sudoku field received a constant positive input current (a constant input +9 on the membrane potential). Neurons coding for conflicting digits in given Sudoku fields received a constant negative input current of strength -11.

A final practical remark concerns the number of neurons coding for each binary variable, $n_{\text{group}} = 4$. We found that networks with $n_{\text{group}} > 1$ have a number of attractive properties compared to networks with single neuron coding. In particular firing rates of individual neurons can be lower (for $n_{\text{group}} = 1$ a pyramidal cell would need to constantly burst to indicate a steady active state). Also, synaptic efficacies among neurons can be made weaker, and overall spike response patterns appear more biologically plausible. In view of a potential implementation in analog neuromorphic hardware, population coded variable assignments are also less prone to single unit failures or device mismatch.

Appendix C

Appendix to Chapter 3: Solving Hard Computational Problems with Networks of Stochastic Spiking Neurons

Contents

C.1 Stochastic neuron model

Neurons are modeled as simple stochastic point neurons with absolute refractory period τ. When not in a refractory state, neuron k spikes at an instantaneous firing rate which depends exponentially on the membrane potential $u_k(t)$ (3.2), according to,

$$\lim_{\delta t \to 0} p(\text{neuron k fires within}(t, t+\delta t])/\delta t = \rho_k(t) = \frac{1}{\tau}\exp(u_k(t)) \ , \qquad \text{(C.1)}$$

with $\tau = 10$ms unless otherwise stated. An exponential dependence of a neuron's firing probability on the membrane potential has been suggested by (Jolivet et al., 2006) based on a fit to experimental data. Similar stochastic neuron models have been suggested by (Truccolo et al., 2005; Buesing et al., 2011).

C.2 Details to Principle 1: stationary distributions and energy functions

Network states

We distinguish between principal neurons and auxiliary neurons. Principal neurons directly represent the random variables (RV) of a problem (e.g. the boolean variables in a satisfiability problem as in Figure 3.2). The state of principal neurons therefore reflects the state of the RVs. Auxiliary neurons (i.e. all auxiliary neurons in circuit motifs and the *lock-in* neuron), on the other hand, do not represent random variables. Their only purpose is to modulate and shape the distribution (and energy function) over principal neurons.

The state $x_k(t)$ of a principal neuron k at time t is defined as,

$$x_k(t) = \begin{cases} 1, & \text{if neuron k fired within } (t-\tau, t] \ , \\ 0, & \text{otherwise} \ , \end{cases} \tag{C.2}$$

where τ is a brief time window corresponding to the duration of a PSP. The state $\xi_m(t)$ of an auxiliary neuron m is defined in an analogous manner. The *full network state*,

$$(\mathbf{x}(t), \boldsymbol{\xi}(t)) = (x_1(t), \ldots, x_N(t), \xi_1(t), \ldots, \xi_M(t)) \tag{C.3}$$

is defined as the vector of states of all principal neurons $k = 1, \ldots, N$ and all auxiliary neurons $m = 1, \ldots, M$ in the network. Similar notions of network state have been suggested by a number of experimental (Schneidman et al., 2006; Berkes et al., 2011) and theoretical (Buesing et al., 2011; Pecevski et al., 2011; Habenschuss et al., 2013) papers. The *principal network state* refers only to the state vector $\mathbf{x}(t)$ of all principal neurons. Unless otherwise stated, the term *network state* refers to the principal network state.

Convergence to stationary distribution

Under mild conditions, activity in a general spiking network with noise can be theoretically guaranteed to converge exponentially fast to a unique stationary distribution $p(\mathbf{x}, \boldsymbol{\xi})$ of full network states (Habenschuss et al., 2013), regardless of initial network conditions. In the context of the stochastic neuron model (3.1-3.2) it can be easily verified that the theoretical conditions for convergence are fulfilled if all weights w_{kl} are bounded from above, i.e. if there exists some $w_{\max}$ such that all $w_{kl} \leq w_{\max}$. Throughout the paper this condition is met. Exponentially fast convergence to a unique *marginal* distribution $p(\mathbf{x})$ over principal network states is a simple corollary that follows from the convergence to a unique joint distribution $p(\mathbf{x}, \boldsymbol{\xi})$.

Energy functions

In analogy with statistical physics (Plischke and Bergersen, 2006), we define the energy function $E(\mathbf{x})$ of a network of spiking neurons with unique stationary distribution $p(\mathbf{x})$ of principal network states $\mathbf{x}$ as

$$E(\mathbf{x}) = -\log p(\mathbf{x}) + C \quad , \tag{C.4}$$

with an arbitrary constant C. The stationary distribution $p(\mathbf{x})$ can then be expressed as,

$$p(\mathbf{x}) = \frac{e^{-E(\mathbf{x})}}{\sum_{\mathbf{x}'} e^{-E(\mathbf{x}')}} \quad . \tag{C.5}$$

Note that according to this definition, energies are defined only up to a constant (a global shift applied to all states). To indicate that two energy functions are identical except for a constant shift we use the notation $E_1(\mathbf{x}) \triangleq E_2(\mathbf{x})$, i.e.

$$E_1(\mathbf{x}) \triangleq E_2(\mathbf{x}) \quad \Leftrightarrow \quad \exists C \in \mathbb{R}\ \forall \mathbf{x}\ \ (E_1(\mathbf{x}) = E_2(\mathbf{x}) + C) \quad . \tag{C.6}$$

C.3 Details to Principle 2: circuit motifs shaping the energy function

A key theoretical question is how the energy function $E(\mathbf{x})$ (or equivalently $p(\mathbf{x})$) over principal network states $\mathbf{x}$ depends on the parameters of a network, in particular on synaptic weights w_{kl} and neuronal excitabilities b_k among principal neurons, as well as on auxiliary circuits connected to the principal neurons. Previous work had shown that pair-wise symmetric connections between neurons map onto second-order dependencies between variables (Buesing et al., 2011). (Pecevski et al., 2011) demonstrated in addition how more complex dependencies can be encoded through the use of pre-processing circuits in the context of probabilistic inference.

Here we consider how in addition to second-order dependencies, common higher-order constraints of hard computational problems can be encoded through the use of simple auxiliary circuit motifs, in a manner suitable for compositionality and large-scale circuit design.

Compositionality

To facilitate systematic design of complex energy landscapes, we would like to find a basic set of auxiliary circuit motifs which can be combined in arbitrarily rich ways with predictable outcomes. A particularly desirable feature to aim for is linear compositionality, such that the energy contribution to the energy landscape of each circuit motif is independent of the presence of other circuits. A precise definition follows.

Definition 1 (Compositionality). *Let $\mathcal{N}$ be a principal network of $k = 1, \ldots, N$ stochastic principal neurons (3.1)-(3.2), symmetric connections $w_{kl} = w_{lk}$ (but no self-connections, i.e. $w_{kk} = 0$) and biases b_k, with energy function $E_{\mathcal{N}}(\mathbf{x})$. Let $\mathcal{C} = \{C_1, \ldots, C_L\}$ be a set of L additional auxiliary circuits which can be reciprocally connected to the principal network $\mathcal{N}$ to modulate the behavior of principal neurons. Denote by $E_{\mathcal{N},\mathcal{I}}(\mathbf{x})$ the modulated energy function of the network in the presence of a subset $\mathcal{I} \subseteq \{1, \ldots, L\}$ of these auxiliary circuits, and define the change in the energy landscape due to the presence of this subset $\mathcal{I}$ as $\Delta E_{\mathcal{N},\mathcal{I}}(\mathbf{x}) \triangleq E_{\mathcal{N},\mathcal{I}}(\mathbf{x}) - E_{\mathcal{N}}(\mathbf{x})$. Then the set of auxiliary circuits $\mathcal{C}$ is said to be compositional with respect to network $\mathcal{N}$ if changes in energies sum up linearly for all possible combinations, i.e.*

$$\Delta E_{\mathcal{N},\mathcal{I}}(\mathbf{x}) \triangleq \sum_{i \in \mathcal{I}} \Delta E_{\mathcal{N},i}(\mathbf{x}) \quad , \tag{C.7}$$

for any subset $\mathcal{I} \subseteq \{1, \ldots, L\}$.

Note that due to linearity of membrane integration (3.2), the membrane potential of a principal neuron k in the presence of some subset $\mathcal{I}$ of arbitrarily complex auxiliary circuits can be written as,

$$u_{k,\mathcal{I}}(t) = b_k + \sum_l w_{kl}\, x_k(t) + \sum_{i \in \mathcal{I}} \Delta u_{k,i}(t) \quad , \tag{C.8}$$

where the current contribution of auxiliary circuit C_i to the membrane potential of principal neuron k is denoted by $\Delta u_{k,i}(t)$. Define $\mathbf{x}_{\setminus k}(t)$ as the state vector of all principal neurons except neuron k, and $\{x_k = \cdot, \mathbf{x}_{\setminus k}(t)\}$ as the state vector $\mathbf{x}(t)$ with the state of neuron k replaced by $\cdot$. Then, building on the analysis of (Buesing et al., 2011), the following theoretical result provides sufficient conditions on the auxiliary circuit contributions $\Delta u_{k,i}(\cdot)$ for compositionality.

Theorem 3 (Sufficient conditions for compositionality). *Let $\mathcal{N}$ be any network of principal neurons and $\mathcal{C}$ a set of auxiliary circuits as defined above. Suppose that for each auxiliary circuit C_i there exists an energy function $U_i(\mathbf{x})$ such that at any time t the following relation holds,*

$$\Delta u_{k,i}(t) = U_i\left(\{x_k = 0, \mathbf{x}_{\setminus k}(t)\}\right) - U_i\left(\{x_k = 1, \mathbf{x}_{\setminus k}(t)\}\right) \tag{C.9}$$

Then the set of auxiliary circuits $\mathcal{C}$ is compositional with respect to $\mathcal{N}$. Furthermore, the energy change due to each individual circuit is given by $\Delta E_{\mathcal{N},i}(\mathbf{x}) \triangleq U_i(\mathbf{x})$.

Theorem 3 suggests that auxiliary circuits should be constructed in a highly specific manner to support compositionality. In particular, (C.9) states that auxiliary circuit contributions to the membrane potential of a principal neuron k should be basically memoryless and reflect a specific function of the current

state of the remaining network, $\mathbf{x}_{\setminus k}(t)$. Note that this function (the right-hand side of (C.9)) has a very intuitive interpretation: a circuit C_i should inform each principal neuron k about the currently expected drop in the energy function U_i that can be achieved by a spike of neuron k (i.e. a switch from $x_k = 0$ to $x_k = 1$).

Proof of Theorem 3: If (C.9) holds for all C_i then the membrane potential of a principal neuron k in the presence of some subset of auxiliary neurons $\mathcal{I}$ is given at time t by,

$$u_{k,\mathcal{I}}(t) = b_k + \sum_{l=1}^{N} w_{kl}\, x_k(t) + \sum_{i\in\mathcal{I}} \left[U_i\left(\{x_k = 0, \mathbf{x}_{\setminus k}(t)\}\right) - U_i\left(\{x_k = 1, \mathbf{x}_{\setminus k}(t)\}\right)\right] \tag{C.10}$$

This can also be expressed as,

$$u_{k,\mathcal{I}}(t) = U_{\mathcal{I}}\left(\{x_k = 0, \mathbf{x}_{\setminus k}(t)\}\right) - U_{\mathcal{I}}\left(\{x_k = 1, \mathbf{x}_{\setminus k}(t)\}\right) \quad , \tag{C.11}$$

with

$$U_{\mathcal{I}}(\mathbf{x}) = -\sum_{k=1}^{N} b_k x_k - \frac{1}{2}\sum_{k=1}^{N}\sum_{l=1}^{N} w_{kl} x_k x_l + \sum_{i\in\mathcal{I}} U_i(\mathbf{x}) \quad . \tag{C.12}$$

One can then verify that the neural computability condition (NCC) from (Buesing et al., 2011) is fulfilled by a network with membrane dynamics (C.11) with respect to stationary distribution $p(\mathbf{x}) \propto \exp(-U_{\mathcal{I}}(\mathbf{x}))$:

$$\log \frac{p(x_k = 1|\mathbf{x}_{\setminus k})}{p(x_k = 0|\mathbf{x}_{\setminus k})} = \log \frac{p(\{x_k = 1, \mathbf{x}_{\setminus k}\})}{p(\{x_k = 0, \mathbf{x}_{\setminus k}\})} \tag{C.13}$$

$$= \log p(\{x_k = 1, \mathbf{x}_{\setminus k}\}) - \log p(\{x_k = 0, \mathbf{x}_{\setminus k}\}) \tag{C.14}$$

$$= -U_{\mathcal{I}}(\{x_k = 1, \mathbf{x}_{\setminus k}\}) + U_{\mathcal{I}}(\{x_k = 0, \mathbf{x}_{\setminus k}\}) \tag{C.15}$$

$$= b_k + \sum_{l=1}^{N} w_{kl} x_l + \sum_{i\in\mathcal{I}} [-U_i(\{x_k = 1, \mathbf{x}_{\setminus k}\}) + U_i(\{x_k = 0, \mathbf{x}_{\setminus k}\})] \tag{C.16}$$

Thus, a network with membrane dynamics (C.12) meets the NCC for $p(\mathbf{x}) \propto \exp(-U_{\mathcal{I}}(\mathbf{x}))$, and the energy function of the network is given by $E_{\mathcal{N},\mathcal{I}}(\mathbf{x}) \triangleq U_{\mathcal{I}}(\mathbf{x})$. Furthermore, from (C.12) it is obvious that energies due to combinations of auxiliary circuits sum up linearly and that the energy contribution due to each single C_i equals $\Delta E_{\mathcal{N},i}(\mathbf{x}) \triangleq U_i(\mathbf{x})$.
□

Note that, in contrast to neural sampling theory (Buesing et al., 2011), Theorem 3 is only concerned with the distribution over a subset of all neurons (the principal neurons $\mathbf{x}$), i.e. the marginal distribution $p(\mathbf{x})$ after integrating out all auxiliary variables $\boldsymbol{\xi}$.

WTA circuit motif

The WTA circuit motif consists of a single auxiliary neuron which is reciprocally connected to some subset $\mathcal{K} \subseteq \{1, \dots N\}$ of principal neurons (Figure 3.1B). The goal of the WTA motif is to achieve that most of the time *exactly* one neuron in $\mathcal{K}$ is active. The WTA motif should thus increase the energies of all network states except those states where exactly one neuron in $\mathcal{K}$ is active. This can be achieved in two steps. First, the energy of all network states where more than one neuron in $\mathcal{K}$ is active is increased. We found that this can be robustly achieved by a single inhibitory neuron which receives strong excitatory connections from $\mathcal{K}$, and sends strong inhibitory connections back to $\mathcal{K}$ (with some weight $-w_{\text{WTA}} \ll 0$). The inhibitory neuron should have a low bias such that it only fires when one of the principal neurons is active. Second, the energy of states where no neuron in $\mathcal{K}$ is active is raised. This can be done most easily by raising the biases of all neurons in $\mathcal{K}$ by some constant b_{WTA} (not shown in Figure 3.1B) with $0 < b_{\text{WTA}} < w_{\text{WTA}}$. Alternatively, this could in principle also be achieved by an additional auxiliary neuron which is constantly active and makes excitatory connections to all neurons in $\mathcal{K}$.

The described implementation of the WTA circuit motif is intended to approximate the requirements of Theorem 3 for compositionality. This can be seen if one considers the energy function

$$U_{\text{WTA}[\mathcal{K}]}(\mathbf{x}) = \begin{cases} b_{\text{WTA}} \ , & \text{if } \sum_{k\in\mathcal{K}} x_k = 0 \ , \\ 0 \ , & \text{if } \sum_{k\in\mathcal{K}} x_k = 1 \ , \\ (w_{\text{WTA}} - b_{\text{WTA}}) \cdot (-1 + \sum_{k\in\mathcal{K}} x_k) \ , & \text{if } \sum_{k\in\mathcal{K}} x_k > 1 \ . \end{cases} \tag{C.17}$$

According to (C.9) the ideal $\Delta u_{k,\text{WTA}[\mathcal{K}]}(t)$ for implementing this energy function in a compositional manner is given by,

$$\Delta u_{k,\text{WTA}[\mathcal{K}]}(t) = \begin{cases} b_{\text{WTA}} \ , & \sum_{l\in\mathcal{K}\setminus k} x_l(t) = 0 \ , \\ b_{\text{WTA}} - w_{\text{WTA}} \ , & \sum_{l\in\mathcal{K}\setminus k} x_l(t) > 0 \ . \end{cases} \tag{C.18}$$

This behavior is closely approximated by the described WTA circuit implementation: Regardless of the network state, there is a bias term b_{WTA}. As soon as one (or more) of the neurons fire, this triggers the auxiliary inhibitory neuron which then strongly inhibits all competitors with weight $-w_{\text{WTA}}$. The nature of the approximation lies mainly in the delay between the onset of activity of a winner and the onset of inhibition at the remaining principal neurons.

OR circuit motif

The OR circuit motif consists of two auxiliary neurons reciprocally connected to some subset $\mathcal{K} \subseteq \{1, \dots N\}$ of principal neurons (Figure 3.1B). The purpose of the OR motif is to ensure that most of the time *at least one* neuron in $\mathcal{K}$ is active.

Thus, the energies of all network states where no neuron in $\mathcal{K}$ is active should be increased. At the same time, however, the energies of all other network states should ideally remain unmodified, regardless of how many neurons ≥ 1 are active. The first part, i.e. increasing the energies of states where no neuron is active, can be done by adding an auxiliary neuron (Figure 3.1B, left auxiliary OR neuron) which excites all neurons in $\mathcal{K}$ with equal synaptic weight w_{OR}. The second part is slightly more tricky, as it requires that the OR circuit should suspend its influence on the network during periods where at least one neuron in $\mathcal{K}$ is active. This can be achieved by a) adding an inhibitory connection from $\mathcal{K}$ to the first auxiliary neuron such that the neuron is only activated when needed, and b) by adding a second auxiliary neuron (Figure 3.1B, right auxiliary OR neuron) which is triggered when a neuron in $\mathcal{K}$ fires in response to the first auxiliary neuron. The goal of the second auxiliary neuron is to immediately cancel any effect of the first auxiliary neuron on the remaining neurons in $\mathcal{K}$ due to sustained post-synaptic potentials. This is achieved through inhibitory connections $-w_{OR}$ to all neurons in $\mathcal{K}$.

Analogous to the WTA circuit, the described implementation of the OR circuit motif aims to approximate the requirements of Theorem 3 for compositionality. To see this, consider the energy function

$$U_{\mathrm{OR}[\mathcal{K}]}(\mathbf{x}) = \begin{cases} 0\,, & \text{if } \sum_{k\in\mathcal{K}} x_k \geq 1\,, \\ w_{\mathrm{OR}}\,, & \text{if } \sum_{k\in\mathcal{K}} x_k = 0\,. \end{cases} \tag{C.19}$$

By (C.9) the corresponding ideal $\Delta u_{k,\mathrm{OR}[\mathcal{K}]}(t)$ supporting compositionality is given by,

$$\Delta u_{k,\mathrm{OR}[\mathcal{K}]}(t) = \begin{cases} 0\,, & \sum_{l\in\mathcal{K}\setminus k} x_l(t) \geq 1\,, \\ w_{\mathrm{OR}}\,, & \sum_{l\in\mathcal{K}\setminus k} x_l(t) = 0\,. \end{cases} \tag{C.20}$$

The OR circuit approximates this behavior as described above through the combination of two auxiliary neurons. The nature of the approximation is three-fold. First, when all principal neurons in an OR circuit have just turned off (and thus the constraint is not met anymore), the additional bias w_{OR} should ideally be communicated instantly to all neurons. However, the first auxiliary neuron fires in general with some small delay, and therefore the additional bias w_{OR} is signaled to the principal neurons slightly later than ideally required. Second, when a principal neuron eventually fires in response to the first auxiliary neuron, there is a delay until the second auxiliary neuron turns on to cancel the bias w_{OR} that is still present due to lingering PSPs from the first auxiliary neuron. Third, there is an "undershoot" effect when the excitatory PSP of the first principal neuron has already vanished, but the inhibitory PSP of the second auxiliary neuron is still present. To minimize the error due to this effect, the overall biases of all principal neurons in an OR circuit should be kept high, in order to keep the typical delay between the activity onset of the first and the second auxiliary neuron as short as possible.

C.4 Details to Principle 3: benefits of asymmetric spike-based signaling

Principles 1 and 2 pave the way towards massively parallel realizations of stochastic search in networks of spiking neurons. A first application of these principles has provided compelling results in simulations, as demonstrated in Figure 3.2 and Figure 3.3. A key theoretical question which then arises is to what extent different components of the system contribute to the observed performance. There are various aspects that can be examined in this context, such as the asynchronicity of message transfer, stochasticity, and the asymmetry of spike-based communication (a spike marks the onset of a fixed-length *on* period, whereas *off* periods vary randomly - hence *on* and *off* states are handled fundamentally different by a spiking network). We focus our analysis here on the role of the asymmetry of spike-based signaling, because its implications are arguably least well understood.

Asymmetric vs. symmetric dynamics

In order to isolate the effect of asymmetric signaling we consider an artificial non-spike-based "symmetrized" system in which *on* and *off* transitions of units are not mediated in an asymmetric fashion via spikes of fixed length, but rather in a symmetric manner. Specifically, we aim to morph neural spiking dynamics into the dynamics of Gibbs sampling (Bishop, 2006), one of the standard methods in statistics and machine learning for sampling from complex probability distributions. By theoretically analyzing and comparing the behavior of the two systems one can then reason about the specific role of asymmetric signaling.

A canonical way of symmetrizing the dynamics of a given spiking network with noise is to make sure that all other components and aspects of the systems remain unchanged (event-based asynchronous signaling, stochasticity, synaptic weights and biases, definition of membrane potential u_k given the current *on*/*off* states of other neurons) and modify only the way the system handles transitions between *on* and *off* states. Importantly, to facilitate a comparison between asymmetric vs. symmetric dynamics, such modification should not alter the stationary distribution and energy function of the system.

For a stochastic spiking neuron embedded in some network, transitions occur from *off* to *on* states according to

$$\rho_{\text{on}}(u_k) = \frac{1}{\tau}\exp(u_k) \ , \tag{C.21}$$

whereas transitions from *on* to *off* occur deterministically after a period of τ time units has passed. Clearly, in a symmetric system transitions must occur stochastically in both directions (they cannot be both deterministic), with transition rates $\rho'_{\text{on}}(u_k)$ and $\rho'_{\text{off}}(u_k)$. Concrete symmetric expressions for $\rho'_{\text{on}}(u_k)$ and $\rho'_{\text{off}}(u_k)$ are

obtained by using a continuous-time variant of Gibbs sampling (Bishop, 2006).

$$\rho'_{\text{on}}(u_k) = \rho_0 \cdot \sigma(u_k) \ , \tag{C.22}$$
$$\rho'_{\text{off}}(u_k) = \rho_0 \cdot \sigma(-u_k) \ , \tag{C.23}$$

where $\sigma(u) = (1 + \exp(-u))^{-1}$ denotes the standard sigmoid function. Such a continuous-time variant of Gibbs sampling has been proposed in the literature, for example, in the context of sampling from second-order Boltzmann machines (Yamanaka et al., 1997).

Asymmetry facilitates transitions across large energy barriers

A somewhat unexpected but striking difference which emerges from the comparative analysis between asymmetric and symmetric dynamics is that transitions across large energy barriers are much more likely and frequently to occur with asymmetric (spike-based) signaling. To see this, define the mean *on*-transition time $m_{\text{on}}(u)$ as the average time from the last *on*$\rightarrow$ *off* transition until the next *off*$\rightarrow$*on* transition, at a given membrane potential u. The mean *off*-transition time is defined in an analogous manner. In the stochastic spiking network these are given by,

$$m_{\text{on}}(u) = \frac{1}{r_{\text{on}}(u)} = \tau \cdot \exp(-u) \ , \tag{C.24}$$
$$m_{\text{off}}(u) = \tau \ . \tag{C.25}$$

In the symmetric system, on the other hand, mean transition times are given by,

$$m'_{\text{on}}(u) = \frac{1}{r'_{\text{on}}(u)} = \frac{1}{\rho_0} \cdot (1 + \exp(-u)) \ , \tag{C.26}$$
$$m'_{\text{off}}(u) = \frac{1}{r'_{\text{off}}(u)} = \frac{1}{\rho_0} \cdot (1 + \exp(u)) \ . \tag{C.27}$$

Notably, one can identify a single translation factor $F(u)$ between the two systems,

$$m_{\text{on}}(u) = m'_{\text{on}}(u) \cdot F(u), \tag{C.28}$$
$$m_{\text{off}}(u) = m'_{\text{off}}(u) \cdot F(u) \tag{C.29}$$

which is given by,

$$F(u) = \underbrace{\tau\rho_0}_{\text{const.}} \cdot (1 + \exp(u))^{-1} \tag{C.30}$$

Note that $F(u)$ is strictly positive and decreases monotonically with increasing membrane potential u. Furthermore, note that small values $F(u)$ signify that the asymmetric dynamics is fast in comparison with the symmetric dynamics (in

both, *on* and *off* directions). Hence, (C.30) shows that the asymmetric dynamics of spiking neurons increases specifically the *on*- and *off*-transition rates of those neurons with high membrane potentials u (i.e. neurons with strong input and/or high biases). This makes sense since *off* transitions in the symmetric case can be arbitrarily slowed down for large u (C.27), whereas the spike-based system will necessarily fall back to an *off* state on a regular basis regardless of u.

Given that transitions are specifically enhanced in the presence of high membrane potentials, and taking into account that large u reflect large energy barriers (according to (C.12)), it follows that the spike-based system is much more inclined to make exploratory *on*→*off* transitions (crossing large energy barriers) on a regular basis. Despite the resulting increased frequency of *transitions* to high-energy states due to (C.30), however, it should be stressed that on average the asynchronous spike-based system does not spend more *time* in high-energy states (both systems sample from the same $p(\mathbf{x})$), because according to (C.28) also the transition back to the corresponding *on* state (i.e. the lower energy state) happens at an increased rate for large u. The critical observation is that the return to the identical previous state *can* be intercepted by other neurons which, while the neuron is *off*, are given the brief opportunity to spike before the previous state is restored, and may thereby, e.g., escape from a previously inhibited state. This is particularly obvious in the context of WTA circuits, where such brief periods of *off*-time of the current winner allow other neurons to take over. Altogether, as we demonstrated in Figure 3.3, it is observed that this enhanced utilization of exploratory moves leads to improved search for low energy states in the asymmetric spike-based system, by facilitating fast escape routes from deep local minima which are not available to such extent in a symmetric system.

Asymmetry facilitates goal-directed transitions

(C.30) states that spike-based transition frequency is enhanced in proportion to u. It was already noted above that this encourages exploratory *on*→*off* transitions which may facilitate the escape from local minima. But clearly also *off*→*on* transitions are affected by (C.30). In particular, consider a situation where a group of neurons in the *off* state is competing for emitting the next spike (e.g. in a WTA circuit). Those neurons with the highest membrane potentials are particularly eager to fire. Suppose, for example, that there are two neurons with $u_a = 3$ and $u_b = 5$, and all other neurons have considerably lower u. In the symmetrized non-spiking system, transition rates scale with $\sigma(u)$ and are therefore approximately equal for the two neurons a and b (due to saturation of the sigmoid function). In the spike-based system, however, instantaneous transition rates scale with $\exp(u)$ and thus the competition will be much easier to win by the neuron which is most eager to fire (i.e. neuron b in the example). Clearly, this makes a substantial difference in the dynamics and performance of the stochastic search, especially since u_k reflects the drop in energy that can be gained by turning on some neuron k. In particular,

it means that a spike-based system is not only more exploratory in the "up-hill" direction (*on*→*off* transitions towards higher energy levels), but also more goal-directed in the "down-hill" direction.

Obviously, the enhanced agility with respect to some transitions must come at a price. Indeed, those transitions which bring about only small changes in the energy landscape (transitions with small u) are considerably disadvantaged by the spike-based dynamics. In terms of convergence properties, however, this seems to be a small price to pay, since stochastic search appears in practice more frequently impeded by the presence of large energy barriers. [1]

C.5 Details to Principle 4: internal temperature control

In order to realize an internal temperature control mechanism which allows network activity to "lock in" when a good solution has been found, the following functional components are required (Figure 3.1D): 1. The generation of *OK* signals in each circuit motif. 2. A *lock-in unit* that integrates individual *OK* signals into a global *all OK* message. 3. The activation of additional circuits which reduce temperature.

The following realizations of these elements have proved effective: For the WTA circuit motif, the activity of the inhibitory neuron can be directly used as an *OK* signal. This works because the probability that two neurons are active at the same time is vanishingly small as long as strong inhibitory connections are used in the WTA motif. Hence, in practice whenever the inhibitory neuron is active it means that exactly one principal neuron is active (and the WTA constraint is met). For the same reasons, one can also simply connect all neurons in a WTA circuit to the *lock-in* neuron. Since at most one neuron is active at a time, the joint impact of these neurons on the *lock-in* unit precisely reflects whether the WTA constraint is met. For the OR circuit motif, the most straightforward way of implementing an *OK* signal is to add another auxiliary neuron with low bias and excitatory connections from all involved principal neurons, such that the neuron fires as long as one of the principal neurons is active, and remains silent otherwise. In simulations, however, a slightly different implementation has proved more effective, which can be used when all principal neurons involved in the OR circuit are also part of some WTA circuit. Then, a *not OK* signal can be derived by adding an auxiliary neuron with low bias which receives connections from all other neurons in the WTA circuits of the involved principal neurons. This works because, whenever principal neuron k

[1] Clearly, also transitions with negative u are disadvantaged by the spike-based dynamics. In general, this may have a negative effect on convergence, and the magnitude of such negative effects would need to be examined in relation to the previously described advantages. In the context of this paper, however, negative u practically only occur in neurons which are not supposed to fire at all, for example neurons which are currently inhibited in a WTA circuit. And in this case it is in fact desirable that such transitions occur with decreased frequency.

(which is involved in the OR circuit and in addition in some WTA circuit) is not active, some other neuron in the WTA circuit of neuron k must be active (most of the time). Hence, whenever the OR constraint is violated and all K principal neurons involved in the OR circuit are inactive, the auxiliary neuron will see that in each of the involved WTA circuits some other neuron is active. A more detailed description of how this was implemented as part of the *lock-in* mechanism for 3-SAT problems is given in Section "Details to 3-SAT application".

The *lock-in neuron* can be implemented by choosing a low bias and connection strengths from *OK* neurons in each circuit in such a manner that the firing probability reaches non-negligible values only when all *OK* signals are active. When circuits send either *OK* or *not OK* signals, the connection strengths from *not OK* should be negative and can be chosen in such a manner that non-negligible firing rates are achieved only if all *OK* but none of the *not OK* signals are active.

Regarding the activation of additional circuitry to reduce temperature, the most straightforward way of achieving this is to duplicate all circuit motifs (as indicated in Figure 3.1D). The biases of auxiliary neurons in duplicated circuits should be much lower, such that these circuits remain inactive unless an additional excitatory drive is provided by the *lock-in* neuron. This is exactly how temperature reduction was implemented for the OR circuit motif. For WTA circuits, however, there exists an even a simpler way of reducing temperature which does not require duplication of circuitry but only excitatory connections from the *lock-in* neuron to all principal neurons in a WTA circuit. This works well because the WTA circuit motif consists of two components, a) excitatory drive (increased bias) to all involved principal neurons, and b) strong mutual inhibition. If inhibition strength is very strong, however, duplication of that second component is not necessary. Hence, a reduction of temperature can be achieved by mere activation of additional excitatory drive. For further implementation details see Section "Details to 3-SAT application".

C.6 Details to simulations

All simulations were performed in NEVESIM, an event-based neural simulator. Optimization of networks, as well as exploration of their properties were done with ZLIB, a library for parallelization and optimization, developed within the scope of this work. The analysis of simulation results was performed in Python and Matlab.

Details to 3-SAT application (Figure 3.2)

A general 3-SAT problem consisting of a set of binary variables and a set of clauses, each involving three variables, can be implemented in a spiking network by representing each binary random variable (RV) with two neurons forming a WTA circuit (biases of the principal neurons: b_{nrn}). In particular, the WTA circuit is implemented by adding a single inhibitory neuron with bias b_{inh} and connecting it to

the two neurons with bidirectional connections w_{inh} and w_{exc} (to and from the inhibitory neuron, respectively). The w_{inh} should be set strong enough to shut down all principal neurons in the WTA circuit (to overcome their biases). w_{exc} should be strong enough such that it activates inhibition almost immediately in order to prevent other neuron(s) from spiking.

For the implementation of a 3-SAT clause one needs to form an OR circuit consisting of those neurons which take part in the clause. In particular, two auxiliary neurons are added, with biases of $0.5B$ and $-3.5B$ for the first and the second auxiliary neuron, respectively, where B is some constant. Both auxiliary neurons should connect to those neurons involved in the clause (in total to 3 neurons), with bidirectional connections w_{OR} and $-B$ (to and from the first auxiliary neuron, respectively), and $-w_{OR}$ and B (to and from the second auxiliary neuron, respectively). Finally, the first auxiliary neuron connects to the second one with strength $3B$.

Therefore, the total number of neurons needed to implement a general 3-SAT problem in a spiking neural network is 3#variables + 2#clauses (2 + 1 per WTA circuit, and 2 per OR circuit), while the number of connections is 4#variables + 13#clauses. Notably, both the number of neurons and the number of synapses depend linearly on the number of variables (the number of clauses linearly depends on the number of variables if problems with some fixed clauses-to-variables ratio are considered).

At any point in time the principal network state $\mathbf{x}$ is defined based on the activity of principal neurons within the last τ time units. If exactly one of the two neurons which code for a RV $\mathbf{X}_i$ is active at some time t, then the variable has a properly defined value. The WTA circuit for each RV ensures that this is the case most of the time for most problem variables. When this is the case, one can simply read off the current assignment of values to the RVs from the network state. Any clause is considered satisfied if at least one of the three neurons, which correspond to the three literals of the clause, is active.

To calculate the current performance of a solution at any point in time we use as a performance measure the ratio between the number of satisfied clauses and total number of clauses. If none of the variables which take part in a clause are properly defined then that clause is considered unsatisfied. As a result, this performance measure is well-defined at any point in time.

In order to implement the lock-in mechanism we use two working regimes which differ in the temperature of the network. While the first one is the normal regime during which the network normally explores possible solutions, the second one is the regime of decreased temperature during which the network locks into the current state (solution) and is very unlikely to escape from it. To implement the second regime we add for each clause two additional auxiliary neurons which are connected in the same way as the original auxiliary neurons (they target the same neurons) but with different weights: w_{OR2} and $-B$ (to and from neuron), and $-w_{OR2}$ and B(to

and from neuron), for the first and second additional auxiliary neuron, respectively. In addition their biases are set to $-0.5B$ and $-6.5B$ (first and second aux. neuron).

These additional auxiliary neurons are activated (i.e. functional) only when a certain state (the solution) was detected, which is signaled by the global *lock-in* neuron with bias b_{glob} that is connected to both additional auxiliary neurons of all clauses with connection strengths B and $3B$ to the first and the second additional auxiliary neuron, respectively. Additionally, the global *lock-in* neuron is connected to every other principal neuron with connection strength w_{glob}. This global neuron is active by default due to the high bias, but is deactivated whenever one of the status neurons, which check if a certain clause is not satisfied, is active (*not OK* signals). There is one status neuron for each clause, with bias set to $-2.5B$. The status neuron receives excitatory connections from all neurons corresponding to inverted literals of the clause, with strength B. Therefore, if all RVs that participate in the clause are set to the wrong values, this triggers the status neuron which reports that the clause is not satisfied. This automatically shuts down the global neuron signaling that the current network state is not a valid solution.

To implement this lock-in mechanism one needs additional 3#clauses+1 neurons and 2#variables + 20#clauses synapses.

The architecture described above was used throughout with the following parameters: $\tau = 10e-3$ and refractory period of 10ms for all neurons except for the global neuron which has $\tau = 9e-3$ and refractory period of 9ms, $b_{nrn} = 2$, $b_{inh} = -10$, $b_{glob} = 10$, $B = 40$, $w_{inh} = -100$, $w_{exc} = 100$, , $w_{inh} = -100$, $w_{OR} = 2.5$, $w_{OR} = 10$, with rectangular PSPs of 10ms duration without transmission delays for all synapses except for the one from the global neuron to the additional auxiliary neurons where the duration is 11ms.

For the analysis in Figure 3.2F of problem size dependence we created 3-SAT problems of different sizes with clause-to-variable ratio of 4.3. To ensure that a solution exists, each of the created problems was checked for satisfiability with zhaff, a freely available 3-SAT solver (Fu et al., 2004).

Details to TSP application (Figure 3.3)

For finding the shortest route for a TSP problem consisting of N_{cities} cities and $N_{resting}$ additional resting steps one needs in total $N_{\text{cities}} + N_{resting}$ variables, where each RV codes for the city visited at a certain step s. To solve TSP problems one needs to consider three types of constraints: (a) each RV must be properly defined, i.e. exactly one city must be visited at each step s. In addition, (b) each value of a variable must appear at least once in all RVs. At the same time only neighboring variables (those coding for consecutive steps) can have the same values (this allows for "resting" steps). Finally, (c) the penalty (in terms of additional energy) that two consecutive variables appear in a given configuration, i.e. a particular transition from city i to some other city j, must reflect the traveling cost between the pair of cities.

Based on Principle 2 these constraints can be implemented in a spiking neural network by forming circuits and interactions between neurons, where each of $N_{\text{cities}}+N_{resting}$ variables with N_{cities} different values can be represented by N_{cities} neurons with bias b_{nrn} each of which represents one city. To implement variable constraints (a) it is enough to form a WTA circuits from neurons that code for different values of the same variable. The WTA is formed in the same way as described for 3-SAT problem by taking corresponding neurons (here WTA circuits have N_{cities} principal neurons). To force the network to visit some desired city at a particular step it is sufficient to set the biases of those neurons which code for the desired city at in the WTA circuit of that step to different values. In particular, the desired value in the WTA circuit of that step is set to b_P and all others to b_N.

The implementation of constraints (b) requires that all variables have different values except if they are neighboring variables. In the spiking network implementation this can be realized by adding negative connections of strength w_{unique} from each neuron that codes for a certain value in a variable to all other neurons which code for the same value in other variables, except for the neighboring variables. This simply prevents, or decreases the chance, that two particular variables have the same value except if they are neighboring variables.

Finally, constraints (c) can be implemented by adding connections between all the neurons which code for two consecutive variables. This results in a network with a ring structure (as the last and the first variable are also connected). We chose to encode weights of these connections such that they reflect the relative distances between cities. To calculate the weights we normalize all the distances with respect to the maximum distance (this procedure applies also for asymmetric problems) and then we rescale and shift them according to $w = w_{offset} + (1 - w_N) * w_{scale}$, where w_N are normalized weights in $[0, 1]$ range.

Such architecture requires $(N_{\text{cities}}+1) * N_{resting}$ neurons and $N(3N_{resting}-1)+(N_{\text{cities}}-1) * (N_{\text{cities}}-1) * N_{resting}$ number of synapses.

Reading out the current assignment to a variable can be done based on the activity of the principal neurons which take part in the WTA circuit (same as for 3-SAT). Note that in this case each variable has N_{cities} values and therefore it multiple neurons within the same WTA could be active. When this happens, the value of the associated RV is briefly undefined. The performance of the network at some time is calculated as the ratio of the optimal path and the current path represented by the network. In order for the currently represented path to be valid all variables have to properly defined and each value (city) has to appear at least once. Although this is not *always* the case, exceptions occur rarely and therefore are not visible in performance plots.

For solving symmetric planar TSP experiments in Figure 3.3 we used the following setup: $\tau = 20e-3$ and refractory period of 20ms for all neurons, $b_{nrn} = -0.3$, $b_P = 100$, $b_N = -100$, $b_{inh} = -10$, $w_{inh} = -100$, $w_{exc} = 100$, $w_{unique} = -14.2$, $w_{scale} = 20.8$, $w_{offset} = -6.6$, $N_{resting} = 3$, with rectangular PSPs of 20ms duration

without transmission delays for all synapses. The value of the first variable (the first step) was fixed to the first city.

For solving asymmetric TSP problems we used the same architecture but slightly different parameters: $b_{nrn} = 1.3$, $w_{unique} = -14.1$, $w_{offset} = -7.9$, $N_{resting} = 8$.

For the comparative analysis between asymmetric and symmetric sampler (Neural Sampling (NS) vs. Boltzmann machine (BM)) we used exactly the same weights, biases and architecture as described above. The only difference here was that no inhibitory neurons were used, so that WTAs were implemented via direct inhibition connections between neurons taking part instead of bi-synaptic connections via inhibitory neurons.

The comparison of the number of state changes between BM and NS implementations was done based on 100 runs, each of which was simulated for 100.000 state changes. In each run and after every state changes we evaluated the current network state, and checked how many RVs were properly defined or not. Combining all runs in each case, we calculated how often transitions occurred in each sampler to states with different numbers $N_{undef} = 0, \ldots, N_{\text{cities}} + N_{resting}$ of undefined RVs. Based on this information we constructed corresponding histograms for BM and NS. To highlight the differences between the two implementations, we calculated the ratios between the normalized histogram values for NS and BM (Figure 3.3C). For the convergence speed comparison, in each run we calculated after each state change the cumulative minimum and mean performance during the whole time leading up to that state change. This was first done for each of the 100 network runs individually. The results were then averaged for each number of steps over all runs.

Appendix D

Appendix to Chapter 4: Emergence of Optimal Decoding in Stochastic Winner-Take-All Networks through STDP

Contents

D.1 Spike-timing dependent plasticity rules

All STDP rules used in this article (see Figure 4.3 for a visual comparison) fit into the phenomenological framework of STDP rules by Gerstner and Kistler (2002). Using their notation, $S_j(t) = \sum_f \delta(t - t_j(f))$ and $S_k(t) = \sum_f \delta(t - t_k(f))$ denoting

pre- and post-synaptic spike trains, respectively, STDP can be expressed as,

$$\frac{d}{dt}w(t) = \eta\, S_j(t) \left[\int_0^\infty a_2^{\text{pre,post}}(s,w)\, S_i(t-s)\, ds\right] \tag{D.1}$$

$$+\, \eta\, S_i(t) \left[a_1^{\text{post}}(w) + \int_0^\infty a_2^{\text{post,pre}}(s,w)\, S_j(t-s)\, ds\right] \quad , \tag{D.2}$$

where $a_2^{\text{pre,post}}(s,w)$ and $a_2^{\text{post, pre}}(s)$ denote the pre-before-post and post-before-pre kernels, respectively, and $a_1^{\text{post}}(w)$ the post-only contributions. The kernels are given by (double-)exponential decays throughout this article,

$$a_2^{\text{pre,post}}(s,w) = A_+(w) \cdot [\exp(s/\tau_+) - \exp(s/\tau_{+\text{rise}})] \quad , \tag{D.3}$$

$$a_2^{\text{post,pre}}(s,w) = A_-(w) \cdot [\exp(s/\tau_-) - \exp(s/\tau_{-\text{rise}})] \quad . \tag{D.4}$$

Post-before-pre effects were weight-independent for all tested rules, i.e. $A_-(w) \equiv -1$. The remaining parameters and functions are specified in Table D.1 for each of the four rules used in this article.

Rules	τ_+ ($\tau_{+\text{rise}}$)	$A_+(w)$	τ_- ($\tau_{-\text{rise}}$)	$a_1^{\text{post}}(w)$	noise
A. Optimal rule (5)	$20ms$ (2)	αe^{-w}	$0ms$	-1	no
B. Optimal with syn. scal.	$20ms$ (2)	$\alpha w e^{-w}$	$0ms$	$-w$	no
C. Longer causal window	$40ms$ (4)	$\alpha e^{-w}/2$	$0ms$	-1	no
D. Common STDP curve	$20ms$ (0)	$e^{\beta-\gamma w}$	$60ms(0)$	0	yes

Table D.1: Comparison of investigated STDP rules (labels A-D as in Figure 4.3). $\tau_- = 0ms$ in A-C corresponds to the absence of spike-timing dependent depression. For these rules, weight stabilization is achieved through the homeostatic plasticity term $a_1^{\text{post}}(w)$ which is triggered for every post-synaptic spike.

We set $\alpha = 4.3$, $\beta = 1.58$ and $\gamma = 0.59$ for the simulations in the main text. Figure D.1 shows the impact on performance of varying β and γ. For the common STDP curve (Figure 4.3D), additional zero-mean Gaussian noise was superimposed for each learning update triggered by a post-synaptic spike. The standard deviation σ_{STDP} of this noise was dependent on the magnitude of the deterministic update: for each deterministic update of magnitude M, Gaussian noise with standard deviation $\sigma_{\text{STDP}} = \kappa\, M + \zeta$ was added. The noise parameters were set to $\kappa = 0.3$ and $\zeta = 10^{-4}$. The optimal rule (4.5) was used for simulations in Figure 4.2-4.6. The rules B-D were used in Figure 4.3.

D.2 Implicit generative model

Neural activity and synaptic learning in the considered cortical microcircuit model can be understood from the perspective of an underlying implicit generative model. This generative model takes the form of a mixture model with $k \in \{1, \ldots, N\}$

hidden causes, uniform priors and conditionally independent Poisson variables x_j:

$$p(\mathbf{x}|\mathbf{W}) = \frac{1}{N}\sum_{k=1}^{N} p(\mathbf{x}|k,\mathbf{w}_k) \ , \tag{D.5}$$

$$p(\mathbf{x}|k,\mathbf{w}_k) = \prod_{j=1}^{M} \text{Poisson}(x_j;\lambda_{kj}=\alpha^{-1}e^{w_{kj}}) \tag{D.6}$$

$$= h(\mathbf{x})\cdot\exp(\mathbf{w}_k^T\mathbf{x} - A(\mathbf{w}_k)) \ . \tag{D.7}$$

where $A(\mathbf{w}_k) = \alpha^{-1}\sum_j e^{w_{kj}}$, and $h(\mathbf{x}) = \prod_j \frac{1}{x_j!}$. We associate each hidden cause k with one readout neuron, k, and the parameters w_{kj} with the synaptic weights of the network. This allows to relate inference and learning in the generative model to the operation of the microcircuit.

D.3 Stochastic Winner-Take-All (WTA) circuit and inference

Building on Nessler et al. (2010), we show here that the WTA circuit in the cortical microcircuit implements inference in the implicit generative model defined by (D.5)-(D.7). According to Bayes' rule, inferring the hidden cause k from input data $\mathbf{x}$ in the generative model can be written as,

$$p(k|\mathbf{x},\mathbf{W}) = \frac{\exp(\mathbf{w}_k^T\mathbf{x} - A(\mathbf{w}_k))}{\sum_l \exp(\mathbf{w}_l^T\mathbf{x} - A(\mathbf{w}_l))} \ . \tag{D.8}$$

In case of a homogeneous input representation, e.g. a population code in which the sum of sensory activations are constant $\sum_j x_k = A_0$, this reduces to ,

$$p(k|\mathbf{x},\mathbf{W}) = \frac{\exp(\mathbf{w}_k^T\mathbf{x})}{\sum_l \exp(\mathbf{w}_l^T\mathbf{x})} \ . \tag{D.9}$$

Note that the restriction $\sum_j x_k = A_0$ is necessary to make the theory tractable. However, as demonstrated in the simulations through this article (Figure 4.2-4.6) in which sensory input neurons fire randomly and only the total input population *rate* was kept constant as in Jazayeri and Movshon (2006), this is not required for the functionality of the model in practice.

Now, consider a population of stochastically spiking readout neurons k which fire at an instantaneous rate $\rho_k = e^{u_k}$, depending on their current membrane potential $u_k = \mathbf{w}_k^T\mathbf{x} - I$. We make two basic assumptions about the inhibitory contribution. First, that I is common to all readout neurons. Second, that the inhibitory circuit which provides I ensures that an approximately constant total target firing rate $\rho_{total} = \sum_k \rho_k$ is maintained. This form of divisive inhibition introduces competition among the readout neurons, since a strongly activated readout neuron

will claim a large fraction of the total target firing rate, thereby suppressing other readout neurons.

In an ideal stochastic WTA circuit, the inhibitory circuit keeps ρ_{total} constant. The ideal inhibitory contribution is given by,

$$I = \log \sum_j \exp(\mathbf{w}_j^T \mathbf{x}) - \log \rho_{total} \quad . \tag{D.10}$$

Then, the readout neuron k will respond to an input $\mathbf{x}$ with an instantaneous rate,

$$\rho_k = \rho_{total} \cdot \frac{\exp(\mathbf{w}_k^T \mathbf{x})}{\sum_l \exp(\mathbf{w}_l^T \mathbf{x})} \quad . \tag{D.11}$$

By comparison with (D.9), one can verify that this is proportional to the posterior probability of the hidden cause k in the generative model. Note that, since the relative spiking probabilities match the posterior probabilities, each spike produced by the readout circuit can be interpreted as a sample from the posterior distribution, independent of the total firing rate ρ_{total} of the circuit.

D.4 Equilibria of theoretically optimal STDP rules and maximum likelihood

Here we discuss two important results. First, we derive conditions for the equilibrium points of STDP: stable weight settings for which the expected STDP update is zero. The significance of these points derives from the fact that they are attractors in the weight dynamics. Hence, STDP will always drive the weights into the neighborhood of such an equilibrium point. Second, we show that these equilibria also have a special interpretation from the generative model perspective: they correspond to locally optimal parameter settings for the generative model in the maximum likelihood sense.

An equilibrium point of STDP is, by definition, invariant to the average STDP update. Our first goal here is to derive from this global definition more concrete statements for single synapses. To this end, we first define the *empirical joint distribution* over $\mathbf{x}$ and the activation of a readout neuron, $z_k \in \{0, 1\}$:

$$\tilde{p}(\mathbf{x}, z_k | \mathbf{W}) = p^*(\mathbf{x}) \cdot p(z_k | \mathbf{x}, \mathbf{W}) \quad . \tag{D.12}$$

where $p^*(\mathbf{x})$ is the input distribution, and $p(z_k = 1 | \mathbf{x}, \mathbf{W}) = p(k | \mathbf{x}, \mathbf{W})$, using the fact that the WTA circuit implements inference according to the generative model.

Then, at equilibrium the following must hold for all k and j,

$$\langle \Delta w_{kj} \rangle_{\tilde{p}(\mathbf{x}, z_k | \mathbf{W})} = 0 \quad , \tag{D.13}$$

where $\langle\cdot\rangle_{q(\cdot)}$ denotes the average (or expectation) taken over the distribution $q(\cdot)$.

After substituting the STDP learning rule specified in (4.5) and rearranging terms, one obtains

$$w_{kj} = \log\alpha + \log\langle x_j\rangle_{\tilde{p}(\mathbf{x}|z_k=1,\mathbf{W})} \quad . \tag{D.14}$$

Indeed, this result is obtained for any learning rule from the optimal family (4.11). Hence, at equilibrium, each synaptic weight w_{kj} is set to the logarithm of the average pre-synaptic activity x_j, the average taken over those input patterns which make the post-synaptic neuron k fire (plus some constant).

The relation of this result to the generative model becomes apparent when considering the maximization of the marginal likelihood $\langle\log p(\mathbf{x}|\mathbf{W})\rangle_{p^*(\mathbf{x})}$, which is equivalent to the minimization of the KL-divergence $\mathrm{KL}(p^*(\mathbf{x})||p(\mathbf{x}|\mathbf{W}))$. For local maxima of the log-likelihood $\langle\log p(\mathbf{x}|\mathbf{W})\rangle_{p^*(\mathbf{x})}$ one has for all k and j,

$$\partial_{w_{kj}}\langle\log p(\mathbf{x}|\mathbf{W})\rangle_{p^*(\mathbf{x})} = \tag{D.15}$$

$$\langle\frac{1}{p(\mathbf{x}|\mathbf{W})}\partial_{w_{kj}}\sum_k p(\mathbf{x},k|\mathbf{W})\rangle_{p^*(\mathbf{x})} = \tag{D.16}$$

$$\langle\frac{1}{p(\mathbf{x}|\mathbf{W})}\sum_k p(\mathbf{x},k|\mathbf{W})\partial_{w_{kj}}\log p(\mathbf{x},k|\mathbf{W})\rangle_{p^*(\mathbf{x})} = \tag{D.17}$$

$$\langle\sum_k p(k|\mathbf{x},\mathbf{W})\,\partial_{w_{kj}}\log p(\mathbf{x},k|\mathbf{W})\rangle_{p^*(\mathbf{x})} = \tag{D.18}$$

$$\langle\partial_{w_{kj}}\log p(\mathbf{x}|k,\mathbf{W})\rangle_{\tilde{p}(\mathbf{x},z_k=1|\mathbf{W})} = 0 \quad . \tag{D.19}$$

By evaluating the derivative, one can verify that this is indeed equivalent to (D.14). Hence, all STDP equilibrium points are automatically local optima with respect to the implicit generative model.

D.5 Link between optimal STDP and Expectation Maximization

EM is a powerful and widely used algorithm for optimizing generative models and extracting hidden causes from high-dimensional input data. In the operating microcircuit model, an online version of this algorithm can be identified. As we showed above, each time a readout neuron fires, a sample from the posterior distribution is drawn. Together with the current input pattern, this sample forms a pair $(\mathbf{x}, k)$. Theoretically optimal STDP (4.5), or more generally any rule from the family (4.11), then increases the log-likelihood $p(\mathbf{x}, k|\mathbf{W})$ of this pair in the model. The main difference to standard EM is the fact that STDP can only access temporally local information (at least the simple model considered here). Hence, just as in online EM (Sato, 1999), STDP does not maximize the likelihood over the whole dataset in each step, but rather makes a small update in the right direction after each

post-synaptic spike. To show this, consider the expected STDP update $\overline{\Delta w_{kj}}$ for a synapse, if the input distribution is $p^*(\mathbf{x})$:

$$\overline{\Delta w_{kj}} = \eta\, f(w_{kj})\, \langle p(k|\mathbf{x},\mathbf{W})(x_j e^{-w_{kj}} - \alpha^{-1})\rangle_{p^*(\mathbf{x})} \quad , \tag{D.20}$$

where η is some small learning rate. At the same time, the log-likelihood gradient with respect to that synaptic weight is,

$$\partial_{w_{kj}} \langle \log p(\mathbf{x}|\mathbf{W})\rangle_{p^*(\mathbf{x})} = e^{w_{kj}} \left\langle p(k|\mathbf{x},\mathbf{W})(x_j e^{-w_{kj}} - \alpha^{-1})\right\rangle_{p^*(\mathbf{x})} \tag{D.21}$$

Comparing (D.20) and (D.21) one can see that the expected update of STDP drives the weights towards a skewed version of the true gradient. Importantly, the dot product between update direction and gradient is strictly non-negative:

$$\sum_{k,j} \overline{\Delta w_{kj}} \cdot \partial_{w_{kj}} \langle \log p(\mathbf{x}|\mathbf{W})\rangle_{p^*(\mathbf{x})} \geq 0 \tag{D.22}$$

This means that the average STDP update will never decrease the performance of the generative model. In fact, one can easily verify that it will always increase performance, unless the weight setting already constitutes a local optimum of the log-likelihood (see also Appendix D: Equilibria of STDP). Altogether, this implies that the average effect of STDP can be described as an attractor dynamics in the weight space, in which the attractors are the equilibrium points of STDP and at the same time the local optima of the generative model. Stochastic deviations from these dynamics are zero-mean, and can thus be suppressed to arbitrary precision via the learning rate η (at the cost of convergence speed).

D.6 Computer simulations

To abbreviate explanations, computer experiments will be referred to by numbers: E1 (Figure 4.2), E2 (Figure 4.2), E3 (Figure 4.4), E4 (Figure 4.5) and E5 (Figure 4.6). The simulation time step throughout the experiments was $\Delta t = 2.5$ms. The external stimulus $\theta(t)$ was generated by a random noise process, in which the value of θ changed every 100ms, according to,

$$\theta(t) = (\theta(t - \Delta t) + \mathcal{N}(0,1)) \bmod 2\pi \quad , \tag{D.23}$$

where we used the modulo operation on real numbers to enforce periodic boundary conditions on θ.

Each sensory neuron $j \in \{1,\ldots,M\}$ was assigned a tuning function $f_j(\theta)$, determining the firing rate of neuron j in response to stimulus θ. For the experiments E1, E2, E4 and E5, the tuning functions were given by,

$$f_j(\theta) = c \cdot \exp(k(\cos(\theta - \Theta_j) - 1)) \quad , \tag{D.24}$$

where each sensory neuron was associated with a different preferred stimulus Θ_j, equally spaced over $[0, 2\pi]$ across the population. The concentration factor k regulates the sharpness of the tuning function, and was chosen $k = 1$ for E1, E2, E4 and E5. In all experiments, the scaling constant was set to $c = 40$, yielding a maximal firing rate of 40 Hertz for the preferred stimulus.

The final tuning functions for the groups G1 and G2 in E3 were set up with concentration parameters $k = 0.5$ and $k = 3$, respectively. G3 developed a bi-modal tuning function,

$$f_j(\theta) = c \cdot \exp(k(\cos(\theta - \Theta_j) - 1)) + c \cdot \exp(k(\cos(\theta - \Theta_j - 2) - 1)) \ . \quad \text{(D.25)}$$

with $k = 6$. The transition between initial and final tuning functions in E3 was done by linear interpolation of the tuning functions.

Input spike trains were generated as inhomogeneous Poisson processes with instantaneous rates $f_j(\theta(t))$. The resulting spike trains were then filtered by the shape of a double-exponential EPSP,

$$\epsilon(t) = D \cdot (\exp(-t/\tau_1) - \exp(-t/\tau_2)) \ , \quad \text{(D.26)}$$

to yield the filtered unweighted spike trains $x_j(t)$. The constant D was chosen such that $x_j(t)$ has equal mean and variance for a fixed firing rate of the sensory neuron. The EPSP time constants were $\tau_1 = 20$ms and $\tau_2 = 2$ms.

Synaptic weights w_{kj} were initialized by drawing independently from a normal distribution with mean μ_w and standard deviation $\sigma_w = 0.1$, where we used $\mu_w = 2$ for E1, E2, E4 and E5, and $\mu_w = 1.81$ for E3. The filtered spike trains $x_j(t)$ were then used as inputs to the readout neurons. The WTA circuit was implemented according to a discrete time approximation of (D.11): at each discrete time step, the firing probability for each neuron k is given by $\Delta t \cdot \rho_k(t)$. A spike occurs for neuron k if a draw from a Bernoulli distribution with this probability is successful. In all simulations, the total firing rate ρ_{total} of the WTA circuit was chosen beforehand to achieve an average firing rate of 3 Hertz per readout neuron (in E4, ρ_{total} was computed for $N = 15$ and kept constant throughout the simulation). Whenever a readout neuron spiked, STDP was applied according to (4.5). The learning rate was set to $\eta = 0.001$ in E1, E3, and E4, and to $\eta = 0.0005$ for E2. For E5, the learning rate $\eta(t)$ depended on $\theta(t)$ as indicated in Figure 4.6.

At regular intervals, the preferred stimuli $\psi_k(t)$ of readout neurons were computed. This was done in a separate offline simulation, in which we swept over all input stimuli and selected for each readout neuron k the stimulus θ which elicited the greatest average response from that neuron. As a performance indicator, a reconstruction of the input signal was computed from the output spikes (shown in Figure 4.2E). First we computed a raw reconstruction signal $\hat{\theta}_{rec}(t)$ which jumps to the preferred stimulus ψ_k of a readout neuron k whenever z_k emits a spike. From this we obtained the reconstruction $\theta_{rec}(t)$ as the population vector average of the raw reconstruction values $\hat{\theta}_{rec}(t)$ within a $[t - 20\text{ms}, t]$ window. After training, in

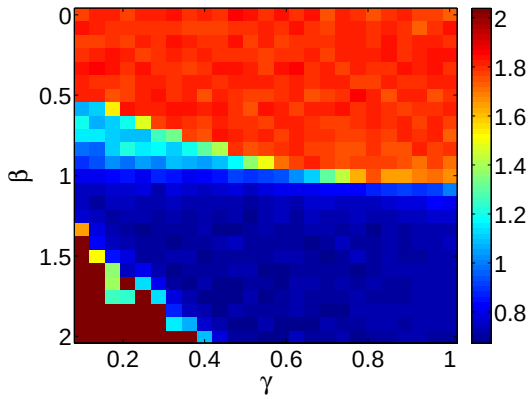

Figure D.1: STDP curve parameter variations. Two tuning parameters β, γ of the STDP learning curve in Figure 4.3D were varied to evaluate its robustness with respect to parameter variations. Shown is RMSE performance after 3000s of learning in the network setup of Figure 4.3.

E1, E2, E3, and E5 readout neurons were sorted for visualization according to their preferred stimuli at the end of the simulation. In E4 sorting was done individually for each visualization point, since different neurons were involved at each time.

D.7 STDP parameter variations

In simulations for Figure 4.3D, an STDP learning curve with standard spike-timing dependent depression was considered. Two parameters of this STDP rule, κ and ζ, define noise components while two others β, γ control weight-dependence and magnitude of the LTP part (see Appendix D: STDP rules). To evaluate not only performance but also stability of this rule, an array of values for β, γ were evaluated with respect to the reconstruction RMSE. As shown in Figure D.1, the performance of the STDP rule is remarkably stable with respect to changes in these parameters, and there exists a continuum of parameter settings which yield near-optimal performance.

D.8 Effect of noise correlation

Simulations in the main text were performed with uncorrelated inputs, i.e. zero noise correlation among input neurons. To study the effect of noise correlation we created a correlated input population code (Figure D.2) in the following simple manner: we introduced a hidden state variable $S \in \{0, 1\}$ which modulates the input tuning functions $f_j(\theta)$ (without the network having access to this variable). S changes randomly every 100ms. For $S = 1$, the tuning functions are modulated according to $f_j^{\mathrm{S}=1}(\theta) = m_j\, f_j(\theta)$, with the fixed modulation con-

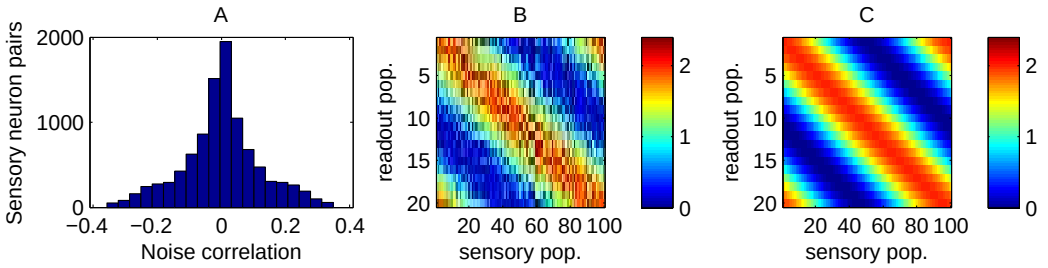

Figure D.2: Effect of noise correlation on learning. **A**. Noise correlations between sensory neurons in a correlated population code. **B**. Weight matrix after learning. **C**. Weight matrix which would be optimal for decorrelated inputs (but is in general sub-optimal for the correlated case).

stants m_j drawn randomly for each sensory neuron from a uniform distribution on $[0.4, 1.6]$. Furthermore, if $S = 0$, tuning functions are modulated according to $f_j^{S=0}(\theta) = (2 - m_j)\, f_j(\theta)$. This ensures that the sensory inputs become correlated (Figure D.2), while on average the original tuning functions $f_j(\theta)$ are preserved (i.e. identical to those in Figure 4.2).

The resulting weight matrix after 2000s of learning is slightly more noisy than for decorrelated inputs (Figure D.2B, RMSE 0.7), and resembles the weights which would be optimal for decorrelated inputs (Figure D.2C, RMSE 0.69). In general, however, it is known that linear decoding weights can be further improved if input correlations (when they exist) are taken into account during learning (Graf et al., 2011). Although we did not implement the supervised learning algorithm of (Graf et al., 2011) to test this possibility, we would expect to find a slight improvement over the weights found through (unsupervised) STDP.

Non-uniform stimulus distribution

The simulations described in the main text were performed with a uniformly distributed input stimulus θ. To examine the impact of non-uniform distributions on the learning process, a normal distribution (Figure D.3A) was tested in the setup of Figure 4.2 (with more WTA neurons (100), and a smaller learning rate $\eta = 0.0005$). As shown in Figure D.3B, after learning automatically more WTA neurons specialized on high-probability patterns. In fact, the number of neurons which specialized on a particular stimulus turns out to be proportional to the probability of that stimulus in the input distribution (Figure D.3C). This demonstrates that through learning not only the likelihood but also the implicit prior over stimuli becomes automatically adapted to the input distribution, in accordance with the predictions of a Bayesian framework.

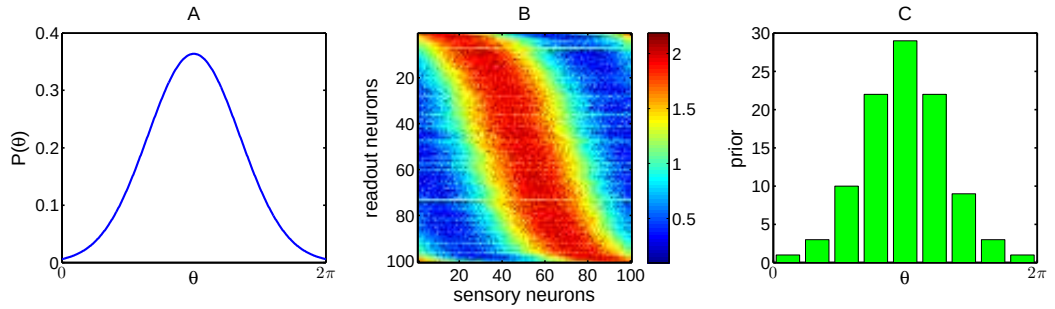

Figure D.3: Non-uniform stimulus distribution. **A**. A normally distributed input stimulus was used. **B**. Weight matrix after 2000s of learning. A higher occurrence rate of the input stimulus around π led to improved specialization of readout neurons on this region. **C**. Implicit prior (computed from the histogram of preferred stimuli of WTA neurons) after learning has correctly adapted to the stimulus distribution.

D.9 Inference with a homogeneous input representation

In (Jazayeri and Movshon, 2006) it was shown that log-likelihoods in a Poisson model effectively reduce to a weighted sum of inputs if the (static) input model is homogeneous, i.e. the average sensory population response to any stimulus sums to a constant. Here, we will show that a similar result holds when the model is not static but learned during experience.

For a Poisson model the log-likelihood can be written as,

$$\log p(\mathbf{x}|k, w_{kj}) = \sum_j w_{kj} x_j - \sum_j A(w_{kj}) + \sum_j \log h(x_j) \quad , \tag{D.27}$$

with $A(x_{kj}) = \alpha^{-1} e^{w_{kj}}$ and $h(x_j) = (x_j!)^{-1}$. The third term does not depend on the parameters and can be safely ignored during inference (it drops out in the WTA operation). The critical second term vanishes if and only if,

$$\sum_j e^{w_{kj}} = \alpha A_0 \quad , \forall k \in \{1, N\}, \tag{D.28}$$

for some constant A_0.

In a dynamic learning scenario these constraints may be not fulfilled initially. However, we can show that the STDP learning rule (4.5) automatically forces the weights to fulfill the constraints, if the real input distribution $p^*(\mathbf{x})$ is strictly homogeneous. We define strict homogeneity of $p^*(\mathbf{x})$ here as follows,

$$p^*(\mathbf{x}) > 0 \Rightarrow \sum_j x_j = A_0 \quad , \tag{D.29}$$

for some A_0, i.e. the sum of inputs must sum to constant.

Then it is easy to see from (D.14) that any equilibrium point of STDP fulfills (D.28), since for all k,

$$\sum_j e^w_{kj} = \alpha \frac{\langle q(z_k = 1|\mathbf{x}, \mathbf{W}) \sum_j x_j \rangle_{p^*(\mathbf{x})}}{\langle q(z_k = 1|\mathbf{x}, \mathbf{W}) \rangle_{p^*(\mathbf{x})}} = \alpha A_0 \quad . \tag{D.30}$$

Note that this is true regardless of the network output behavior $q(z_k|\mathbf{x}, \mathbf{W})$, and in particular for the WTA dynamics considered here:

$$q(z_k = 1|\mathbf{x}, \mathbf{W}) = \frac{e^{\mathbf{w}_k^T \mathbf{x}}}{\sum_l e^{\mathbf{w}_l^T \mathbf{x}}} \quad . \tag{D.31}$$

Hence, if the input is strictly homogeneous, STDP will automatically drive the weights to the constraint set and the simple WTA dynamics (D.31) will perform correct inference in terms of the generative model. In simulations we approximated strict homogeneity by a "homogeneous rate" representation (the average total activation of sensory neurons is independent of the stimulus θ) which worked well in practice.

D.10 Optimal STDP rules for non-Poisson input statistics

Poisson firing variability is a reasonable first assumption for input neurons in the considered cortical microcircuit model. Yet, not all neurons in cortex are well described by such a model. In fact, electrophysiological studies indicate that the brain is populated by neurons covering a whole range of firing characteristics, from bursty to regularly firing neurons (Shinomoto et al., 2009). Is the present theory also consistent with non-Poisson statistics? If yes, what are the necessary changes in the STDP rules predicted by theory? Here we show that a) there exists a large family of distributions consistent with our theory, and b) each type of distribution predicts a slightly different weight-dependence of STDP.

In (D.7) we assumed a Poisson model for the inputs of the network. An interesting property of the Poisson distribution is that it factorizes into three separate objects,

$$p(x_j|k, w_{kj}) = h(x_j) \cdot \exp(w_{kj} x_j) \cdot \exp(-A(w_{kj})) \quad , \tag{D.32}$$

where for Poisson one has $A(w_{kj}) = \alpha^{-1} e^{w_{kj}}$, and $h(x_j) = \frac{1}{x_j!}$. However, in principle one could plugin other functions $A(\cdot)$ and $h(\cdot)$ here to obtain different distributions, e.g. to model other firing statistics. The class of all distributions which can be written in this form is called natural exponential family(Bishop, 2006). Some prominent examples are the normal distribution with fixed variance, the Gamma distribution

with fixed shape, and –of course– the Poisson distribution. In the natural exponential family form, these different distributions are uniquely characterized by their "log-partition function" $A(\cdot)$.

From a theoretical perspective, the gradient of the log-likelihood with respect to a synaptic weight w_{kj} is then given by,

$$\partial_{w_{kj}} \langle \log p(\mathbf{x}|\mathbf{W}) \rangle_{p^*(\mathbf{x})} = \left\langle p(k|\mathbf{x}, \mathbf{W})(x_j - A'(w_{kj})) \right\rangle_{p^*(\mathbf{x})} \tag{D.33}$$

Using this observation, consider the following STDP rule for a natural exponential family,

$$\Delta w_{kj} \propto \begin{cases} \frac{x_j - A'(w_{kj})}{A''(w_{kj})} & \text{if hidden cause } k \text{ is active,} \\ 0 & \text{otherwise.} \end{cases} \tag{D.34}$$

Similar to the Poisson case then, the expected application of this STDP rule will never decrease the performance of the generative model, since the dot-product between update and the performance gradient is non-negative:

$$\sum_{k,j} \overline{\Delta w_{kj}} \cdot \partial_{w_{kj}} \langle \log p(\mathbf{x}|\mathbf{W}) \rangle_{p^*(\mathbf{x})} \geq 0 \tag{D.35}$$

Analogous to (D.28) for the Poisson model, the theory also requires the weights to fulfill normalization constraints,

$$\sum_j A(w_{kj}) = A_0 \ \ , \forall k \in \{1, N\}, \tag{D.36}$$

for some constant A_0, such that inference as performed by the WTA circuit is correct. In practice, this could be achieved by additional heterosynaptic plasticity acting on all synaptic weights w_{kj} of the same readout neuron k.

An interesting variation of (D.34) features a constant negative decay, as has been observed in experimental studies on STDP (Jacob et al., 2007):

$$\Delta w_{kj} \propto \begin{cases} \frac{x_j}{A'(w_{kj})} - 1 & \text{if hidden cause } k \text{ is active,} \\ 0 & \text{otherwise.} \end{cases} \tag{D.37}$$

Note that the two learning rules (D.34) and (D.37) are theoretically equivalent in the sense of asymptotic convergence, but they may have slightly different properties in terms of convergence speed for different input distributions. Note also that for simplicity we assumed here that the WTA circuit performs correct inference. In general, this requires the that the weights can fulfill the constraints $\sum_j A(w_{kj}) = A_0 \ \ , \forall k \in \{1, N\}$ for any natural exponential family.

The latter formulation (D.37) facilitates the comparison of learning rules for different exponential families, since the rules differ only in a single factor $\frac{1}{A'(w_{kj})}$ of the positive STDP part. This factor is shown in Figure D.4 for four different exponential families. In Table D.2, five types of distributions are listed along with the associated STDP learning rules.

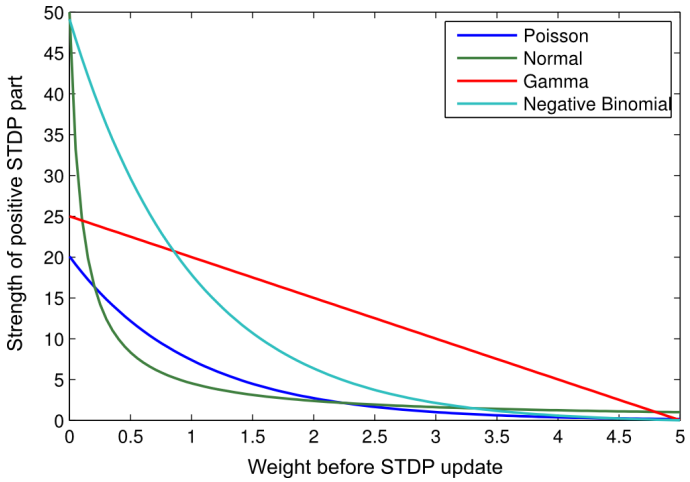

Figure D.4: The weight-dependence of the positive STDP part in (D.37), $\frac{1}{A'(w_{kj})}$, is shown for four different types of model: the Poisson distribution ($A(w_{kj}) = Ce^{-w_{kj}}$), the Normal distribution ($A(w_{kj}) = Cw_{kj}$), the Gamma distribution ($A(w_{kj}) = C/(w_0 - w_{kj})$) and the Negative Binomial distribution ($A(w_{kj}) = C/(e^{w_0 - w_{kj}} - 1)$), with respective parameters C and w_0 chosen to facilitate visual comparison. The scaling of the ordinate is arbitrary.

Model type	$A(w_{kj})$	Parameter range	Learning rule Δw_{kj}
Poisson	$Ce^{w_{kj}}$	$w_{kj} \in [0, \infty)$	$x_j(t) \cdot \frac{1}{C} e^{-w_{kj}} - 1$
Normal	$C(w_{kj} - w_0)$	$w_{kj} \in (0, \infty)$	$x_j(t) \cdot \frac{1}{C(w_{kj} - w_0)} - 1$
Gamma	$C/(w_0 - w_{kj})$	$w_{kj} \in [0, w_0)$	$x_j(t) \cdot \frac{w_0 - w_{kj}}{C} - 1$
Neg. Binomial	$-C \log(1 - e^{w_{kj} - w_0})$	$w_{kj} \in [0, w_0)$	$x_j(t) \cdot \frac{e^{w_0 - w_{kj}} - 1}{C} - 1$
Binomial	$C \log(1 + e^{w_{kj} - w_0})$	$w_{kj} \in [0, \infty)$	$x_j(t) \cdot \frac{e^{w_0 - w_{kj}} + 1}{C} - 1$

Table D.2: Comparison of natural exponential families with corresponding STDP rules

D.11 The relation between EPSP shape and optimal STDP rule

Similar to an external observer who counts spikes in fixed time window, a readout neuron "observing" a pre-synaptic input can detect the co-occurence of multiple input spikes on the same channel by adding them up. However, in contrast to the "rectangular" observation window often used in external analyses, neurons rely on the low-pass filtering properties of the membrane to achieve this task, giving rise to a smoothly decaying filtering (EPSP) shape. In the context of an implicit generative model implemented by readout neurons, this is relevant since the EPSP shape determines the observed variables and hence also the statistics $p^*(\mathbf{x})$. In particular, even if the input spike trains have Poisson characteristics, after filtering with a non-rectangular EPSP shape the observed, filtered variables $\mathbf{x}$ will be non-Poissonian. Here we show that these effects can be accounted for theoretically, by

deriving the optimal STDP rule in dependence of the EPSP shape.

We will assume here that input spike trains are generated by inhomogeneous Poisson processes with slowly varying rates, and consider piece-wise constant EPSP shapes $\epsilon(t)$ of finite length L.Suppose that there are n individual pieces with different magnitudes ϵ_i, each of length δ, such that $L = n \cdot \delta$. Formally,

$$\epsilon(t) = \sum_{i=0}^{n-1} \epsilon_i H(t - i \cdot \delta) H(-t + (i+1) \cdot \delta) \ , \tag{D.38}$$

where $H(\cdot)$ denotes the Heaviside step function. Then, the convolution of an input spike train $\xi(t)$ with such an EPSP can be written as,

$$x(t) = \int_0^L \xi(t-\tau)\epsilon(\tau)\, d\tau \tag{D.39}$$

$$= \sum_{i=0}^{n-1} \epsilon_i \int_0^\delta \xi(t - i \cdot \delta - \tau)\, d\tau \ . \tag{D.40}$$

Assuming a constant Poisson rate λ throughout the integration period, all sub-integrals in (D.40) will be Poisson distributed with ensemble mean and variance $\delta\lambda$. Let a single contribution to this sum be,

$$x^i(t) = \epsilon_i \int_0^l \xi(t - i \cdot \delta - \tau)\, d\tau \ . \tag{D.41}$$

Since sub-integrals are Poisson distributed, the ensemble mean and variance of $x^i(t)$ can be found easily as,

$$\mathrm{E}(x^i(t)) = \epsilon_i \delta\lambda \tag{D.42}$$

$$\mathrm{Var}(x^i(t)) = \epsilon_i^2 \delta\lambda \ . \tag{D.43}$$

Furthermore, since different $x^i(t)$ are independent, the sum of contributions is distributed with mean and variance,

$$\mathrm{E}(x(t)) = \sum_{i=0}^{n-1} \epsilon_i \delta\lambda = \lambda \sum_{i=0}^{n-1} \epsilon_i \delta = \lambda \int_0^L \epsilon(\tau)\, d\tau \ , \tag{D.44}$$

$$\mathrm{Var}(x(t)) = \sum_{i=0}^{n-1} \epsilon_i^2 \delta\lambda = \lambda \int_0^L \epsilon^2(\tau)\, d\tau \ . \tag{D.45}$$

In the context of unsupervised learning from data with exponential family distributions, this suggests an over- or under-dispersed Poisson model for $x(t)$. As detailed below, this results only in a small adjustment in our model. The so-called dispersion factor ϕ of the model can be derived from the EPSP shape as,

$$\phi = \frac{\int_0^L \epsilon^2(\tau)\, d\tau}{\int_0^L \epsilon(\tau)\, d\tau} \tag{D.46}$$

Then, using an exponential family with

$$A(w_{kj}) = \alpha^{-1} \exp(\phi w_{kj}) \ , \tag{D.47}$$

one obtains the learning rule,

$$\Delta w_{kj} \propto z_k \cdot (x_j \alpha e^{-\phi w_{kj}} - \phi) \ . \tag{D.48}$$

In summary, this shows that the optimal STDP rule depends on the EPSP shape in two different ways. First, through x_j which is filtered by the EPSP shape. It is easy to see that this leads to an STDP shape for pre-before-post pairings which must correspond to the EPSP shape. Second, through the dispersion factor ϕ which directly derives from the EPSP shape (D.46). Note that in the computer simulations of this article we scaled the EPSP such that $\phi = 1$, leaving only the first effect.

D.12 Convergence proof for natural exponential families

We argued that different STDP rules are optimal for different input statistics (Poisson, Gamma, etc.). Here we provide a convergence proof for a whole class of learning rules, including the local STDP rules (D.34) for natural exponential families with conditionally independent inputs.

Consider a mixture model with k exponential family components,

$$p(\mathbf{x} \,|\, \mathbf{w}, \boldsymbol{\pi}) = \sum_k \exp(\pi_k) p(\mathbf{x} \,|k, \mathbf{w}_k) \tag{D.49}$$

$$= \sum_k \exp(\pi_k) h(\mathbf{x}) \exp(\mathbf{w}_k^T \, \mathbf{x} - A(\mathbf{w}_k)) \tag{D.50}$$

$$= h(\mathbf{x}) \sum_k \exp(\underbrace{\pi_k + \mathbf{w}_k^T \, \mathbf{x} - A(\mathbf{w}_k)}_{v_k}) \tag{D.51}$$

$$= h(\mathbf{x}) \sum_k \exp(v_k) \ . \tag{D.52}$$

The normalization or log-partition function $A(\cdot)$ is a scalar function of the parameter vector $\mathbf{w}_k$, which decomposes into a sum for conditionally independent sufficient statistics $\mathbf{x}$.

The goal of generative modeling is to maximize the expected log-likelihood of the data under the model, $\langle \log p(\mathbf{x} \,|\, \boldsymbol{\pi}, \mathbf{w}) \rangle_{p^*(\mathbf{x})}$, here under the constraint $\sum_k e^{\pi_k} = 1$. To this end one can define the Lagrangian,

$$L(\boldsymbol{\pi}, \mathbf{w}, \lambda) = \langle \log p(\mathbf{x} \,|\, \boldsymbol{\pi}, \mathbf{w}) \rangle_{p^*(\mathbf{x})} - \lambda (\sum_k e^{\pi_k} - 1) \ , \tag{D.53}$$

and obtain critical points by requiring

$$\nabla L(\boldsymbol{\pi}, \mathbf{w}, \lambda) = \mathbf{0} \ . \tag{D.54}$$

At a local maximum this gives $\lambda = 1$. Using the fact that λ does not depend on parameters or inputs, and dropping terms which are independent of parameters, one can define a simplified objective function

$$l(\boldsymbol{\pi}, \mathbf{w}) = \langle \sum_k e^{v_k} \rangle_{p^*(\mathbf{x})} - \sum_k e^{\pi_k} \ , \tag{D.55}$$

for which it is easy to show that the local maxima coincide with the original constrained optimization goal. Maximization of $l(\boldsymbol{\pi}, \mathbf{w})$ then yields the following necessary conditions:

$$\pi_k = \log \langle p(k|\, \mathbf{x}, \mathbf{w}) \rangle_{p^*(\mathbf{x})} \ , \tag{D.56}$$

$$\nabla A(\mathbf{w}_k) = \frac{\langle p(k|\, \mathbf{x}, \mathbf{w})\, \mathbf{x} \rangle_{p^*(\mathbf{x})}}{\langle p(k|\, \mathbf{x}, \mathbf{w}) \rangle_{p^*(\mathbf{x})}} \ . \tag{D.57}$$

Now consider the following learning rules,

$$\Delta_{\pi_k} = \eta (z_k e^{-\pi_k} - 1) \ , \tag{D.58}$$

$$\Delta_{\mathbf{w}_k} = \eta z_k \cdot \mathbf{H}_A^{-1}(\mathbf{w}_k) \left[\mathbf{x} - \nabla A(\mathbf{w}_k) \right] \ , \tag{D.59}$$

with some constant learning rate η. (D.58) is a learning rule for the biases of hidden causes, which we did not make use of in this article's simulations. It is considered here only for the sake of completeness. (D.59) is a non-local generalization of the local STDP rules (D.34) which could also capture conditional dependencies among inputs. $\mathbf{H}_A$ denotes the Hessian matrix of $A(\cdot)$,

$$h_{A,ij}(\mathbf{w}_k) = \frac{\partial^2 A(\mathbf{w}_k)}{\partial w_{ki} \partial w_{kj}} \ , \tag{D.60}$$

$\mathbf{H}_A$ can be also interpreted as the covariance matrix of the k-th mixture component, since the cumulant-generating function of an exponential family is $g(\mathbf{t}) = A(\mathbf{w}_k + \mathbf{t}) - A(\mathbf{w}_k)$, and

$$\mathrm{Cov}(x_i, x_j) = \frac{\partial^2 g(\mathbf{t})}{\partial t_i \partial t_j}|_{\mathbf{t}=\mathbf{0}} = \frac{\partial^2 A(\mathbf{w}_k)}{\partial w_{ki} \partial w_{kj}} \ . \tag{D.61}$$

We assume that the inverse of $\mathbf{H}_A(\mathbf{w}_k)$ exists, in which case it is guaranteed to be symmetric and positive definite. In the relevant case of conditional independence, this is trivially true for natural exponential families.

In the following, it will be shown that the expected update of the above learning rule converges to a local optimum of $l(\boldsymbol{\pi}, \mathbf{w})$ with respect to the optimized parameters. First note that the equilibrium points of the learning rule coincide with the necessary optimality conditions developed above:

$$\langle \Delta_{\pi_k} \rangle_{p^*(\mathbf{x})} = 0 \Leftrightarrow \pi_k = \log \langle p(k|\,\mathbf{x}, \mathbf{w}) \rangle_{p^*(\mathbf{x})} \quad , \tag{D.62}$$

$$\langle \Delta_{\mathbf{w}_k} \rangle_{p^*(\mathbf{x})} = \mathbf{0} \Leftrightarrow \nabla A(\mathbf{w}_k) = \frac{\langle p(k|\,\mathbf{x}, \mathbf{w})\, \mathbf{x} \rangle_{p^*(\mathbf{x})}}{\langle p(k|\,\mathbf{x}, \mathbf{w}) \rangle_{p^*(\mathbf{x})}} \quad . \tag{D.63}$$

From that it is concluded that the expected learning update is always non-zero for non-optimal settings, and zero at local optima. Next, it will be proven that the learning rule always drives the parameters in the right direction, by showing that the dot product $\nabla_{\pi,\mathbf{w}} l(\pi, \mathbf{w}) \cdot \langle \Delta_{\boldsymbol{\pi},\mathbf{w}} \rangle_{p^*(\mathbf{x})} \geq 0$. The derivative of $l(\pi, \mathbf{w})$ with respect to π_k is,

$$\frac{\partial l(\boldsymbol{\pi}, \mathbf{w})}{\partial \pi_k} = \langle\ \frac{\partial [\sum_{k'} e^{v_{k'}}]}{\partial \pi_k} \rangle_{p^*(\mathbf{x})} - e^{\pi_k} \tag{D.64}$$

$$= \langle\ \frac{e^{v_k}}{\sum_{k'} e^{v_{k'}}} \rangle_{p^*(\mathbf{x})} - e^{\pi_k} \tag{D.65}$$

$$= \langle p(k|\,\mathbf{x}) \rangle_{p^*(\mathbf{x})} - e^{\pi_k} \quad . \tag{D.66}$$

Similarly, differentiating with respect to $\mathbf{w}_k$ gives,

$$\frac{\partial l(\boldsymbol{\pi}, \mathbf{w})}{\partial\, \mathbf{w}_k} = \langle\ \frac{\partial [\sum_{k'} e^{v_{k'}}]}{\partial\, \mathbf{w}_k} \rangle_{p^*(\mathbf{x})} \tag{D.67}$$

$$= \langle\ \frac{e^{v_k}}{\sum_{k'} e^{v_{k'}}} (\mathbf{x} - \nabla A(\mathbf{w}_k)) \rangle_{p^*(\mathbf{x})} \tag{D.68}$$

$$= \langle p(k|\,\mathbf{x}, \mathbf{w}) (\mathbf{x} - \nabla A(\mathbf{w}_k))^T \rangle_{p^*(\mathbf{x})} \quad . \tag{D.69}$$

Then,

$$\begin{aligned} &\nabla_{\boldsymbol{\pi},\mathbf{w}} l(\boldsymbol{\pi}, \mathbf{w}) \cdot \langle \Delta_{\boldsymbol{\pi},\mathbf{w}} \rangle_{p^*(\mathbf{x})} = \\ &= \sum_k \frac{\partial l(\boldsymbol{\pi}, \mathbf{w})}{\partial \pi_k} \langle \Delta_{\pi_k} \rangle_{p^*(\mathbf{x})} + \sum_k \frac{\partial l(\boldsymbol{\pi}, \mathbf{w})}{\partial\, \mathbf{w}_k} \langle \Delta_{\mathbf{w}_k} \rangle_{p^*(\mathbf{x})} \qquad \text{(D.70)} \\ &= \sum_k [\langle p(k|\,\mathbf{x}, \mathbf{w}) \rangle_{p^*(\mathbf{x})} - e^{\pi_k}] \eta [e^{-\pi_k} \langle p(k|\,\mathbf{x}, \mathbf{w}) \rangle_{p^*(\mathbf{x})} - 1] + \\ &\quad \sum_k \langle p(k|\,\mathbf{x}, \mathbf{w}) (\mathbf{x} - \nabla A(\mathbf{w}_k))^T \rangle_{p^*(\mathbf{x})} \cdot \\ &\qquad \cdot \eta \cdot \mathbf{H}_A(\mathbf{w}_k)^{-1} \underbrace{[\langle p(k|\,\mathbf{x}, \mathbf{w}) (\mathbf{x} - \nabla A(\mathbf{w}_k)) \rangle_{p^*(\mathbf{x})}]}_{\mathbf{a}_k} \quad . \qquad \text{(D.71)} \end{aligned}$$

Since $\mathbf{H}_A(\mathbf{w}_k)^{-1}$ is symmetric and positive definite,

$$\begin{aligned}\nabla_{\boldsymbol{\pi},\mathbf{w}} l(\boldsymbol{\pi},\mathbf{w}) \cdot \langle \Delta_{\boldsymbol{\pi},\mathbf{w}} \rangle_{p^*(\mathbf{x})} &= \\ &= \sum_k \underbrace{\eta e^{-\pi_k} [\langle p(k|\,\mathbf{x},\mathbf{w})\rangle_{p^*(\mathbf{x})} - e^{\pi_k}]^2}_{\geq 0} + \sum_k \underbrace{\eta\, \mathbf{a}_k^T\, \mathbf{H}_A(\mathbf{w}_k)^{-1}\, \mathbf{a}_k}_{\geq 0} \qquad \text{(D.72)} \\ &\geq 0 \ , \qquad \text{(D.73)}\end{aligned}$$

completing the proof.

In summary, this proves that the expected learning update according to (D.58) and (D.59) will always drive the weights in a direction which increases the performance of the generative model, until a local optimum is reached.

D.13 What is the right learning rate?

The learning rate is a crucial parameter in any adaptive system. Unfortunately, in gradient based methods it is often difficult to predict a good learning rate for a particular task, since there is no simple mathematical relation between the learning problem and the learning rate. Here, we show that for the STDP learning rules (4.5) and (D.34) considered in this article, the learning rate has a tangible meaning: it is directly related to the number of inputs in the past which were involved in determining a particular weight. Hence, the learning rate reflects how long a synapse should remember previously collected statistics, as opposed to storing new information. Interpreted in an online setting, this reveals an intimate link between the learning rate and the time constants at which the input distribution is expected to change, which could in principle be tested experimentally.

Consider a generative mixture model,

$$\begin{aligned}p(\mathbf{x}\,|\,\mathbf{w}) &= \sum_{k=1}^{N} p(k) p(\mathbf{x}\,|k,\mathbf{w}_k) \qquad \text{(D.74)} \\ &= \sum_{k=1}^{N} \exp(\pi_k) p(\mathbf{x}\,|k,\mathbf{w}_k). \qquad \text{(D.75)}\end{aligned}$$

with $\boldsymbol{\pi} = (\pi_1, \ldots \pi_N)^T$, constrained by $\sum_k \exp(\pi_k) = 1$. This defines a general parametric mixture model with class probabilities $p(k) = \exp(\pi_k)$ and component parameters $\mathbf{w}_k$. We restrict ourselves to mixtures of exponential-family type distributions with linear sufficient statistics, such that the k-th component density can be written as

$$p(\mathbf{x}\,|k,\mathbf{w}_k) = h(\mathbf{x}) \exp\left(\sum_{j=1}^{M} w_{kj} x_j - A(\mathbf{w}_k)\right),$$

parametrized by $\mathbf{w}_k = (w_{k1}, \ldots, w_{kM})^T$.

For the following derivation, it is assumed that access to an ensemble of independent observations $(\mathbf{x}^{(t)}, \mathbf{z}^{(t)})$ is provided. Let $p^*(\mathbf{x}, \mathbf{z})$ denote the empirical distribution defined by these observations. Then the maximum likelihood parameters fulfill:

$$\pi_k = \log p^*(k) \tag{D.76}$$

$$\frac{\partial A(\mathbf{w}_k)}{\partial w_{kj}} = \langle x_j \rangle_{p^*(\mathbf{x}\,|k)} \tag{D.77}$$

Now, in contrast to standard batch learning, here we will derive an online learning rule which updates parameters incrementally while data is observed. The update at time step t should depend only on the current data $(\mathbf{x}^{(t)}, \mathbf{z}^{(t)})$ and the parameters from the last step $(\boldsymbol{\pi}^{(t-1)}, \mathbf{w}^{(t-1)})$.

First, let $\hat{\mathrm{p}}_k^{(t)}$ and $\hat{\mathbf{x}}_k^{(t)}$ be the empirical estimates of $p^*(k)$ and $\langle \mathbf{x} \rangle_{p^*(\mathbf{x}\,|k)}$, respectively, based on the observations up to time step t:

$$\hat{\mathrm{p}}_k^{(t)} = \frac{1}{t} \sum_{t'=1}^{t} z_k^{(t')},$$

$$\hat{\mathbf{x}}_k^{(t)} = \frac{\sum_{t'=1}^{t} z_k^{(t')}\, \mathbf{x}^{(t')}}{\sum_{t'=1}^{t} z_k^{(t')}}$$

where $z_k^{(t)} = 1$ if the hidden cause k was active at time t, and zero otherwise. Now suppose that the optimal parameters up to time step t have already been computed, that is for all k,

$$\pi_k^{(t)} = \log \hat{\mathrm{p}}_k^{(t)},$$

$$\nabla A(\mathbf{w}_k^{(t)}) = \hat{\mathbf{x}}_k^{(t)}\,. \tag{D.78}$$

For the next time step $t+1$ with $\eta^{(t+1)} = \frac{1}{t+1}$, one has,

$$\begin{aligned}
\pi_k^{(t+1)} &= \log \hat{\mathrm{p}}_k^{(t+1)} \\
&= \log \left((1 - \eta^{(t+1)})\, \hat{\mathrm{p}}_k^{(t)} + \eta^{(t+1)} z_k^{(t+1)} \right) \\
&= \log \left(\hat{\mathrm{p}}_k^{(t)} + \eta^{(t+1)} [z_k^{(t+1)} - \hat{\mathrm{p}}_k^{(t)}] \right) \\
&\approx \log \hat{\mathrm{p}}_k^{(t)} + \eta^{(t+1)} \frac{z_k^{(t+1)} - \hat{\mathrm{p}}_k^{(t)}}{\hat{\mathrm{p}}_k^{(t)}} \qquad \text{(D.79)} \\
&= \pi_k^{(t)} + \eta^{(t+1)} (e^{-\pi_k^{(t)}} z_k^{(t+1)} - 1) \\
&= \pi_k^{(t)} + \eta^{(t+1)} \Delta\pi_k^{(t+1)} \qquad \text{(D.80)}
\end{aligned}$$

with

$$\eta^{(t+1)}\Delta\pi_k^{(t+1)} = \begin{cases} \eta^{(t+1)}(e^{-\pi_k^{(t)}} - 1) & \text{if } z_k^{(t+1)} = 1, \\ -\eta^{(t+1)} & \text{otherwise.} \end{cases} \tag{D.81}$$

The approximation in (D.79) is valid for small $\eta^{(t+1)}$.

In order to obtain a recursive formula for $\mathbf{w}$, note that at time step $t+1$ the conditional expectation on the right-hand side of (D.78) changes only for the active hidden cause k. Therefore one can immediately write

$$\mathbf{w}_{k'}^{(t+1)} = \mathbf{w}_{k'}^{(t)}, \quad \forall k' \in \{1, \ldots, N\} \setminus \{k\}. \tag{D.82}$$

For k an update of the form $\mathbf{w}_k^{(t+1)} = \mathbf{w}_k^{(t)} + \eta_k \Delta\, \mathbf{w}_k^{(t+1)}$ is desired, and can be plugged in to give:

$$\begin{aligned} \nabla A(\mathbf{w}_k^{(t)} + \eta_k^{(t+1)} \Delta_{\mathbf{w}_k}^{(t+1)}) &= \hat{\mathbf{x}}_k^{(t+1)} \\ &= (1 - \eta_k^{(t+1)})\, \hat{\mathbf{x}}_k^{(t)} + \eta_k^{(t+1)}\, \mathbf{x}^{(t+1)} \\ &= \hat{\mathbf{x}}_k^{(t)} + \eta_k^{(t+1)} [\mathbf{x}^{(t+1)} - \hat{\mathbf{x}}_k^{(t)}] \\ &= \nabla A(\mathbf{w}_k^{(t)}) + \eta_k^{(t+1)} [\mathbf{x}^{(t+1)} - \nabla A(\mathbf{w}_k^{(t)})] \end{aligned}$$

Now, assuming small $\eta_k^{(t+1)}$ one may approximate the left-hand side by a first-order Taylor series

$$\nabla A(\mathbf{w}_k^{(t)} + \eta_k^{(t+1)} \Delta_{\mathbf{w}_k}^{(t+1)}) \approx \nabla A(\mathbf{w}_k^{(t)}) + \eta_k^{(t+1)} H_A(\mathbf{w}_k^{(t)}) \Delta_{\mathbf{w}_k}^{(t+1)},$$

and obtain

$$\eta_k^{(t+1)} \Delta_{\mathbf{w}_k}^{(t+1)} = \eta_k^{(t+1)} H_A^{-1}(\mathbf{w}_k^{(t)}) [\mathbf{x}^{(t+1)} - \nabla A(\mathbf{w}_k^{(t)})], \tag{D.83}$$

where $H_A(\cdot)$ is the Hessian of $A(\cdot)$ w.r.t. $\mathbf{w}_k$. The learning rate $\eta_k^{(t+1)}$ depends on the number of previous updates which have been performed with k:

$$\eta_k = \frac{1}{\sum_{t'=1}^{t+1} z_k^{(t)}} \tag{D.84}$$

(D.81), (D.82) and (D.83) define an approximate recursive estimation procedure for mixtures of exponential-family type distributions.

From this one can extract useful guidelines for choosing the global learning rate η in an unsupervised setting, where the hidden variables $\mathbf{z}$ need to inferred from the inputs $\mathbf{x}$. As was shown above, the learning rate η essentially determines the mixing between previous knowledge (the current parameter value) and new knowledge (the current input). Based on this, one can make a rough estimate of what an appropriate learning rate would be in a realistic input scenario: if the input distribution is stationary during approximately N_s updates for a typical synaptic weight, an appropriate choice would be $\eta \approx \frac{1}{N_s}$. Conversely, with a learning rate $\eta = 0.001$ used in the simulations of this article, a readout neuron's synaptic weights reflect approximately the last $N_s = 1000$ input patterns which triggered its firing.

Bibliography

Abbott, L. F. and Nelson, S. B. (2000). Synaptic plasticity: taming the beast. *Nature Neuroscience*, 3:1178–1183. 67

Abeles, M., Bergman, H., Gat, I., Meilijson, I., Seidemann, E., Tishby, N., and Vaadia, E. (1995). Cortical activity flips among quasi-stationary states. *Proc Natl Acad Sci U S A*, 92(19):8616–8620. 12, 39

Aire Technologies (2013). Sudoku solutions. http://www.sudoku-solutions.com. Accessed February 27, 2013. 121

Allen, C. and Stevens, C. F. (1994). An evaluation of causes for unreliability of synaptic transmission. *PNAS*, 91:10380–3. 1, 8

Angelaki, D. E., Gu, Y., and DeAngelis, G. C. (2009). Multisensory integration: psychophysics, neurophysiology and computation. *Current opinion in neurobiology*, 19(4):452–458. 82

Arieli, A., Sterkin, A., Grinvald, A., and Aertsen, A. (1996). Dynamics of ongoing activity: explanation of the large variability in evoked cortical responses. *Science*, 273:1868–1871. 19

Arnsten, A. F. T., Wang, M. J., and Paspalas, C. D. (2012). Neuromodulation of thought: flexibilities and vulnerabilities in prefrontal cortical network synapses. *Neuron*, 76:223–239. 33, 36

Austerweil, J. L., Abbott, J. T., and Griffiths, T. L. (2012). Human memory search as a random walk in a semantic network. In *Proceedings of NIPS 2012: Advances in Neural Information Processing Systems 25*, pages 3050–3058. MIT Press. 3

Bastos, A. M., Usrey, W. M., Adams, R. A., Mangun, G. R., Fries, P., and Friston, K. J. (2012). Canonical microcircuits for predictive coding. *Neuron*, 76(4):695–711. 10

Bejjanki, V., Beck, J., Lu, Z., and Pouget, A. (2011). Perceptual learning as improved probabilistic inference in early sensory areas. *Nature Neuroscience*, 5:642–648. 55

Berkes, P., Orban, G., Lengyel, M., and Fiser, J. (2011). Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment. *Science*, 331:83–87. 3, 9, 11, 12, 17, 38, 82, 124

Bi, G. and Poo, M. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *The Journal of Neuroscience*, 18(24):10464–10472. 59

Biere, A. (2009). *Handbook of satisfiability*, volume 185. IOS Press. 47, 48, 51

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer, New York. 2, 63, 65, 84, 91, 130, 131, 149

Blake, R. and Logothetis, N. K. (2002). Visual competition. *Nature Reviews Neuroscience*, 3(1):13–21. 25

Borovkov, A. A. (1998). *Ergodicity and stability of stochastic processes*. Hoboken, NJ: Wiley. 104

Borovkov, K., Decrouez, G., and Gilson, M. (2012). On stationary distributions of stochastic neural networks. 9, 32, 37, 98, 102, 104, 111

Borst, J. G. (2010). The low synaptic release probability in vivo. *Trends in Neurosciences*, 33(6):259–266. 1, 8, 100

Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239. 36

Branco, T. and Staras, K. (2009). The probability of neurotransmitter release: variability and feedback control at single synapes. *Nature Reviews Neuroscience*, 10(5):373–383. 2, 100

Brea, J., Senn, W., and Pfister, J.-P. (2012). Sequence learning with hidden units in spiking neural networks. In *Proc. of NIPS 2011*, volume 24, pages 1422–1430. MIT Press. 82

Brémaud, P. and Massoulié, L. (1996). Stability of nonlinear Hawkes processes. *The Annals of Probability*, 24(3):1563–1588. 9, 32, 111

Brooks, S., Gelman, A., Jones, G., and Meng, X.-L. (2010). *Handbook of Markov Chain Monte Carlo: Methods and Applications*. Chapman & Hall. 19, 20, 115, 116

Brooks, S. P. and Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455. 116

Brooks, S. P. and Roberts, G. O. (1998). Assessing convergence of Markov chain Monte Carlo algorithms. *Statistics and Computing*, 8(4):319–335. 115

Buesing, L., Bill, J., Nessler, B., and Maass, W. (2011). Neural dynamics as sampling: A model for stochastic computation in recurrent networks of spiking neurons. *PLoS Comput Biol*, 7(11):e1002211. 5, 9, 17, 18, 25, 34, 82, 90, 91, 92, 123, 124, 125, 126, 127

Buonomano, D. and Merzenich, M. (1998). Cortical plasticity: from synapses to maps. *Annual Reviews of Neuroscience*, 21(1):149–186. 70, 71, 73, 74

Buzsaki, G. (2009). *Rhythms of the brain*. Oxford: Oxford University Press. 23, 38

Buzsáki, G. (2010). Neural syntax: cell assemblies, synapsembles, and readers. *Neuron*, 68(3):362–385. 12, 26

Caporale, N. and Dan, Y. (2008). Spike timing-dependent plasticity: a Hebbian learning rule. *Annual Reviews of Neuroscience*, 31:25–46. 67

Cappé, O. and Moulines, E. (2009). Online EM algorithm for latent data models. *Journal of the Royal Statistical Society Series B (Statistical Methodology)*, 71(3):593–613. 65

Carandini, M. and Ferster, D. (2000). Membrane potential and firing rate in cat primary visual cortex. *The Journal of Neuroscience*, 20(1):470–484. 78

Chaisanguanthum, K. and Lisberger, S. (2011). A neurally efficient implementation of sensory population decoding. *The Journal of Neuroscience*, 31(13):4868–4877. 54, 77

Churchland, M. M. and Abbott, L. (2012). Two layers of neural variability. *Nature neuroscience*, 15(11):1472–1474. 38

Churchland, M. M., Yu, B. M., Cunningham, J. P., Sugrue, L. P., Cohen, M. R., Corrado, G. S., Newsome, W. T., Clark, A. M., Hosseini, P., Scott, B. B., Bradley, D. C., Smith, M. A., Kohn, A., Movshon, J. A., Armstrong, K. M., Moore, T., Chang, S. W., Snyder, L. H., Lisberger, S. G., Priebe, N. J., Finn, I. M., Ferster, D., Ryu, S. I., Santhana, G., Sahani, M., and Shenoy, K. V. (2010). Stimulus onset quenches neural variability: a widespread cortical phenomenon. *Nature Neuroscience*, 13(3):369–378. 2

Clarke, P. G. (2012). The limits of brain determinacy. *Proc Biol Sci*, 279(1734):1665–1674. 1, 8, 38

Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, STOC '71, pages 151–158, New York, NY, USA. ACM. 51

Cowles, M. K. and Carlin, B. P. (1996). Markov chain Monte Carlo convergence diagnostics: A comparative review. *Journal of the American Statistical Association*, 91(434):883–904. 19, 115

Craenen, B., Eiben, A., and van Hemert, J. I. (2003). Comparing evolutionary algorithms on binary constraint satisfaction problems. *Evolutionary Computation, IEEE Transactions on*, 7(5):424–444. 28

Dan, Y. and Poo, M.-M. (2006). Spike timing-dependent plasticity: From synapse to perception. *Physiological Reviews*, 86:1033–1048. 74

Davenport, A., Tsang, E., Wang, C. J., and Zhu, K. (1994). GENET: a connectionist architecture for solving constraint satisfaction problems by iterative improvement. In *Proceedings of the National Conference on Artificial Intelligence*, pages 325–325. John Wiley & Sons Ltd. 11

Daw, N. D., O'Doherty, J. P., Dayan, P., Seymour, B., and Dolan, R. J. (2006). Cortical substrates for exploratory decisions in humans. *Nature*, 441(7095):876–879. 3

Dayan, P., Hinton, G., Neal, R., and Zemel, R. (1995). The Helmholtz machine. *Neural Computation*, 7(5):889–904. 64, 77

Deneve, S. (2008a). Bayesian spiking neurons I: Inference. *Neural Computation*, 20(1):91–117. 82

Deneve, S. (2008b). Bayesian spiking neurons II: Learning. *Neural Computation*, 20(1):118–145. 77, 82, 84

Deneve, S., Latham, P., and Pouget, A. (1999). Reading population codes: a neural implementation of ideal observers. *Nature Neuroscience*, 2(8):740–745. 54, 76, 77

Denison, S., Bonawitz, E., Gopnik, A., and Griffiths, T. L. (2009). Preschoolers sample from probability distributions. In *Proceedings of the 32nd Annual Conference of the Cognitive Science Society*, volume 29, pages 1–10. 36

Desai, N., Rutherford, L., and Turrigiano, G. (1999). Plasticity in the intrinsic excitability of cortical pyramidal neurons. *Nature Neuroscience*, 2(6):515. 83, 86

Destexhe, A., Rudolph, M., Fellous, J.-M., and Sejnowski, T. J. (2001). Fluctuating synaptic conductances recreate *in vivo*-like activity in neocortical neurons. *Neuroscience*, 107(1):13–24. 114

Doeblin, W. (1937). Sur le propriétés asymtotiques de mouvement régis par certain types de chaînes simples. *Bull Math Soc Roumaine Sci*, 39:(1) 57–115, (2) 3–61. 102

Douglas, R. J. and Martin, K. A. (2004). Neuronal circuits of the neocortex. *Annual Reviews of Neuroscience*, 27:419–451. 10, 28, 31, 56

Down, D., Meyn, S., and Tweedie, R. (1995). Exponential and uniform ergodicity of Markov processes. *The Annals of Probability*, 23(4):1671–1691. 105, 106, 107, 110

Doya, K., Ishii, S., Pouget, A., and Rao, R. P. N. (2007). *Bayesian Brain: Probabilistic Approaches to Neural Coding*. MIT-Press. 9, 77

Dragoi, G. and Buzsaki, G. (2006). Temporal encoding of place sequences by hippocampal cell assemblies. *Neuron*, 50(1):145–157. 10, 23, 25

Durstewitz, D. (2006). A few important points about dopamine's role in neural network dynamics. *Pharmacopsychiatry*, 39:572–575. 33

Durstewitz, D. (2009). Implications of synaptic biophysics for recurrent network dynamics and active memory. *Neural Networks*, 22:1189–1200. 33

Durstewitz, D. and Deco, G. (2008). Computational significance of transient dynamics in cortical networks. *European Journal of Neuroscience*, 27:217–227. 34

Ecker, A. S., Berens, P., Keliris, G. A., Bethge, M., Logothetis, N. K., and Tolias, A. S. (2010). Decorrelated neuronal firing in cortical microcircuits. *Science*, 327:584–587. 76

El Adlouni, S., Favre, A.-C., and Bobée, B. (2006). Comparison of methodologies to assess the convergence of Markov chain Monte Carlo methods. *Computational Statistics & Data Analysis*, 50(10):2685–2701. 115

Engel, A. K., Fries, P., and Singer, W. (2001). Dynamic predictions: oscillations and synchrony in top-down processing. *Nat Rev Neurosci*, 2(10):704–16. 23

Ercsey-Ravasz, M. and Toroczkai, Z. (2012). The chaos within Sudoku. *Scientific Reports*, 2. 119

Faisal, A. A., Selen, L. P. J., and Wolpert, D. M. (2008). Noise in the nervous system. *Nature Reviews Neuroscience*, 9(4):292–303. 1, 8, 42

Fino, E. and Yuste, R. (2011). Dense inhibitory connectivity in neocortex. *Neuron*, 69(6):1188–1203. 74

Fiser, J., Berkes, P., Orban, G., and Lengyel, M. (2010). Statistically optimal perception and learning: from behavior to neural representation. *Trends in Cognitive Sciences*, 14(3):119–130. 3, 9, 15, 17, 37, 77, 82

Fox, M. D., Snyder, A. Z., Vincent, J. L., and Raichle, M. (2007). Intrinsic fluctuations within cortical systems account for intertrial variability in human behavior. *Neuron*, 56(1):171–184. 8

Freedman, D., Riesenhuber, M., Poggio, T., and Miller, E. (2006). Experience-dependent sharpening of visual shape selectivity in inferior temporal cortex. *Cerebral Cortex*, 16(11):1631–1644. 79

Friston, K. (2010). The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience*, 11(2):127–138. 9

Fu, Z., Mahajan, Y., and Malik, S. (2004). New features of the sat04 versions of zchaff. *SAT Competition*. 136

Garey, M. and Johnson, D. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Mathematical Sciences. Freeman, New York, NY. 35

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004). *Bayesian Data Analysis, Second Edition (Chapman & Hall/CRC Texts in Statistical Science)*. Chapman and Hall/CRC, 2 edition. 20, 116

Gelman, A. and Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472. 19, 115, 116

Gershman, S., Vul., E., and Tenenbaum, J. (2009). Perceptual multistability as Markov chain Monte Carlo inference. *Advances in Neural Information Processing Systems*, 22:611–619. 3

Gershman, S., Vul, E., and Tenenbaum, J. (2012). Multistability and perceptual inference. *Neural Computation*, 24(1):1–24. 25, 36

Gerstner, W. and Kistler, W. (2002). *Spiking Neuron Models*. Cambridge University Press, Cambridge. 1, 42, 59, 113, 139

Geweke, J. (1991). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. Staff Report 148, Federal Reserve Bank of Minneapolis. 115

Gilbert, C. D., Li, W., and Piech, V. (2009). Perceptual learning and adult cortical plasticity. *Journal of Physiology*, 387:2743–2751. 56

Gilson, M., Burkitt, A., and van Hemmen, J. (2010). STDP in recurrent neuronal networks. *Frontiers in Computational Neuroscience*, 4(23). 59

Gjoka, M., Kurant, M., Butts, C. T., and Markopoulou, A. (2010). Walking in facebook: A case study of unbiased sampling of osns. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE. 19, 115

Goel, A. and Lee, H. (2007). Persistence of experience-induced homeostatic synaptic plasticity through adulthood in superficial layers of mouse visual cortex. *The Journal of Neuroscience*, 27(25):6692–6700. 69

Graca, J., Ganchev, K., and Taskar, B. (2008). Expectation maximization and posterior constraints. In *Proc. of NIPS 2007*, volume 20. MIT Press. 83, 87

Graf, A., Kohn, A., Jazayeri, M., and Movshon, J. (2011). Decoding the activity of neuronal populations in macaque primary visual cortex. *Nature Neuroscience*, 14(2):239–245. 76, 147

Gray, R. M. (2009). *Probability, random processes, and ergodic properties*. New York: Springer. 112

Griffiths, T. L. and Tenenbaum, J. B. (2006). Optimal predictions in everyday cognition. *Psychological Science*, 17(9):767–773. 36, 54, 82

Gupta, A., der Meer, M. A. A., Touretzky, D. S., and Redish, A. D. (2012). Segmentation of spatial experience by hippocampal theta sequences. *Nature Neuroscience*, 15(7):1032–1039. 11, 25, 38

Gupta, A., Wang, Y., and Markram, H. (2000). Organizing principles for a diversity of gabaergic interneurons and synapses in the neocortex. *Science*, 287(5451):273–278. 15

Gutin, G. and Punnen, A. (2002). *The traveling salesman problem and its variations*, volume 12. Springer. 49

Habenschuss, S., Jonke, Z., and Maass, W. (2013). Stochastic computations in cortical microcircuit models. *PLoS Computational Biology*, 9(11):e1003311. 124

Haeusler, S. and Maass, W. (2007). A statistical analysis of information-processing properties of lamina-specific cortical microcircuit models. *Cerebral Cortex*, 17(1):149–162. 10, 14, 15, 16, 19, 21, 22, 114, 115, 117

Haeusler, S., Schuch, K., and Maass, W. (2009). Motif distribution, dynamical properties, and computational performance of two data-based cortical microcircuit templates. *Journal of Physiology, Paris*, 103(1-2):73–87. 10

Haider, B., Häusser, M., and Carandini, M. (2013). Inhibition dominates sensory responses in the awake cortex. *Nature*, 493(7430):97–100. 28

Harris, K., Csicsvari, J., Hirase, H., Dragoi, G., and Buzsáki, G. (2003). Organization of cell assemblies in the hippocampus. *Nature*, 424(6948):552–556. 38

Harvey, C. D., Coen, P., and Tank, D. W. (2012). Choice-specific sequencis in parietal cortex during a virtual-navigation decision task. *Nature*, 484:62–68. 16

Hinton, G. E., Sejnowski, T. J., and Ackley, D. H. (1984). Boltzmann machines: constraint satisfaction networks that learn. Technical Report CMS-CS-84-119, CMU Computer Science Department. 11

Hopfield, J. and Tank, D. (1986). Computing with neural circuits: a model. *Science*, 233(4764):625–633. 34

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *PNAS*, 79:2554–2558. 34

Hoyer, P. O. and Hyvärinen, A. (2003). Interpreting neural response variability as Monte Carlo sampling of the posterior. *Advances in Neural Information Processing Systems 15*, pages 293–300. 9, 25

Huang, X. and Lisberger, S. (2009). Noise correlations in cortical area MT and their potential impact on trial-by-trial variation in the direction and speed of smooth-pursuit eye movements. *Journal of Neurophysiology*, 101(6):3012–3030. 76

Hubel, D. and Wiesel, T. (1963). Shape and arrangement of columns in cat's striate cortex. *Journal of Physiology*, 165:559–568. 75

Huk, A. and Shadlen, M. (2005). Neural activity in macaque parietal cortex reflects temporal integration of visual motion signals during perceptual decision making. *The Journal of Neuroscience*, 25(45):10420–10436. 9

Jacob, V., Brasier, D., Erchova, I., Feldman, D., and Shulz, D. E. (2007). Spike timing-dependent synaptic depression in the in vivo barrel cortex of the rat. *The Journal of Neuroscience*, 27(6):1271–84. 59, 150

Jazayeri, M. and Movshon, J. (2006). Optimal representation of sensory information by neural populations. *Nature Neuroscience*, 9(5):690–696. 54, 55, 57, 62, 64, 65, 66, 74, 75, 76, 77, 141, 148

Jezek, K., Henriksen, E., Treves, A., Moser, E., and Moser, M. (2011). Theta-paced flickering between place-cell maps in the hippocampus. *Nature*, 478(7368):246–249. 11, 26, 27

Jiang, X., Wang, G., Lee, A. J., Stornetta, R. L., and Zhu, J. J. (2013). The organization of two new cortical interneuronal circuits. *Nature Neuroscience*, 16(2):210–218. 113

Jolivet, R., Rauch, A., Lüscher, H., and Gerstner, W. (2006). Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *Journal of Computational Neuroscience*, 21:35–49. 15, 57, 75, 84, 113, 123

Jones, B., Stekel, D., Rowe, J., and Fernando, C. (2007). Is there a liquid state machine in the bacterium escherichia coli? *Artificial Life*. ALIFE'07, IEEE Symposium. 39

Kandel, E. R., Schwartz, J. H., and Jessel, T. M. (1991). *Principles of Neural Science*. Prentice-Hall, third edition. 41

Karlsson, M., Tervo, D. G., and Karpova, A. Y. (2012). Network resets in medial prefrontal cortex mark the onset of behavioral uncertainty. *Science*, 338(6103):135–139. 8

Karp, R. (1972). *Reducibility among combinatorial problems*. Springer. 51

Kass, R. E., Carlin, B. P., Gelman, A., and Neal, R. M. (1998). Markov Chain Conte Carlo in practice: A roundtable discussion. *The American Statistician*, 52(2):93–100. 19, 116

Keck, C., Savin, C., and Lücke, J. (2012). Feedforward inhibition and synaptic scaling - two sides of the same coin? *PLoS Computational Biology*, 8(3):e1002432. 77, 82, 84, 93

Kelemen, E. and Fenton, A. A. (2010). Dynamic grouping of hippocampal neural activity during cognitive control of two spatial frames. *BLoS Biology*, 8(6):e1000403. 8, 23

Kenet, T., Bibitchkov, D., Tsodyks, M., Grinvald, A., and Arieli, A. (2003). Spontaneously emerging cortical representations of visual attributes. *Nature*, 425(6961):954–956. 8

Kim, C. Y. and Blake, R. (2005). Psychophysical magic: rendering the visible 'invisible'. *Trends in Cognitive Sciences*, 9(8):381–388. 8

Kirkpatrick, S., C.D., G., and Vecchi, M. (1983). Optimization by simmulated annealing. *science*, 220(4598):671–680. 47

Klampfl, S., David, S. V., Yin, P., Shamma, S. A., and Maass, W. (2012). A quantitative analysis of information about past and present stimuli encoded by spikes of A1 neurons. *Journal of Neurophysiology*, 108:1366–1380. 33

Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques (Adaptive Computation and Machine Learning)*. MIT Press. 15, 36

Körding, K. P. and Wolpert, D. M. (2004). Bayesian integration in sensorimotor learning. *Nature*, 427(6971):244–247. 82

Kumar, V. (1992). Algorithms for constraint-satisfaction problems: A survey. *AI magazine*, 13(1):32. 28

Larkum, M. (2012). A cellular mechanism for cortical associations: an organizing principle for the cerebral cortex. *Trends in Neurosciences*, 951:1–11. 113

Larkum, M. (2013). The yin and yang of cortical layer 1. *Nature Neuroscience*, 16(2):114–115. 113

Latash, M. L., Scholz, J. F., Danion, F., and Schöner, G. (2001). Structure of motor variability in marginally redundant multifinger force production tasks. *Experimental Brain Research*, 141(2):153–165. 2

Law, C. and Gold, J. (2009). Reinforcement learning can account for associative and perceptual learning on a visual-decision task. *Nature Neuroscience*, 12:655–663. 78

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, volume 86, pages 2278–2324. 86

Leopold, D. A. and Logothetis, N. K. (1996). Activity changes in early visual cortex reflect monkeys' percepts during binocular rivalry. *Nature*, 379(6565):549–553. 8

Leopold, D. A. and Logothetis, N. K. (1999). Multistable phenomena: changing views in perception. *Trends in Cognitive Sciences*, 3(7):254–264. 8

Lewis, C. M., Baldassarre, A., Committeri, G., Romani, G. L., and Corbetta, M. (2009). Learning sculpts the spontaneous activity of the resting human brain. *PNAS*, 106(41):17558–17563. 8, 38

Liao, D., Jones, A., and Malinow, R. (1992). Direct measurement of quantal changes underlying long-term potentiation in CA1 hippocampus. *Neuron*, 9(6):1089–1097. 59

Lieder, F., Griffiths, T., and Goodman, N. (2013). Burn-in, bias, and the rationality of anchoring. In *Proc. of NIPS 2012*, volume 25, pages 2690–2698. MIT Press. 33

Lisman, J. E., Raghavachari, S., and Tsien, R. W. (2007). The sequence of events that underlie quantal transmission at central glutamatergic synapses. *Nature Reviews Neuroscience*, 8(8):597–609. 100

Litwin-Kumar, A. and Doiron, B. (2012). Slow dynamics and high variability in balanced cortical networks with clustered connections. *Nature neuroscience*, 15(11):1498–1505. 38

Louie, K., Grattan, L., and Glimcher, P. (2011). Reward value-based gain control: Divisive normalization in parietal cortex. *The Journal of Neuroscience*, 31(29):10627–10639. 58

Luczak, A., Barthó, P., and Harris, K. D. (2009). Spontaneous events outline the realm of possible sensory responses in neocortical populations. *Neuron*, 62:413–425. 8

Luczak, A., Barthó, P., Marguet, S. L., Buzsáki, G., and Harris, K. D. (2007). Sequential structure of neocortical spontaneous activity in vivo. *PNAS*, 104(1):347–352. 12, 16, 23, 39

Luczak, A. and MacLean, J. N. (2012). Default activity patterns at the neocortical microcircuit level. *Frontiers in Integrative Neuroscience*, 6(30):doi: 10.3389/fnint.2012.00030. 12

Lynce, I. and Marques-Silva, J. (2006). Efficient haplotype inference with boolean satisfiability. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, page 104. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. 51

Maass, W. and Markram, H. (2002). Synapses as dynamic memory buffers. *Neural Networks*, 15:155–161. 114

Maass, W., Natschlaeger, T., and Markram, H. (2002). Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Computation*, 14(11):2531–2560. 33, 34

Maass, W. and Sontag, E. (1999). Analog neural nets with Gaussian or other common noise distributions cannot recognize arbitrary regular languages. *Neural Computation*, 11:771–782. 103

Markram, H., Wang, Y., and Tsodyks, M. (1998). Differential signaling via the same axon of neocortical pyramidal neurons. *PNAS*, 95:5323–5328. 15, 114

Massoulié, L. (1998). Stability results for a general class of interacting point processes dynamics, and applications. *Stochastic processes and their applications*, 75(1):1–30. 111

Mazor, O. and Laurent, G. (2005). Transient dynamics versus fixed points in odor representations by locust antennal lobe projection neurons. *Neuron*, 48:661–673. 16

Meyn, S. P. and Tweedie, R. L. (1993). Stability of markovian processes ii: Continuous-time processes and sampled chains. *Advances in Applied Probability*, pages 487–517. 107

Michalewicz, Z. and Fogel, D. (2000). *Escaping Local Optima*. Springer. 47

Miller, L. and Recanzone, G. (2009). Populations of auditory cortical neurons can accurately encode acoustic space across stimulus intensity. *Proceedings of the National Academy of Sciences*, 106(14):5931–5935. 54

Montgomery, J., Pavlidis, P., and Madison, D. (2001). Pair recordings reveal all-silent synaptic connections and the postsynaptic expression of long-term potentiation. *Neuron*, 29(3):691–701. 59

Morrison, J. and Hof, P. (2007). Life and death of neurons in the aging cerebral cortex. *International Review of Neurobiology*, 81:41–57. 69

Mountcastle, V. B. (1998). *Perceptual Neuroscience: The Cerebral Cortex*. Harvard University Press, Cambridge. 10

Nessler, B., Pfeiffer, M., and Maass, W. (2010). STDP enables spiking neurons to detect hidden causes of their inputs. *Proceedings of NIPS 2009: Advances in Neural Information Processing Systems*, 22:1357–1365. 5, 56, 64, 77, 82, 83, 84, 85, 89, 90, 93, 141

Nikolic, D., Haeusler, S., Singer, W., and Maass, W. (2009). Distributed fading memory for stimulus properties in the primary visual cortex. *PLoS Biology*, 7(12):1–19. 33

Ohshiro, T., Angelaki, D., and DeAngelis, G. (2011). A normalization model of multisensory integration. *Nature Neuroscience*, 14:775–782. 58

Okun, M., Yger, P., Marguet, S. L., Gerard-Mercier, F., Benucci, A., Katzner, S., and Harris, K. D. (2012). Population rate dynamics and multineuron firing patterns in sensory cortex. *Journal of Neuroscience*, 32(48):17108–17119. 38

Orban, G., Fiser, J., Aslin, R., and Lengyel, M. (2008). Bayesian learning of visual chunks by human observers. *Proceedings of the National Academy of Sciences*, 105(7):2745–2750. 82

Pecevski, D. (2013). Nevesim – an event based simulator for networks of spiking neurons. http://sim.igi.tugraz.at/. 121

Pecevski, D., Buesing, L., and Maass, W. (2011). Probabilistic inference in general graphical models through sampling in stochastic networks of spiking neurons. *PLoS Computational Biology*, 7(12):e1002294. 5, 9, 17, 34, 82, 90, 124, 125

Pecevski, D., Natschlaeger, T., and Schuch, K. (2009). PCSIM: a parallel simulation environment for neural circuits fully integrated with python. *Frontiers in Neuroinformatics*, 3. doi:10.3389/neuro.11.011.2009. 113

Petersen, R., Panzeri, S., and Diamond, M. (2002). Population coding in somatosensory cortex. *Current Opinion in Neurobiology*, 12(4):441–447. 54

Pipa, G., Staedtler, E. S., Rodriguez, E. F., Waltz, J. A., Muckli, L., Singer, W., Goebel, R., and Munk, M. H. J. (2009). Performance- and stimulus-dependent oscillations in monkey prefrontal cortex during short-term memory. *Frontiers in Integrative Neuroscience*, 3(25). 23

Plischke, M. and Bergersen, B. (2006). *Equilibrium statistical physics*. World Scientific. 125

Potjans, T. C. and Diesmann, M. (2012). The cell-type specific cortical microcircuit: relating structure and activity in a full-scale spiking network model. *Cerebral Cortex*, page doi:10.1093/cerror/bhs358. 10

Pouget, A., Dayan, P., and Zemel, R. (2000). Information processing with population codes. *Nature Reviews Neuroscience*, 1(2):125–132. 54

Priebe, N., Mechler, F., Carandini, M., and Ferster, D. (2004). The contribution of spike threshold to the dichotomy of cortical simple and complex cells. *Nature Neuroscience*, 7:1113–1122. 75

Qin, Y., Xiang, J., Wang, R., Zhou, H., Li, K., and Zhong, N. (2012). Neural bases for basic processes in heuristic problem solving: take solving sudoku puzzles as an example. *PsyCh Journal*, 1(2):101–117. 36

Rabinovich, M., Huerta, R., and Laurent, G. (2008). Transient dynamics for neural processing. *Science*, 321:45–50. 34

Raftery, A. E., Lewis, S., et al. (1992). How many iterations in the Gibbs sampler. *Bayesian Statistics*, 4(2):763–773. 115

Raichle, M. E. (2010). Two views of brain function. *Trends in Cognitive Sciences*, 14(4):180–190. 8, 35

Rao, R. and Ballard, D. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1):79–87. 64, 77

Rasch, M. J., Schuch, K., Logothetis, N. K., and Maass, W. (2011). Statistical comparision of spike responses to natural stimuli in monkey area V1 with simulated responses of a detailed laminar network model for a patch of V1. *J Neurophysiol*, 105:757–778. 10

Rezende, D. J., Wierstra, D., and Gerstner, W. (2012). Variational learning for recurrent spiking networks. In *Proc. of NIPS 2011*, volume 24, pages 136–144. MIT Press. 82

Rolls, E. T. and Deco, G. (2010). *The Noisy Brain: Stochastic Dynamics as a Principle of Brain Function*. Oxford Univ. Press. 34

Sato, M. (1999). Fast learning of on-line EM algorithm. Technical report, ATR Human Information Processing Research Laboratories, Kyoto, Japan. 65, 84, 143

Schemmel, J., Brüderle, D., Grübl, A., Hock, M., Meier, K., and Millner, S. (2010). A wafer-scale neuromorphic hardware system for large-scale neural modeling. *Proc. of ISCAS'10*, pages 1947–1950. 82

Schneidman, E., Berry, M., Segev, R., and Bialek, W. (2006). Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007–1012. 124

Seitz, A., Kim, D., and Watanabe, T. (2009). Rewards evoke learning of unconsciously processed visual stimuli in adult humans. *Neuron*, 61(5):700–707. 72

Seitz, A. and Watanabe, T. (2005). A unified model for perceptual learning. *Trends in Cognitive Sciences*, 9(7):329–334. 70, 71

Selen, L. P., Beek, P. J., and van Dieën, J. H. (2006). Impedance is modulated to meet accuracy demands during goal-directed arm movements. *Experimental Brain Research*, 172(1):129–138. 2

Shadlen, M. N. and Newsome, W. T. (2001). Neural basis of a perceptual decision in the parietal cortex (area lip) of the rhesus monkey. *Journal of Neurophysiology*, 86(4):1916–1936. 17

Shinomoto, S., Kim, H., Shimokawa, T., Matsuno, N., Funahashi, S., Shima, K., Fujita, I., Tamura, H., Doi, T., Kawano, K., Inaba, N., Fukushima, K., Kurkin, S., Kurata, K., Taira, M., Tsutsui, K., Komatsu, H., Ogawa, T., Koida, K., Tanji, J., and Toyama, K. (2009). Relating neuronal firing patterns to functional differentiation of cerebral cortex. *PLoS Computational Biology*, 5(7):e1000433. 66, 149

Siapas, A. G., Lubenov, E. V., and Wilson, M. A. (2005). Prefrontal phase locking to hippocampal theta oscillations. *Neuron*, 46(1):141–151. 28

Siegel, M., Warden, M. R., and Miller, E. K. (2009). Phase-dependent neuronal coding of objects in short-term memory. *PNAS*, 106(50):21341–21346. 23, 38

Simoncelli, E. and Heeger, D. (1998). A model of neuronal responses in visual area MT. *Vision Research*, 38(5):743–761. 58, 84

Sjöström, P., Turrigiano, G., and Nelson, S. (2001). Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. *Neuron*, 32(6):1149–1164. 59, 60, 67

Smith, M. and Kohn, A. (2008). Spatial and temporal scales of neuronal correlation in primary visual cortex. *The Journal of Neuroscience*, 28(48):12591–12603. 76

Sporns, O. (2011). The human connectome: a complex network. *Annals of the New York Academy of Sciences*, 1224(1):109–125. 35

Steimer, A., Maass, W., and Douglas, R. (2009). Belief propagation in networks of spiking neurons. *Neural Computation*, 21:2502–2523. 82

Sterzer, P., Kleinschmidt, A., and Rees, G. (2009). The neural bases of multistable perception. *Trends in cognitive sciences*, 13(7):310–318. 25

Tenenbaum, J., Kemp, C., Griffiths, T., and Goodman, N. (2011). How to grow a mind: statistics, structure, and abstraction. *Science*, 331(6022):1279–1285. 36, 82

Thomson, A. M., West, D. C., Wang, Y., and Bannister, A. P. (2002). Synaptic connections and small circuits involving excitatory and inhibitory neurons in layers 2 - 5 of adult rat and cat neocortex: triple intracellular recordings and biocytin labelling in vitro. *Cerebral Cortex*, 12(9):936–953. 15

Trachtenberg, J., Chen, B., Knott, G., Feng, G., Sanes, J., Welker, E., and Svoboda, K. (2002). Long-term in vivo imaging of experience-dependent synaptic plasticity in adult cortex. *Nature*, 420(6917):788–794. 69

Truccolo, W., Eden, U., Fellows, M., Donoghue, J., and Brown, E. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089. 123

Tsodyks, M., Kenet, T., Grinvald, A., and Arieli, A. (1999). Linking spontaneous activity of single cortical neurons and the underlying functional architecture. *Science*, 286(5446):1943–1946. 8

Turesson, H. K., Logothetis, N. K., and Hoffman, K. L. (2012). Category-selective phase coding in the superior temporal sulcus. *PNAS*, 109(47):19438–19443. 23, 38

Turrigiano, G. (2010). The self-tuning neuron: synaptic scaling of excitatory synapses. *Cell*, 135(3):422–435. 67

Ullman, T. D., Goodman, N. D., and Tenenbaum, J. B. (2010). Theory Acquisition as Stochastic Search. In *Proceedings of the Thirty-Second Annual Meeting of the Cognitive Science Society*. 3

Valero-Cuevas, F. J., Venkadesan, M., and Todorov, E. (2009). Structured variability of muscle activations supports the minimal intervention principle of motor control. *Journal of neurophysiology*, 102(1):59–68. 2

Van Hooser, S., Heimel, J., Chung, S., Nelson, S., and Toth, L. (2005). Orientation selectivity without orientation maps in visual cortex of a highly visual mammal. *The Journal of Neuroscience*, 25(1):19–28. 75

van Rossum, G. and Drake, F. L. (2001). Python reference manual. Pythonlabs, Virginia, USA, 2001. Available at http://www.python.org. 113

Vilares, I. and Kording, K. (2011). Bayesian models: the structure of the world, uncertainty, behavior, and the brain. *Annals of the New York Academy of Sciences*, 1224(1):22–39. 9

Vul, E., Goodman, N. D., Griffiths, T. L., and Tenenbaum, J. B. (2009). One and done? optimal decisions from very few samples. In *Proceedings of the 31st Annual Conference of the Cognitive Science Society*, volume 1, pages 66–72. 33

Vul, E. and Pashler, H. (2008). Measuring the crowd within: probabilistic representations within individuals. *Psychological Science*, 19(7):645–647. 36

Wainwright, M. J. and Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305. 17, 19

Wang, X. J. (2010). Neurophysiological and computational principles of cortical rhythms in cognition. *Physiological Reviews*, 90(3):1195–1268. 23

Watt, A. and Desai, N. (2010). Homeostatic plasticity and STDP: keeping a neuron's cool in a fluctuating world. *Frontiers in Synaptic Neuroscience*, 2. 83, 86

Xu, S., Jiang, W., Poo, M., and Dan, Y. (2012). Activity recall in a visual cortical ensemble. *Nature Neuroscience*, 15(3):449–456. 38

Yamanaka, K., Agu, M., and Miyajima, T. (1997). A continuous-time asynchronous boltzmann machine. *Neural networks*, 10(6):1103–1107. 131

Yarom, Y. and Hounsgaard, J. (2011). Voltage fluctuations in neurons: signal or noise? *Physiol Rev*, 91(3):917–929. 1, 8

Zhang, Q. F., Wen, Y., Zhang, D., She, L., Wu, J. Y., Dan, Y., and Poo, M. M. (2012). Priming with real motion biases visual cortical response to bistable apparent motion. *PNAS*, 109(50):20691–20696. 38, 39

Zhang, Y., Meyers, E. M., Bichot, N. P., Serre, T., Poggio, T. A., and Desimone, R. (2011). Object decoding with attention in inferior temporal cortex. *Proceedings of the National Academy of Sciences*, 108(21):8850–8855. 17

Zoccolan, D., Cox, D., and DiCarlo, J. (2005). Multiple object response normalization in monkey inferotemporal cortex. *The Journal of Neuroscience*, 25(36):8150–8164. 58