THE UNIVERSITY OF CHICAGO

UNDERSTANDING NEOCORTICAL DYNAMICS AND COMPUTATION THROUGH SPIKING NEURAL NETWORK MODELING

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE BIOLOGICAL SCIENCES
AND THE PRITZKER SCHOOL OF MEDICINE
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

COMMITTEE ON COMPUTATIONAL NEUROSCIENCE

BY
YUQING ZHU

CHICAGO, ILLINOIS
AUGUST 2023

COPYRIGHT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my family. Thank you to my mother, Dr. Wenmei Li. You are the first reason I wanted to pursue a science doctorate. Thank you to my father. Thank you both for teaching me math, encouraging me to take community college coding classes, and showing me that learning is a lifelong act. Thank you to my sibling. I am so grateful for the time you shared with me in Chicago, and I look forward to being in the same city again in the future.

I would like to thank my friends. Thank you to my friends from home and from college for raising me. Thank you to my friends I met in Chicago for helping me to grow. Thank you to my Pilsen housemates Maayan and Kaavya, plus Divya and Kaden. You all saved me throughout the pandemic and the middle years of my degree.

I would like to thank the city of Chicago, which has been endlessly generous to me. I would like to thank Lake Michigan.

I would like to thank Dr. Stewart Hendry and Dr. Kathryn Schaffer, who both took a chance on me and showed me the way to the next chapter of my career. I would like to thank my first ever research mentor, Dr. Marina Bedny. I would like to thank my thesis committee members: Drs. Stephanie Palmer, John Maunsell, and Dave Freedman. Thank you for guiding me throughout the development of this dissertation.

I would like to thank the members of the MacLean lab. You have always challenged me to be a better scientist, better friend, better peer, mentor, mentee. I would especially like to thank Chad

for scoping out hot cheetos for me and for being my partner in the struggles of training SNNs. Without you this dissertation would not be possible. I would like to thank Mufeng, Tarek, Josh, Kyle, and Franz for that last reason as well. Thank you to Liz, Gabriella, and Hal for all of your insightful comments, which were crucial to guiding this work into its form today. The final member of the MacLean lab I would like to thank is, of course, Dr. Jason MacLean. Thank you for striking out into the unknown with me and advising me on so much more than science. I look forward to future opportunities to share good news.

I would like to thank two people who have made the final year of my degree–which could have been a very hard time–a time I will treasure instead. Thank you to Aim for joining me at UChicago and for keeping me company as we wrote our theses side-by-side. Thank you for always being down to reminisce, to catch a film, and to find me in Mansueto, even when it was truly too sunny. Thank you to Benton for being so generous with your time, humor, and support. Thank you for appreciating poetry with me, showing me how to take some things less seriously and to value other things more, and for always being down to go for a walk together.

I would like to thank Bryant Smith, who was there for me through the worst times, and who showed me the good in striving to be good. Thank you for making our little family with Hailey possible for a little while.

I would like to thank Katherine Xiang, who has been beside me through it all. You inspire me to be more adventurous and true every day. Thank you for rekindling my sense of wonder each time it needed a little tending to. You are my morning light.

ABSTRACT

Through the use of biofidelic spiking neural network models (SNNs), this work offers mechanistic insights into the relationship between neocortical structure, dynamics, and computation. To set the stage, the topics of cortical computation, ANNs as applied to neuroscience, SNNs as applied to studying structure-function relationships in neocortex, and the challenges of model interpretability are introduced. In the first chapter, in the tradition of using SNNs to address structure-function questions, we ask why stable dynamics are possible in neocortex. Biofidelic SNNs were created through a grid search for architectures that yielded realistic spiking activity that was low-rate, asynchronous, and near-critical. The maintenance of this activity is linked to patterns of higher order coordination of synaptic activity, and this coordination takes the form of transitions in time between specific three-unit motifs. These motifs summarize the way spikes traverse the underlying synaptic topology. The second chapter turns its focus to computation, which occurs in neocortex on a substrate of stable activity that we studied in the first. Specifically, this chapter covers why computation becomes possible through specific synaptic changes and resulting dynamic changes that occur through learning. After training SNN models to perform an ethologically relevant task, models come to selectively adjust firing rates in response to the stimulus input. Excitatory and inhibitory connectivity between input and recurrent layers changed in accordance with this rate modulation. In particular, recurrent inhibitory units which were tuned to one input over the other strengthened their connections to recurrent units of the opposite tuning. We conclude by discussing the potential of task-trained SNNs for hypothesis generation and testing in future research on neocortical computation. Additional neocortical features that may be important for computation are surveyed, and questions of model interpretation are revisited in the context of these results.

INTRODUCTION

On good explanations

Take the object of a single copper atom on a bronze statue of a man reading. This statue sits on a low bench outside the entrance to the Sunnyvale, California Public Library. The book that he is holding, also cast in bronze, is open to a page of *Pedro Páramo*, a 1955 novel by Mexican writer Juan Rulfo. Acclaimed authors have considered this to be the greatest work of literature ever written in any language (Lewis 2008, Saadi 2009).

Now take the question: **why is the copper atom there?** One could chart, from the beginning of the universe, the formation of the element copper, the exhaustive trajectory of this particular atom over millennia, and the discovery of bronze through alloying copper with tin and other trace elements.

One could also answer that the sculptor was moved by this particular novel and wanted to create a statue dedicated to it in a place for literature, that this work of literature was so significant to the sculptor because it captured the lasting impact of the Mexican Revolution on ordinary people, as well as the power of those forces which transcend history, such as cruelty and hope, to move our lives.

Both of these answers have their place, but only one of them tells us ***why*** the copper atom is there (Deutsch 1998).

Here's another question: why is this atom at the center of this asterisk * ? We could again trace atomic trajectories over all of time, or we could answer that Yuqing seeks to defend her PhD and is trying to do so in a way that's mildly interesting to everyone.

We might sense that one explanation is better than the other.

Of course, all dissertations fall somewhere along this continuum of levels of explanation, from particle trajectories to the emergent human forces of war and notions of the poetic. This dissertation is a scientific one; the search for good scientific explanations should beg the same questions: Are we truly asking *why*? Can we expect based on our approach to be satisfied with the answers?

As scientists, we seek good explanations for natural phenomena. The curiosity I seek to explain is the following: "**why is it that, through learning, neocortical networks become capable of computations?**"

On this dissertation

Through the work that constitutes this dissertation, I seek to contribute a part of an answer to the question above. I intentionally phrased the question to use the word "why" rather than "how" (as in: "how does neocortex achieve computation?") Although "how" and "why" can be equivalent depending on phrasing, "how" questions are more prone to leading to *descriptions* instead of good explanations.

I approach the question through two sub-questions, each occupying a chapter of this dissertation: first, I answer why the brain exhibits some of the structure that it does. Second, I answer why the structural and dynamical changes that occur during learning enable a particular task computation. I approach both questions using the tool of biofidelic spiking neural network (SNN) models.

I would like to pause and note that SNNs are a step towards greater biological fidelity from existing machine learning frameworks. For this reason, the thrust of the work in this dissertation may read as promoting reductionism. The particular illustrations at the beginning–such as regarding the atom in the asterisk–may appear to have an anti-reductionist agenda. Yet my work is neither pro- nor anti-reductionist. It is agnostic about levels and happy to encounter a good explanation wherever it may arise.

On modeling

Models are one way of approaching "why" questions. They are necessarily limited representations of real world systems, and, while this can sometimes be construed as a negative, their focus is often beneficial for inquiry. By concentrating on essential features and the relationships between them, models can capture fundamental principles and mechanisms underlying the system of interest. This focus also enables the formulation of precise and tractable mathematical descriptions under some circumstances, further facilitating analysis and understanding. Through model simulation and manipulation, scientists can explore hypothetical scenarios that may be difficult to instantiate or observe in real systems, and test the consequences of various assumptions. When a model successfully predicts new experimental findings or

replicates existing ones, it provides support for the validity of the underlying assumptions and offers avenues for elucidating the mechanisms at work (Winsberg 2010).

Models also provide a framework for integrating knowledge from different fields and disciplines. By combining empirical data, theoretical principles, and mathematical formulations, models can bridge gaps between disparate areas of research and facilitate interdisciplinary collaboration. They allow researchers to organize and synthesize existing knowledge, leading to new insights and discoveries. The models used in this work integrate experimental results from neurobiology with tools from machine learning and insights on principles from theoretical neuroscience.

Finally, modeling is (or should be) an iterative process by definition. Models are revised and refined based on new data, insights, and advancements in theory and methodology (Trensch et al. 2018). This allows for the progressive development of more accurate, comprehensive, and useful representations of the system under study. We hope to incorporate even more biofidelity into our future modeling work, and to revise results as needed to include new experimental evidence.

Scientists model the brain at multiple levels, each providing different insights into its function. The brain is generally not modeled at the level of particles, nor at the level of wars. Instead, we tend to focus on the limited range of the molecular to the behavioral. At the molecular and cellular levels, models simulate the behavior of ion channels, synaptic transmission, and biochemical processes within cells. Network level models focus on the interactions between groups of neurons, and they aim to capture how patterns of connectivity and activity give rise to emergent properties such as neural oscillations, synchronization, firing rate modulation, and

other network dynamics. Models at the systems level explore how different brain regions are interconnected and work together to perform higher level processes such as memory and attention. Brain modeling at the cognitive and behavioral levels focus on understanding how brain activity translates into observable behaviors such as for decision-making and problem-solving. These models aim to link neural activity to motor output, sensory perception, and other behavioral responses.

It's important to note that these levels are not mutually exclusive; there is in fact continuous interplay between them. For example, the network and systems levels are often studied together. It is also possible–and common–to model the effects of lower level features, such as a particular gene, on high level features like behaviors (Flint et al. 2020). In addition, even within a given level as categorized above, there are subdivisions. For example, most of this work is situated at the network level, as I explore how synaptic and functional connections in a local recurrent network change to support different dynamic regimes, but so is work that uses convolutional neural networks (CNNs) to model vision (e.g. Lindsay 2021), and so is work that characterizes networks as their low-dimensional neural state space geometries (e.g. Chung & Abbott 2021); these are all very different levels of inquiry that co-exist under the umbrella of “network”.

There is real explanatory power to be found at all these levels. As outlined, each level has certain strengths and types of questions it is suited to answering. We, among many other researchers, have previously chosen to use neuronal network level modeling to understand the mechanisms underlying neocortical features such as signal propagation, which is a prerequisite to computation.

On signal propagation

In the first chapter of this dissertation, we ask "why are neural signals able to propagate in neocortex?" Signal propagation is the generation and transmission of electrical and chemical signals between neurons, which allows information to be communicated and transformed. At its core, signal propagation is about the nonlinear transformation of structure into function, which in turn corresponds to computation.

Given the fact that structurally, the vast majority of excitatory synapses are weak and connections are sparse and recurrent, successful and meaningful activity propagation is highly non-trivial (Song et al. 2005, Perin et al. 2011, Seeman et al. 2018, Vogels & Abbott 2005), and is an area of active research. Theoretical and experimental studies have characterized several architectural features that have the capacity to promote and shape stable spiking activity, such as a heavy-tailed synaptic weight distribution, excitatory clustering, and the ratio between incoming and outgoing connections (Song et al. 2005; Lefort et al. 2009; Teramae et al. 2012; Litwin-Kumar & Doiron 2012; van Vreeswijk & Sompolinsky 1996).

Additionally, dynamical properties of ongoing activity, such as a balance between excitation and inhibition (Litwin-Kumar & Doiron 2012), are shaped by connectivity and in turn impact the continuation of spiking activity. We know that overall, signals in the form of spiking activity in neocortex of healthy, awake mammals is asynchronous, low rate, and near-critical (Beggs et al. 2003, Zylberberg et al. 2017). Theories that explain stable activity propagation in neocortex first aim to capture these features of the background activity. Several studies have focused explicitly on self-sustained activity in the asynchronous state and the propagation of inputs in this state.

These studies found a complex relationship between synaptic strength and firing rate in the maintenance of an asynchronous spiking regime (Kriener et al. 2014, Brunel 2000, Zerlaut et al. 2019). Other studies have demonstrated that sustained activity co-occurs within a specific range of firing rates, supported by a balance between excitatory and inhibitory conductance (Vogels & Abbott 2005, Brunel 2000). Firing rates that are too low or too high contribute to instability within the network.

Neocortex is also characterized as having critical or near-critical dynamics, which is supported by a balance between excitation and inhibition. This balance is thought to be essential for signal maintenance as well as enabling efficient computation (Haider et al. 2006, van Vreeswijk & Sompolinsky 1996, Shew et al. 2011). For example, networks tuned near the critical point display maximum information transmission (Beggs & Plenz 2003), information storage (Haldeman & Beggs 2005), and computational power (Bertschinger & Natschläger 2004). Imbalances in excitation and inhibition have been implicated in various neuropsychiatric disorders, such as epilepsy, schizophrenia, and autism spectrum disorders (Sohal & Rubenstein 2019). Research that aims to understand the underlying mechanisms of E-I balance can also contribute to understanding the pathophysiology of these disorders and potentially identifying therapeutic targets.

Finally, cooperativity between synapses is a crucial contributor to stable cortical activity. The low firing rates and individually weak synaptic connections found in neocortex necessitate correlated inputs onto individual neurons for signal propagation to occur (Brunel 2000; Koulakov 2009). Indeed, higher-order interaction between multiple neurons has been reported to

be an intrinsic feature of both neocortical structure and function (Yu et al. 2011). Excitatory synaptic connectivity exhibits a prevalence of specific triplet motifs (Perin et al. 2011; Song et al. 2005) and cliques of neurons (Litwin-Kumar & Doiron 2012). Activity in real neuronal networks show elevated clustering (Yu et al. 2011; Sadovsky et al. 2014; Rothschild et al. 2010; Pajevic & Plenz 2009; Orlandi et al. 2013; Shimono & Beggs 2015; Nigam et al. 2016; Dechery & MacLean 2018) that is dominated by triplet motifs which improve signal integration by coordinating the presynaptic pool (Chambers & MacLean 2016). Correlations between three units have been shown to be necessary to recapitulate spatiotemporal spiking patterns (Ganmor et al. 2015).

In the first chapter of this dissertation, we focus on canonical neocortical dynamics and ask why its maintenance is possible. We find that the answer relies on a specific mechanism of higher-order synaptic cooperativity. To reach this result, we used SNNs that were built to capture crucial factors of neocortical activity. In particular, they match the ratio of excitatory to inhibitory units, the connectivity parameters, and the measurements of rate, synchrony, and criticality observed in neocortex. Following this, we turn our attention to computation using similarly constructed SNNs in chapter two.

On computation

Computation refers to the manipulation, processing, and transformation of information according to well-defined rules or algorithms. It involves the use of symbols, representations, and operations to perform calculations, solve problems, simulate systems, or make decisions. In the case of the brain and models of the brain, computation often refers to the transformation of

sensory input according to the structural and dynamical properties of the brain, along with the brain's internal state, into appropriate cognitive states and behaviors (Churchland & Sejnowski 1992, Churchland & Grush 1999, Rolls 2021).

Since computation involves the use of representations to encode and manipulate information, philosophical discussions often revolve around questions of whether computational processes alone are sufficient to account for the emergence of semantics and understanding (Baggio 2018). We will not focus on this question in this dissertation; rather, we will assume that neural activity–and the structures that give rise to this activity–constitute representations, and that *correspondence* between a pattern of activity and some input or output suffices to establish a representation. A stronger case for neural activity being a representation can be made through causal manipulations to establish functional roles and teleology for the hypothesized representation (Baker et al. 2021). Because we observe specific activity patterns co-emerge with improvement in goal-oriented performance through task training, we believe we have satisfied these criteria for a representation as well.

Computation can sometimes be seen as a reductionist approach to understanding complex systems, since it breaks them down into smaller computational units, symbols, and rules. Philosophical discussions on emergence consider whether computational processes can fully capture the emergent properties of complex systems (Symons 2018), or if there are additional factors at play. This again is not within the scope of this dissertation; we will instead adopt the stance that all underlying features that support computation in the brain also make up generally-regarded emergent properties like behavior.

Finally, computation can be implemented in various physical substrates, such as electronic circuits, abstract mathematical structures, and, of course, networks of neurons in the brain. Philosophical considerations arise regarding the relationship between computation and physical instantiation and the role of embodiment in computation (Cooper 2013, Pezzulo et al. 2011). We will take these not as problems to address here, but rather as a lucky fact of brains and computer models that we can study computation in both. Again, there are surely differences between model and brain, but those differences here are more likely due to improper accounting for crucial details or stochasticity or feedback, etc., rather than because of something inherent to physicality.

We will now survey how our understanding of–and ability to wield–computation has advanced through the use of artificial neural networks (ANNs) and experiments in the brain.

On computation in artificial neural networks (ANNs)

The study of computation in neural networks began in earnest in the 1940s, when Warren McCulloch, a neurophysiologist, and Walter Pitts, a logician, proposed a mathematical model of a simplified neuron. They described how neurons in the brain could be represented as binary logic gates, processing information based on input signals and thresholds (McCulloch & Pitts 1943). This work laid the foundation for artificial neural networks. The perceptron model was the next significant milestone in the development of SNNs. It extended the McCulloch-Pitts neuron by introducing adjustable weights and an algorithm that allowed it to learn–through weight adjustments–using labeled training data (Rosenblatt 1957). The perceptron model was based on the idea of weighted connections between neurons in the brain, with the weights

adjusted during the learning process. The changes that take place in the models used in this work during learning are similarly made possible through changeable weights.

Interest in computation with ANNs experienced a surge in the 1980s with the development of new learning algorithms and the recognition of the potential of different neural network architectures. In 1982, John Hopfield introduced the Hopfield network, a type of recurrent neural network (RNN) and a predecessor of the architecture used in the work of this dissertation. Hopfield networks are known for their ability to store and retrieve binary patterns by leveraging the concept of energy minimization (Hopfield 1982). Then, in 1986, David Rumelhart, Geoffrey Hinton, and Ronald Williams published their work on backpropagation, a learning algorithm for training multi-layer neural networks (Rumelhart et al. 1986). While the concept of backpropagation was first introduced in the 1960s by Rosenblatt (Rosenblatt 1961, Schmidhuber 2022), Hinton's work was the breakthrough in implementation. It focused on the efficient training of deep neural networks using gradient-based optimization techniques, which laid the foundation for the renaissance of research with ANNs. Backpropagation has led to many breakthroughs in AI and the application of ANNs to other fields of science, including their adoption into computational neuroscience. A variant of backpropagation through time (BPTT, Robinson & Fallside 1987, Werbos et al. 1988, 1990), itself an extension of Rumelhart and colleagues' algorithm to recurrent networks, is used in this work (Bellec et al. 2020).

On cortical computation

Our understanding of computation through networks of neurons in the neocortex is continually evolving, and this work seeks to build upon and contribute to that understanding. Several key

features of neocortex have been identified as potentially beneficial to computation, and are built into the spiking neural network models that are used in this work. They are:

Recurrence: Recurrent connections within the neocortex play a crucial role in information processing. These connections enable feedback loops, allowing information to flow in both bottom-up and top-down directions between cortical–as well as subcortical–regions. Recurrent connectivity contributes to the generation of complex dynamics and the ability to integrate information over time. It supports processes like attention, time-dependent tasks, and context-dependent processing in artificial networks (Goehring et al. 2019, Kietzmann et al. 2019). Theoretical work supports the computational importance of recurrent connectivity; in a network that maximizes the number of stored patterns of activity in a robust fashion, bidirectionally coupled pairs of neurons are over-represented and more strongly weighted (Brunel 2016). The same study demonstrated that most connections in the network went to 0, indicating that it is *sparse* recurrent connectivity which may be most beneficial for storage of representations that underlie computation. This sparse recurrent connectivity is experimentally observed in neocortex (Lefort et al. 2009, Song et al. 2005, Wang et al. 2006, Perin et al. 2011, Billeh et al. 2020) and is optimal in several other experiments with different models (Clopath et al. 2021, Brunel et al. 2004, Chapeton et al. 2012, Clopath et al. 2014).

Sparse Coding: Not only do neocortical circuits exhibit sparse connectivity, they also display sparse activity. Only a subset of neurons is active at any given time, which points to sparse coding (Olshausen 2002). Sparse coding allows efficient representation of information, reduces redundancy, and supports the formation of selective and invariant representations (Olshausen &

Field 2004, Chalk et al. 2018). In addition to supporting coding, sparse activity is also believed to be a strategy for minimizing metabolic costs of computation (Hasenstaub et al. 2010, Sengupta et al. 2013, Verzi et al. 2018), and this has led to the interest of engineers in spiking neural networks and neuromorphic systems. The topic of spiking and its computational benefits is explored further in a later section.

Plasticity and Learning: Neural plasticity, particularly synaptic plasticity, is a fundamental mechanism in the neocortex. It allows the strengthening or weakening of connections between neurons based on experience and learning. Plasticity mechanisms, such as Hebbian learning and spike-timing-dependent plasticity, contribute to the formation and modification of cortical representations for computation (Abbott & Nelson 2000). For our models, we chose to use a global learning rule that is more suited to machine learning than neocortex (Bellec et al. 2020), although it can be refactored into a learning rule with possible biological merits (Gerstner et al. 2018). We did so because our main objective was to establish computations in our SNNs through synaptic changes, which we could then investigate; this method accomplished this goal.

While significant progress has been made in understanding computation in the neocortex, many questions remain. Researchers are actively studying the relationship between structure, dynamics, and computation in neocortex, and computational models have been a powerful tool for those inquiries. In such studies, computation is often measured or targeted through metrics such as information capacity (e.g. in Brunel 2016), dynamic range (e.g. in Shew et al. 2009), information transmission (e.g. in Mejias & Longtin 2012), and decodability (e.g. in Cohen et al. 2020). Since computation involves the manipulation and transmission of information, and these

measures provide insights into the capacity and precision of such operations, they also capture important facets of computation. More recently, ANNs have been levied in our study of neocortical computation, so that we can directly measure computation through performance on a task that makes computational demands and alters the network accordingly.

On ANNs, behavior, and neuroscience

The ability of ANNs to exhibit dynamics, form representations, and perform learned behaviors has made it a favored tool in recent theoretical neuroscience research. Establishing ethologically relevant learned behaviors in ANNs is a promising top-down approach (Eliasmith & Trujillo 2014) to creating biofidelic models with explanatory power. This approach has been demonstrated to create correspondences between neocortical activity and several varieties of ANNs.

For example, using a hierarchical CNN, Yamins & DiCarlo demonstrated that optimizing for human-level performance in object recognition tasks also led layers of their model to be highly predictive of V4 and IT spiking responses to complex images (Yamins & DiCarlo 2014). Therefore, optimizing a minimally biofidelic model for behavior can suffice to create systems with similar computational signatures to neocortex. More recent work that carefully characterized the behavioral patterns of primates to individual images revealed that deep CNN models cannot account for all such patterns (Rajalingham et al. 2018). This work suggests that large-scale behavioral benchmarks could be direct guides for searching for appropriate computational models, and that alternative models may be necessary to precisely capture all computations that underlie primate object vision.

Indeed, a direct comparison of feedforward and recurrent neural network models revealed that only recurrent models could account for the multi-region, dynamic transformations of representations in visual cortex (Kietzmann et al. 2019). RNNs have been a favorite architecture to use for modeling temporal / memory-related processes in neocortex. For example, RNNs' reaction times on an image recognition task predicts human reaction times for the same image better than several parameter-matched feedforward models (Spoerer et al. 2020). These results suggest that recurrent connectivity, a hallmark of neocortex, may be crucial for understanding the dynamics of cortical computation over time.

In terms of non-correspondence to animal behavior, catastrophic forgetting is a non-biofidelic fact that has badgered ANN research. By building models that solve the problem of catastrophic forgetting, researchers seek to discover why biological brains are uniquely good at robust memory and continual learning. For example, by using dynamical systems analysis on RNNs that have been trained on sequential tasks and with a learning rule to minimize catastrophic forgetting, one possible explanation has been posited. These models re-use similar dynamical structures across similar tasks, revealing shared computations as a possible cortical solution (Duncker et al. 2020). Rapid multitask learning in RNNs has also been shown to be possible under conditions of localized and sparse synaptic plasticity with local learning rules, which is more directly analogous to neocortex (Masse et al. 2022).

While vision has been the most popular domain for modeling and studying computation, ANNs have also been applied to other systems. For example, grid cells and place cells are involved in spatial navigation, and ANNs have been used to explore these mechanisms. Banino and

colleagues used a deep reinforcement learning approach on RNNs (called the GridNet) to model cell representations and their role in navigation (Banino et al. 2018). They demonstrated that optimization on a path integration task led to the emergence of grid-cell-like representations in their model, as well as other entorhinal cell types. These representations allowed model agents to conduct shortcut behaviors like mammals are capable of, and overall their results support the neurobiological theory that grid cells are critical for navigation.

Taken together, these results suggest that a behavior-first, top-down optimization approach, combined with an exploration of different architectures and levels of detail, is promising for directly studying computation in ANNs and determining the features of neocortex that explain computation (Eliasmith & Trujillo 2014). However, attempting to explain computation in ANNs comes with several challenges.

On interpretability in neural network models

Thinking back to our atom in the asterisk–only the second explanation (that included such abstract concepts as "dissertation") truly gave us a *why* answer. Researchers have discovered that trying to understand *why* neural networks work is a substantial challenge, and surprising given that we built them. Deep neural network architectures operate through many interconnected layers of artificial neurons, making their inner workings highly intricate. In recurrent neural network architectures, the same set of weights is present across sequential time steps. This allows RNNs to perform temporal tasks, but it also creates complex interactions and makes it harder to discern the contribution of each parameter to the overall output. In addition, more biofidelic neural network models often employ nonlinear activation functions in their model neurons,

adaptation mechanisms, and additional sources of stochasticity. This enables them to capture some of the nonlinear behavior of the brain. However, this nonlinearity further complicates interpretability, as it becomes challenging to trace back how specific input features influenced the final model output.

Addressing the challenge of interpretability in ANNs–and also in ANN models of the brain–is an active area of research. Researchers are exploring techniques such as model visualization (Karpathy et al. 2015), attribution methods (Sundararajan et al. 2017), and relevance propagation (Bach et al. 2015) to shed light on the internal workings of neural network models. Models can also be built with attention mechanisms, which allow the network to alternately focus on different parts of the input sequence. Researchers can then work backwards to identify the important elements that contributed to the network's predictions (Bahdanau et al. 2014). However, achieving full interpretability–being able to answer satisfactorily *why* a network is capable of solving a task–remains a significant, ongoing challenge. Neural network models are so challenging to interpret that they have been referred to as black boxes, and criticisms have been levied that it does not make sense to use black box models to study black box brains.

Furthermore, the form of a satisfying interpretation for ANN function can sometimes differ from what would constitute interpretation for neuroscientists (Kar et al. 2022). Kar and colleagues argue that interpretability of ANNs for neuroscience necessarily means correspondence between brain and model. When ANNs contain features that do not map to neuroscientific constructs, this renders the models uninterpretable. While these model-specific features could ultimately have no bearing on the brain’s functions, and some correspondence to the brain should be a first step in

modeling (i.e. why I am focused on building biofidelic models), correspondence far from constitutes interpretability. In fact, I argue below that *the confusion of **interpretable** ANNs with ANNs that **predict** brain activity has been a major conceptual hindrance in our search for good explanations of cortical computation.*

The authors also state that the extent to which a model should be made interpretable depends on the intended purpose of the model; thus a model that explains one set of experiments may not explain another. I believe that this is a slightly incorrect way of asserting (correctly) that good explanations can be found at all levels, and there is real explanatory power to abstractions like "dissertation" and "oscillation". One does not need to dive to the finest level of detail to seek interpretability in models in all instances. Insights into computation that have been gained using ANNs that do not predict all the detailed structural, dynamical, or behavioral features of real brain systems still constitute genuine knowledge.

In fact, if one is only interested in building a model that is precisely predictive of the brain, then we should build a precise replica of the brain. Once we have the experimental data and technology to do so, what will we have achieved in terms of further understanding? Thus a major pitfall of the prediction agenda is that it does not generate new knowledge. The best it can achieve–which is what it aims to achieve–is recapitulation of prior knowledge in newly engineered systems.

There is one additional, related consideration for researchers who seek to build ANNs to make the brain more interpretable. It is that even when ANNs are highly predictive of the brain, they

may have arrived there due to specific modeling choices. If models are to be purely evaluated based on their predictive abilities, then we should be satisfied with such an approach. But we find that we as a field are not satisfied, and are instead actively campaigning against it (Schaeffer et al. 2022). This is because researchers should and do genuinely care about *underlying mechanisms* as part of understanding the brain.

Sometimes, under very similar experiments, whether or not a study reveals a "why" answer depends on whether or not the researchers asked a "why" question to begin with. For example, Bashivan and colleagues used a deep ANN ventral stream model to evolve images that optimally drove the activity of neural sites in macaque V4 (Bashivan et al. 2019). Not only does the "V4" layer of their model closely predict V4 recordings, they also succeeded in controlling the activity of V4 neural sites individually and together. The authors comment that ANN models are opaque to direct understanding, and they therefore chose to focus on application of ANNs' predictive power for neural control.

Neural control is an extremely important therapeutic goal of neuroscientific research, but contrast the explanatory power of the above work with the work of Ponce and colleagues, who, around the same time, also used deep image synthesis to study the primate ventral stream (Ponce et al. 2019). Through a deep ANN and an evolutionary algorithm, they generated images that maximized neuronal firing in macaque IT. They interpreted these images as possible internal representations of neurons in IT cortex, forming a set of components used in the computation of visual scenes. Furthermore, the authors comment on the potential of this approach to reveal the internal representations of other brain systems. Together, these two studies demonstrate that it is

possible for interpretability to be gained through a shift in perspective. This gives us one good reason to be optimistic about our future–and even retroactive–forays into greater interpretability.

Thus I wholly agree with Kar and colleagues in their statements that 1) synergy is needed between AI and neuroscience to interpret ANNs, and that 2) ANNs (especially ones with features that do not correspond to contemporary theories of the brain) can be hypothesis generators for new computational mechanisms. Validating models against neural measurements and behavior is a first step. We can next apply the same approaches that AI researchers use to explain ANNs, which is to elucidate the models' inner representations, transformations, and decision-making processes.

Fortunately, when it comes to interpreting task-trained spiking neural network models, there are reasons to be optimistic and even more tools at our disposal. Not only are SNNs performing computations in potentially spike-based and more biofidelic ways, they are also capable of being explored using the same suite of tools which experimentalists have developed for directly interpreting neocortical data. Interpretability in SNNs can have an additional set of ambitions that other ANNs would struggle to meet. Since they are built with greater biofidelity at a greater level of detail, there is the potential to explain their function–and extend those findings to the brain–on the level of single spikes and synaptic integration. The power of SNNs to yield insights on mechanism has been previously harnessed to study function in terms of dynamics; researchers are just beginning to use SNNs to study function in terms of computation.

On spiking

Several lines of evidence point to the importance of spikes for computation (Brette 2015). The relative timing of spikes has been demonstrated to carry information about stimulus features in neocortex (deCharms & Merzenich 1996). Populations of neurons in the primary auditory cortex can coordinate the timing of their action potentials in a way that signals stimuli, even while firing rates remain unchanged. Spiking networks also exhibit robustness to noise and variations in input signals (Calaim et al. 2022). Spiking neurons can perform computations through the integration of asynchronous and distributed input events. Inputs that arrive at temporal offsets can still together drive a neuron to fire. The discrete nature of spikes, their sparsity in time, and the temporally permissive nature of their generation can help mitigate the impact of noisy inputs or other fluctuations, especially when neuronal units themselves are also leaky (Sharmin et al. 2020). This makes neural network models with spiking units most suitable for modeling neural systems operating under realistic, noisy conditions.

Spikes also enable efficient coding, as they represent information through selective, transient activation. This sparse coding is believed to enhance the computational capacity of neural networks. It has been shown that balanced synaptic input currents–which is more biofidelic of neocortex overall–generate fewer spikes per second, and that these spikes are more informative, carrying more bits per spike (Sengupta et al. 2013). Thus, approximately balanced synaptic currents in spiking networks like neocortex can promote both computational efficiency and energy efficiency.

Finally, in addition to bolstering theoretical measurements like information, individual spikes have genuine consequence in the world (Brette 2015). They have been demonstrated to be useful

to animals to drive their own behaviors. Electrophysiological recordings and optogenetic stimulation has revealed that the earliest stimulus-evoked spikes in mouse V1 are preferentially weighted for guiding behavior (Day-Cooney et al. 2021). In line with this preference, single spikes themselves can be sufficient. Mice are capable of performing a visual discrimination task using V1 activity within a narrow time window; during this window the vast majority of neurons discriminating the stimulus fire either one or no spikes (Resulaj et al. 2018).

Considering these lines of evidence that spikes are consequential for neural function, building models that utilize spiking units is a promising approach to answering the "why" questions of cortical computation that we care about. Previously, spiking neural network simulations have strived to capture the structural and dynamical properties observed in real neocortical networks. They have been–and are continuing to be–used to address structure function questions. More recently, SNNs have gained the capability to capture some of the behavioral and potentially computational properties of neocortex as well.

The development and research of trained SNNs is a relatively new field, with the first direct application of backpropagation to SNNs reported in 2016 (Lee et al. 2016). Previously, task-performing SNNs were mostly made possible through translation of pre-trained RNNs' continuous activations into spikes (Bu et al. 2023). However, RNNs are not congruent to SNNs outside of a narrow set of constraints (Schmutz et al. 2023). In addition, in the case of translation of pre-trained RNNs, discrete spikes and temporal integration are not original, crucial parts of learning the appropriate computations. Now, directly trained SNNs have the potential to offer novel insights into the mechanisms of neural computation.

The energy efficiency of spiking networks is of increasing interest in the development of low-power neuromorphic systems. Neuromorphic engineering is a field that aims to emulate the functionality of biological neural networks in hardware systems and through the use of SNNs. Notable examples include the BrainScaleS, TrueNorth, Intel Loihi projects (Davies et al. 2021). Owing to the relative novelty of trained SNNs and the focus of the neuromorphic field on engineering rather than understanding, there is little pre-existing work on interpretability of SNNs, and even less on using trained SNNs as an avenue for probing cortical computations. Scientists are beginning to apply traditional methods of interpretability in ANNs to SNNs, such as model visualization (Kim & Panda 2021). Through the work in this dissertation, we seek to contribute to interpretability of SNNs as a way of interpreting cortical computation as well.

On account of the demonstrated computational benefits of sparse spiking and sparse recurrent connectivity–defining features of both neocortex and spiking neural network models–as well as the driving power of task optimization, we are optimistic that our work with SNNs is one additional step towards a good explanation of neocortical computation. More than just a description, it offers mechanistic accounts as to 1) *why* stable dynamics are possible in neocortex, and 2) *why* computation becomes possible through specific synaptic changes and resulting dynamic changes that occur through learning.

To address the first *why*, we created SNNs through a grid search for architectures that yielded realistic spiking activity (Roxin et al. 2011, Salinas & Sejnowski 2001). We demonstrated a link between the maintenance of this activity and triplet motifs. We showed that higher order coordination of synapses is always present during sustained, low-rate, asynchronous activity, and

that this coordination takes the form of transitions in time between specific triangle motifs (Bojanek & Zhu et al. 2020). These motifs summarize the way spikes traverse the underlying synaptic topology (Chambers & MacLean 2016). At this point, I became interested in neocortical computation, which occurs on a substrate of stable, asynchronous, low-rate neuronal activity.

To address the second *why*, we used very similar SNNs (Zhu et al. 2020); the crucial difference is that these SNNs were trained to perform an ethologically relevant task. We find that SNNs selectively adjust firing rates in response to the stimulus input, and that excitatory and inhibitory connectivity between input and recurrent layers changed in accordance with this rate modulation. Input channels that began by responding more to a specific input developed stronger connections to recurrent excitatory units over training, while channels that responded more to the other input developed stronger connections to inhibitory units. Furthermore, recurrent inhibitory units which were tuned to one input over the other strengthened their connections to recurrent units of the opposite tuning. We draw a biological parallel between our observed recurrent connectivity pattern and cross-orientation-tuning inhibition in visual cortex (Morrone et al. 1982, Eysel et al. 1990, Katzner 2011), as well as hypothesize on the role of parvalbumin-expressing (PV) interneurons (Bos et al. 2020, Kepecs & Fishell 2014, Lagzi et al. 2021).

So let us begin.

CHAPTER 1

Cyclic transitions between higher order motifs underlie sustained asynchronous spiking in sparse recurrent networks

This work was previously published as: Bojanek, K.*, Zhu, Y.*, MacLean, J. N. (2020). Cyclic transitions between higher order motifs underlie sustained asynchronous spiking in sparse recurrent networks. *PLOS Comput Biol, 16*(9), e1007409. https://doi.org/10.1371/journal.pcbi.1007409. *co-first-authors.

AUTHOR CONTRIBUTIONS

I contributed to this work through uncovering and characterizing the main result, i.e. the necessity of cyclic transitions between triplet motifs for self-sustained spiking activity in our SNN simulations. I performed triplet motif analyses of functional and recruitment graphs of completed vs. truncated runs. The text is written almost entirely by me, with edits by Jason MacLean and Kyle Bojanek. I assembled the figures and created the plots in Figures 1-5 and 7. Kyle Bojanek first originated this work through collaboration with Brendan Chambers. He also built the infrastructure of grid search and simulation of SNNs in Julia. Finally, he performed the majority of statistical testing and created the plots in Figures 6 and 8-10. This work would not have been possible without all of us working in close collaboration. Kyle Bojanek is listed first as co-first-author per alphabetical order.

ABSTRACT

A basic—yet nontrivial—function which neocortical circuitry must satisfy is the ability to maintain stable spiking activity over time. Stable neocortical activity is asynchronous, critical, and low rate, and these features of spiking dynamics contribute to efficient computation and optimal information propagation. However, it remains unclear how neocortex maintains this

asynchronous spiking regime. Here we algorithmically construct spiking neural network models, each composed of 5000 neurons. Network construction synthesized topological statistics from neocortex with a set of objective functions identifying naturalistic low-rate, asynchronous, and critical activity. We find that simulations run on the same topology exhibit sustained asynchronous activity under certain sets of initial membrane voltages but truncated activity under others. Synchrony, rate, and criticality do not provide a full explanation of this dichotomy. Consequently, in order to achieve mechanistic understanding of sustained asynchronous activity, we summarized activity as functional graphs where edges between units are defined by pairwise spike dependencies. We then analyzed the intersection between functional edges and synaptic connectivity- i.e. recruitment networks. Higher-order patterns, such as triplet or triangle motifs, have been tied to cooperativity and integration. We find, over time in each sustained simulation, low-variance periodic transitions between isomorphic triangle motifs in the recruitment networks. We quantify the phenomenon as a Markov process and discover that if the network fails to engage this stereotyped regime of motif dominance "cycling", spiking activity truncates early. Cycling of motif dominance generalized across manipulations of synaptic weights and topologies, demonstrating the robustness of this regime for maintenance of network activity. Our results point to the crucial role of excitatory higher-order patterns in sustaining asynchronous activity in sparse recurrent networks. They also provide a possible explanation why such connectivity and activity patterns have been prominently reported in neocortex.

1 INTRODUCTION

Network connectivity shapes dynamics in many systems and on many scales, ranging from gene transcription networks to epidemic spreading (Barzel & Barabasi 2013). In the brain, neocortical

architecture supports myriad complex functions. Before any of these functions can occur, neuronal spiking activity must be maintained throughout the lifespan of an animal. Stable maintenance of spiking activity—both as "background" activity and when tasked with inputs and outputs—is a basic function that arises from the structure of synaptic connectivity in the brain. Given the fact that the vast majority of excitatory synapses are weak and connections are sparse and recurrent, achieving stable activity is highly non-trivial (Perin et al. 2011; Song et al. 2005; Seeman et al. 2018; Zylberberg et al. 2017). Theoretical and experimental studies have characterized several architectural features that have the capacity to promote and shape spiking activity, such as a heavy-tailed synaptic weight distribution, excitatory clustering and the ratio between incoming and outgoing connections (Song et al. 2005; Lefort et al. 2009; Teramae et al. 2012; Litwin-Kumar & Doiron 2012; van Vreeswijk & Sompolinsky 1996). Additionally, dynamical properties of ongoing activity, such as a balance between excitation and inhibition (Litwin-Kumar & Doiron 2012) and correlated spiking (Reimann et al. 2017), are shaped by connectivity and in turn impact the continuation of spiking activity.

Neocortical activity in the awake mammal is characterized by low-rate, near-critical, and asynchronous dynamics. Any theory which purports to explain stable activity in neocortex must take these features of activity into consideration. Previous work has demonstrated that sustained activity co-occurs within a specific range of firing rates, supported by a balance between excitatory and inhibitory conductance (Vogels & Abbott 2005; Brunel 2000]. Firing rates that are too low or too high contribute to instability within the network (Vogels & Abbott 2005). Neocortex is also often characterized as having critical or near-critical dynamics (Beggs & Plenz 2003). In a practical sense, this entails activity which, in the absence of external input, is

maintained without becoming epileptic nor dying out, and which follows a power law distribution in its active population size. The idea that neocortex operates near a critical point has a long history in neuroscience, going back to Alan Turing (Turing 2009), and has been implicated in a number of desirable properties for neural networks (Karimipanah et al. 2016). For example, networks tuned near the critical point display maximum information transmission (Beggs & Plenz 2003), information storage (Haldeman & Beggs 2005), and computational power (Bertschinger & Natschläger 2004). Any incoming signals interact with and rely upon the activity state already present in a network, or the "background" activity, which in neocortex is generally asynchronous and irregular. Several theoretical studies have focused explicitly on self-sustained activity in the asynchronous state and the propagation of inputs in this state. They found a complex relationship between synaptic strength and firing rate in the maintenance of an asynchronous spiking regime (Kriener et al. 2014; Brunel 2000; Zerlaut et al. 2019). Here we focus on said background activity and ask how it may be stably maintained in the absence of inputs. We build on previous theoretical and experimental studies by uncovering the dynamic mechanisms behind self-sustained activity using models that capture crucial factors of neocortical activity. Namely, the models match the ratio of excitatory to inhibitory units, the connectivity parameters, and the measurements of rate, synchrony, and criticality observed in neocortex.

Experimental results suggest that pairwise measurements alone may be insufficient to explain network dynamics. Higher-order patterns in both structure and activity have been reported to be intrinsic features of neocortex (Yu et al. 2011) and may be key to its function. Excitatory synaptic connectivity displays a prevalence of specific triplet motifs (Perin et al. 2011; Song et

al. 2005) and cliques of neurons (Litwin-Kumar & Doiron 2012). Activity in real neuronal networks exhibit elevated clustering (Yu et al. 2011; Sadovsky et al. 2014; Rothschild et al. 2010; Pajevic & Plenz 2009; Orlandi et al. 2013; Shimono & Beggs 2015; Nigam et al. 2016; Dechery & MacLean 2018) that is dominated by triplet motifs which can improve synaptic integration by coordinating the presynaptic pool (Chambers & MacLean 2016). Moreover, correlations between three units are necessary to recapitulate spatiotemporal spiking patterns (Ganmor et al. 2015). Computationally, triplet motifs may improve coding (Cayco-Gajic et al. 2015; Shi et al. 2015) and enhance perceptual accuracy and the prediction of responses in visual cortex (Dechery & MacLean 2018; Shahidi et al. 2019). The causal relation from underlying synaptic connections to functional connections and correlations in activity within a network is complex (Chambers & MacLean 2016; Chambers et al. 2018). The presence of certain motifs in synaptic graphs greatly affects the strength of higher-order correlations in network neuronal activity (Jovanović & Rotter 2016). Furthermore, the low firing rates and weak synaptic connections found in neocortex necessitate correlated inputs onto individual neurons (Brunel 2000; Koulakov 2009). Here we build functional networks and then identify the intersection of the functional network with the synaptic network. We then analyze this subset of active synapses, or recruitment network, to study the dynamics that correspond to the activation of the underlying synaptic connectivity. Our focus is on the presence of higher order functional motifs and its relation with the maintenance of an asynchronous spiking state. First, a novel algorithmic approach is used to build large numbers of sparsely-connected recurrent spiking neural network models. Simulations are run on these models to explore the roles of realistic dynamics and higher-order interactions in sustained spiking activity. These models are recurrent and sparsely-connected; they are composed of excitatory and inhibitory adaptive exponential leaky

integrate-and-fire (AdEx) neurons with conductance-based synapses (Brette & Gerstner 2005). Network topology parameters are varied and informed by connectivity seen in cortex (Song et al. 2005; Lefort et al. 2009; Teramae et al. 2012). By design, the models closely approximate both the statistics of connectivity as well as spiking activity in neocortex (Zerlaut et al. 2019; Ecker et al. 2010; Renart et al. 2010). We find that simulations on the same network topology can either spontaneously stop (truncated run) or show sustained activity (sustained run), corresponding to different sets of initial membrane voltages. The dichotomy between sustained runs and this subset of late-truncating runs on the same networks, in addition to our ability to algorithmically construct and simulate very large numbers of networks, provided us with the opportunity to study the conditions which underlie sustained asynchronous activity.

2 RESULTS

2.1 Network construction and simulation

Each spiking neuronal network was composed of 4000 excitatory and 1000 inhibitory adaptive exponential leaky integrate-and-fire (AdEx) units (Brette & Gerstner 2005). Synaptic connections were recurrent, sparse and conductance-based (Fig 1A). Excitatory connection strengths followed a heavy-tailed, log-normal distribution, where $\mu = -5.0*10^{-5.0}$ nS, $\sigma = 0.5$ nS, corresponding to a mean of 1.13 nS and a variance of 0.365 nS (Fig 1B). Networks therefore had a large number of weak connections and few strong excitatory synapses. Given that local connectivity in neocortex is clustered (Perin et al. 2011; Song et al. 2005)—although global statistical patterns of connectivity cannot be precisely determined from paired patch clamp recordings (Vegué et al. 2017)—we also created clusters within our networks. For each network

we defined 50 clusters and randomly assigned each excitatory unit to two of these clusters.

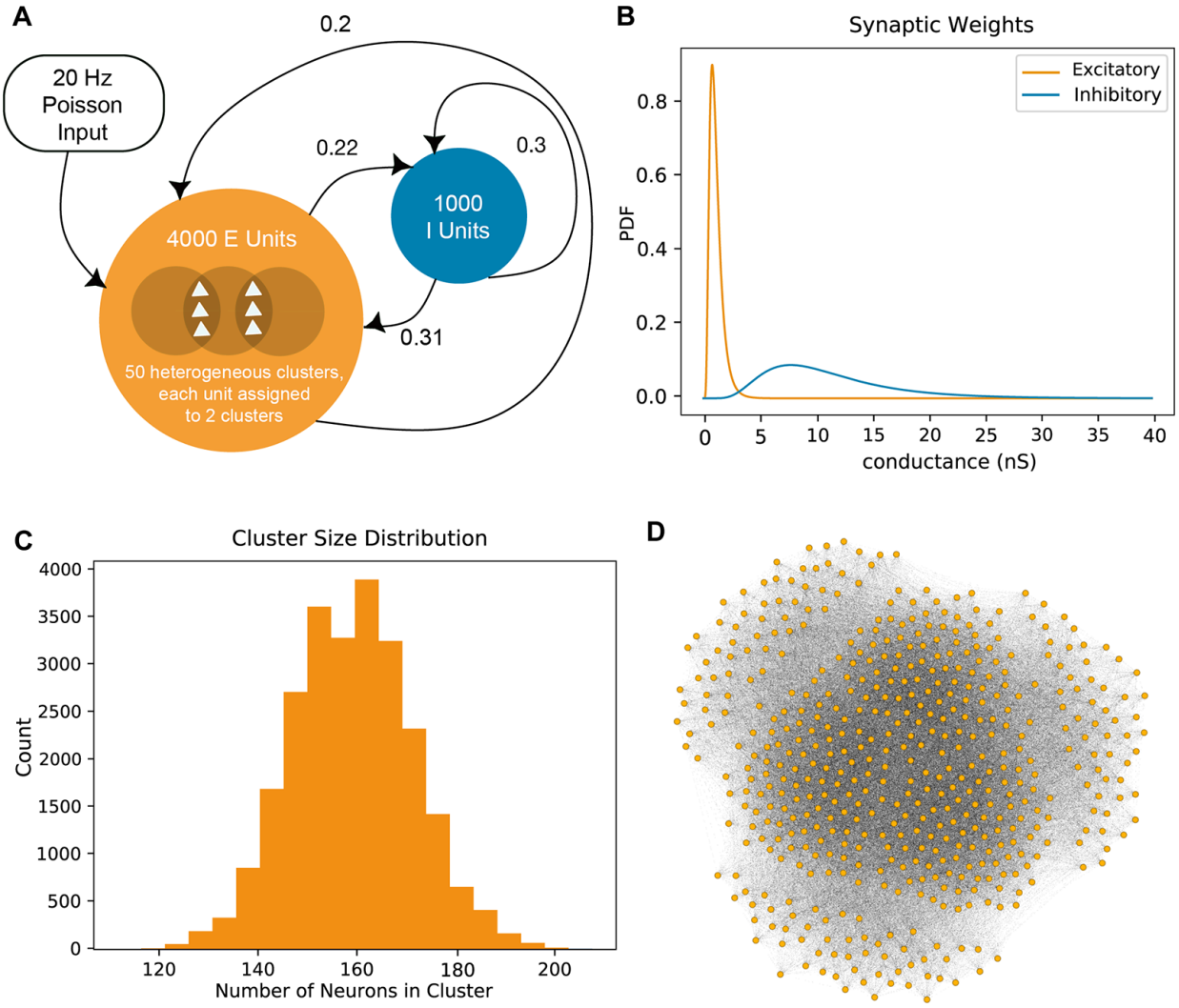

Figure 1.1: Network construction and search

A. Our networks were constructed with 4000 clustered excitatory and 1000 unclustered inhibitory units. Probabilities of connection (those from excitatory to excitatory units, and from inhibitory to inhibitory units) were inferred from experimental literature and determined via grid search (those from excitatory to inhibitory units and vice versa). Simulation runs began with 30ms of 20Hz Poisson input onto a subset of 500 units.

B. Synaptic weights followed a log-normal (heavy-tailed) distribution. Synapses were conductance-based, so weights are in units of nanosiemens. Connections originating from inhibitory units were 10x stronger than those from excitatory units.

C. For each network, we defined 50 clusters in total and randomly assigned each excitatory unit to two of these clusters. The wiring probability between units within the same cluster is twice that of units in different clusters. This resulted in heterogeneously-sized clusters. Here we show the cluster size distribution (in counts) for 500 networks.

D. Visualization of a subset of 300 clustered excitatory units in our network.

Intra-cluster connection probability was twice that of inter-cluster connection probability. This resulted in excitatory clusters of different sizes (mean = 158.40, std = 12.27 units per cluster) (Litwin-Kumar & Doiron 2012). The excitatory subgraphs had an average connection density (ratio of the actual over the total possible number of connections) of 0.211 (std: $1.10*10^{-4}$), of which 22.4% were reciprocal (std: 0.02%). Total density within each cluster was 0.389 (std: $3.11*10^{-3}$) and density between units in different clusters was 0.196 (std: $9.94*10^{-5}$) (Fig 1C and 1D).

At the beginning of a simulation trial, or run, initial resting membrane voltages were randomly assigned from a uniform distribution of -60 to -50 mV across all units. Activity was then initiated by 30 ms of 20 Hz Poisson input onto a set of 500 randomly chosen excitatory units (Fig 1A).

2.2 Algorithmically identifying networks for analysis

We focus our study exclusively on network simulations which displayed naturalistic spiking dynamics. In order to evaluate large numbers of networks for biological realism while minimizing sampling bias, models were constructed, simulated, and scored algorithmically. We restricted the search for viable topologies to a range of connection likelihoods bounded by experimental observations (Chambers & MacLean 2016). This should not be interpreted to suggest that these connection likelihoods are the only viable solution for realistic spiking activity—we did not comprehensively survey the range of possibilities.

We identified viable topologies iteratively; in the first iteration, we performed a low resolution grid search for connection probabilities (Fig 2A). The values for the probabilities of connection

from excitatory to excitatory units, $p_{e\rightarrow e}$, and from inhibitory to inhibitory units, $p_{i\rightarrow i}$, were inferred from experimentally measured connection probabilities in neocortex. They were 0.20 and 0.30 respectively (Song et al. 2005; Chambers & MacLean 2016). We performed grid search for the values of $p_{e\rightarrow i}$ and for $p_{i\rightarrow e}$. The existence of multiple classes of interneurons in neocortex means that this parameterization is a generalization—the values used for $p_{i\rightarrow i}$ and found for $p_{e\rightarrow i}$ and $p_{i\rightarrow e}$ represent summary values for a generic interneuron. During grid search, we rewired topologies within a limited range of $p_{e\rightarrow i}$ and $p_{i\rightarrow e}$, to identify sets of connection likelihoods that resulted in networks with sustained low-rate, near-critical, and asynchronous dynamics as observed in neocortex (Zerlaut et al. 2019; Yu et al. 2011; Sadovsky & MacLean 2014; Rothschild et al. 2010; Pajevic & Plenz 2009; Orlandi et al. 2013; Shimono & Beggs 2015; Nigam et al. 2016; Dechery & MacLean 2018; Chambers & MacLean 2016; Ganmor et al. 2015; Cayco-Gajic et al. 2015; Shi et al. 2015; Shahidi et al. 2019; Brette & Gertsner 2005; Ecker et al. 2010; Renart et al. 2010; Roxin et al. 2011; Griffith & Horn 1966; Koch & Fuster 1989; Hromádka et al. 2008). Criticality was measured using a branching parameter that is the ratio of active descendant units to active ancestor units across time (Beggs & Plenz 2003). A value of 1—where the number of active descendants is equal to the number of active ancestors—indicates critical dynamics (Fig 2B). We used a fast, on-line synchrony heuristic (variance of the mean voltage divided by the mean of voltage variances, see Methods) for the sake of grid search speed. A run was considered to be asynchronous if this heuristic value was below 0.5. Runs below this threshold correspond to a high mean Van Rossum distance which we employed throughout the remainder of the study (van Rossum 2001; Houghton & Kreuz 2012) (see Methods).

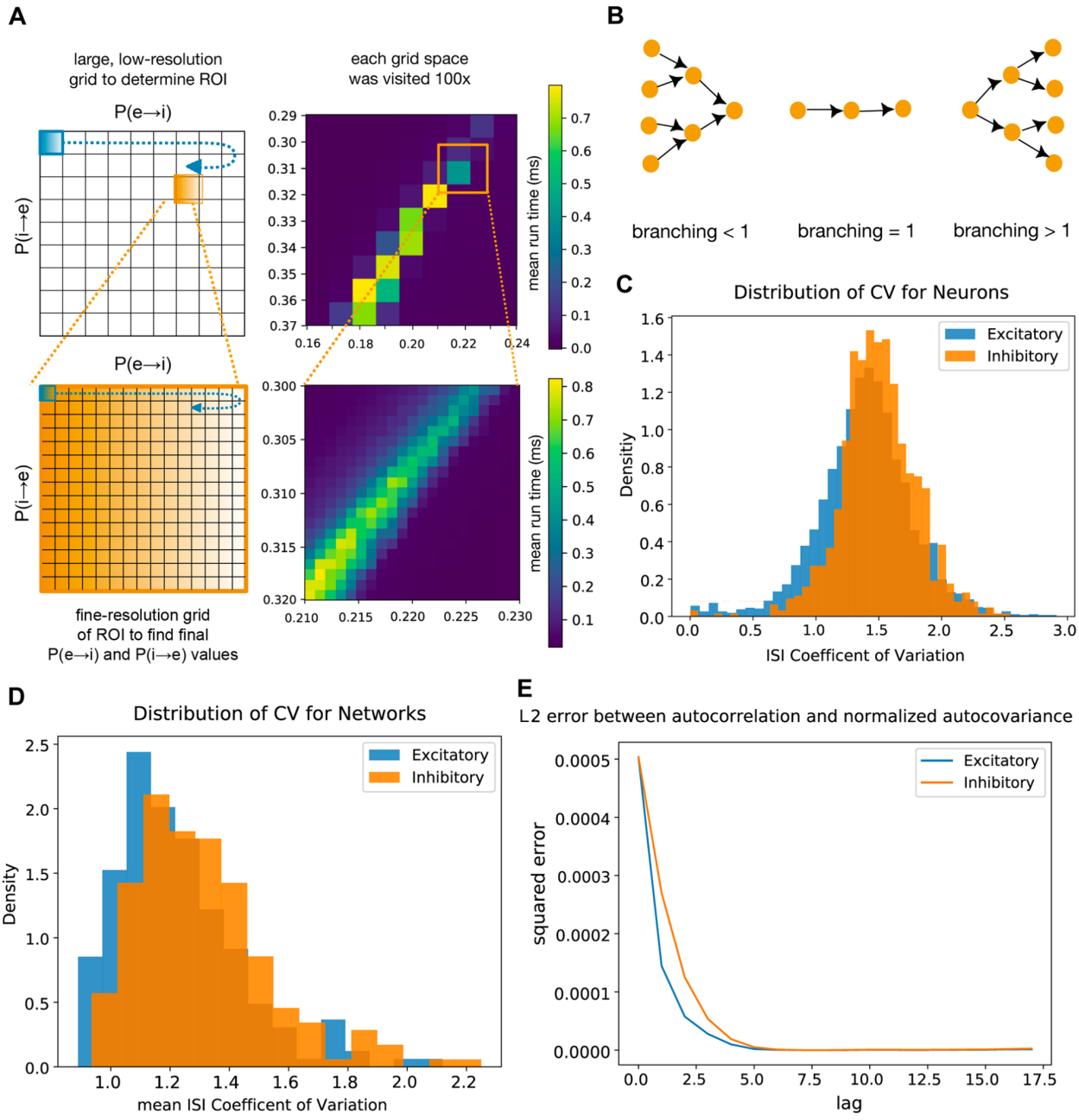

Figure 1.2: Grid search yields networks with dynamics resembling neocortex

A. We performed two rounds of grid search for the topological parameters that yielded consistent low-rate, critical, and asynchronous dynamics. The first search was at a lower resolution to narrow down our region of interest, and the second was at a finer resolution.

B. One scoring metric we used was branching. The branching parameter (Beggs & Plenz 2003) is a proxy for criticality. It measures the ratio of active descendant units to active ancestor units. A branching value of 1 indicates a balanced (or critical) network, which is the value we optimized for.

C. Distribution of the interspike interval coefficient of variation for individual neurons.

D. Distribution of the interspike interval coefficient of variation for networks. E: L2 error between autocorrelation and normalized autocovariance.

The first iteration of grid search isolated a region of interest, and we next used a higher resolution grid to find specific topologies each with the same probabilities of connection but differing in the specifics of connections (Fig 2A). To find these topologies we used the best results obtained from the second round of grid search, which were $p_{e\rightarrow i} = 0.22$ and $p_{i\rightarrow e} = 0.31$. These values were chosen for their ability to yield network simulations that had both low spiking rates and a high proportion of completion (see Methods).

These connectivity parameters were used to generate 2,761 synaptic topologies, where each unique topology is referred to as a network. For each network we randomly created 100 sets of input units, with 500 excitatory units per set. We ran 50 simulations on each set of input units, where each simulation began with different membrane voltages for all units. Each simulation lasted for as long as spiking activity was sustained, up to a maximum of 1 second. The spiking activity of each run on each network was then scored according to the average firing rate, the level of asynchrony, how balanced—or critical—the network was, and the duration of time over which spiking activity was maintained (see Methods). If a network's average firing rate within excitatory units for all complete runs out of the initial 500 was less than 8 spikes/second, we added this network to the set of low-rate networks for subsequent analyses. High-rate networks were eliminated to maintain consistency with the low spike rates generally measured in neocortex. This yielded a final count of 87 low-rate networks. For each of these networks, we determined the set of input units which led to the most consistently sustained simulations, with the trade-off of rate increasing slightly. We will refer to these as a network's optimal input units.

Optimal input units were then fixed and used to generate 100 additional runs on each synaptic network; only the initial network state (i.e. membrane voltages of all units) varied. This generated a total of 8,700 unique runs, which we then analyzed.

To ensure that our simulations truly demonstrate sustained asynchronous irregular activity as seen in neocortex, we measured the distribution of the coefficient of variation of the interspike intervals of all neurons in every completed network simulation (Fig 2C), and also averaged across units within each completed simulation (Fig 2D). In both instances we find that networks are in an irregular regime (CV > 1). Averaging across networks we find a mean CV of 1.23 ± .21 for excitatory neurons and 1.31 ± .23 for inhibitory neurons. Furthermore, we measured the L2 error between the autocovariance normalized by variance and the autocorrelation of activity (after rate was no longer increasing in each trial) in both excitatory and inhibitory populations. The low level of difference seen between the two, and the rapidly decaying autocovariance, are qualitatively consistent with a stationary process (Fig 2E) (Nielsen 2006). Previous work has defined the asynchronous irregular state by using coefficients of variation greater than 1 to define irregularity and by using stable firing rate to define asynchrony (Kumar et al. 2008a; Brunel 2000; Kumar et al. 2008b). Within this framework, our observations indicate our networks exhibit an asynchronous irregular regime with stationary activity.

2.3 Rate, branching scores, and synchrony values on sustained and truncated simulations

We found that the same topology was capable of producing both sustained and truncated activity when only initial membrane voltages were varied. To be clear, the optimal input units and

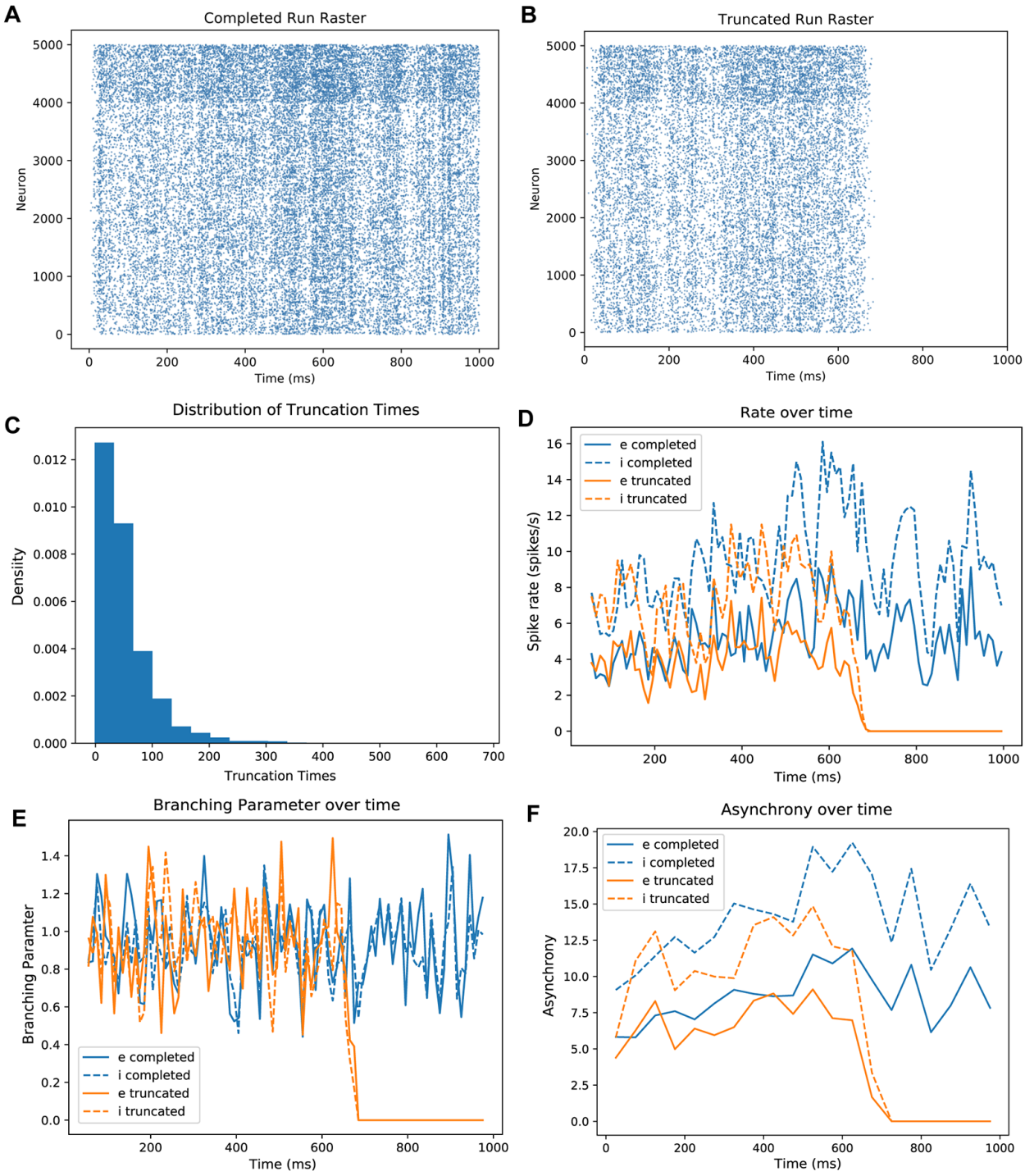

Figure 1.3: Simulations on the same network topologies yield sustained or truncated runs

A. A raster plot of a single complete 1000ms simulation on one of our networks. Excitatory units are numbered 1-4000 on the y axis, and inhibitory units are 4001-5000.

B. A raster plot of a single truncated simulation (700ms) on the same network with the same input.

C. Distribution of truncation times in ms for all truncated runs.

D. Instantaneous rate across time for the simulations in rasters A and B, binned at 10 ms. Blue: sustained run; orange: truncated run; solid line: excitatory units; dashed line: inhibitory units.
E. Instantaneous branching across time for the simulations in rasters A and B, binned at 10 ms. Same legend conventions as in D.
F. Instantaneous Van Rossum distance across time for the simulations in rasters A and B, binned at 50 ms. Same legend conventions as in D.

Poisson input trains were always the same but the membrane voltages varied. A run was sustained (or complete) if it displayed stable activity for the duration of a 1-second trial (Fig 3A). We found that all network simulations which reached 1 second were in each case able to sustain activity for the full duration of the simulation: we surveyed run times up to 10 seconds. We therefore chose one second as an indication of a network's ability to sustain activity indefinitely, and as the definition of a successful run. If a network ceased all spiking before reaching the 1-second mark, that simulation was considered truncated (Fig 3B). Scoring analysis of the network spiking dynamics of rate, branching and synchrony between sustained and truncated run types revealed substantial overlap regardless of outcome.

Duration of truncated runs followed a long-tailed distribution, with the majority of runs truncating early (Fig 3C). Since activity within the excitatory units tended towards fewer spikes as a run approached truncation, we did not include the final 50ms of truncating runs in the calculation of rate, branching, or asynchrony scores. We also did not consider the stimulus period (initial 30ms), as we wished to analyze self-sustained network dynamics rather than stimulus-driven spikes. By focusing our analyses on the 'middle portion' of each run, we found that the rate, branching and synchrony values within excitatory units of both sustained and truncated run populations overlapped substantially. Runs that truncated within the epoch spanning 500ms to 990ms and sustained runs shared similar mean excitatory firing rates (9.77

and 10.14 spikes/s, respectively; $p < .0005$, $n = 657$, two sample chi square test). Runs that truncated between 140 and 400 ms had a significantly higher mean rate (15.65 spikes/s; $p < 1 \cdot 10^{-15}$, $n = 1354$, two sample chi square test), suggesting that higher firing rates contribute to instability of a network (Vogels & Abbott 2005) (Fig 4A). The overlap index between rates of runs truncating later than 500 ms and sustained runs was 0.41 (95% CI 0.14 - .043). In runs that truncated earlier than 500 ms the overlap index with sustained runs was 0.27 (95% CI 0.23-0.30). Unlike rate, there was no difference in the scores between longer and shorter run times for both criticality and synchrony. The criticality score, measured using the branching parameter, was 0.997 ± 0.136 for truncated runs and 1.009 ± 0.004 for completed runs ($p < 1 \cdot 10^{-15}$, $n = 979$, two sample chi square test). The overlap index for criticality for the two run types was 0.31 (95% CI 0.21-0.35) (Fig 4B). Thus first order descriptors of rate and criticality within excitatory units, while different, substantially overlapped. This eliminated the possibility of a simple explanation of how and why activity was sustained in some cases while truncated in others.

As described above, only spikes from the middle portion of each run—that is, 30ms after trial start until 50ms prior to truncation—were used for the calculation of asynchrony scores. For the sake of computational efficiency during grid search, synchrony was defined as the variance of the mean voltage divided by the mean of voltage variances of all excitatory units (see Methods). Due to the dependence of this rapid measure upon voltages, and the fact that all voltages decay to resting potential after truncation, it will always yield high synchrony values for truncated runs. We therefore used Van Rossum spike distance for all excitatory units, normalized for run duration (see Methods) (van Rossum 2001; Houghton & Kreuz 2012), as our measure for each

simulation's asynchrony score outside of the initial grid search. As asynchrony increases, the Van Rossum distance also increases. The Van Rossum spike distance for our simulations was 3.82 ± 0.62 for truncated runs and 4.27 ± 0.51 for completed runs ($p < 1 \cdot 10^{-15}$, $n = 1033$, two sample chi square test). To provide some context, the corresponding Van Rossum spike distance of a

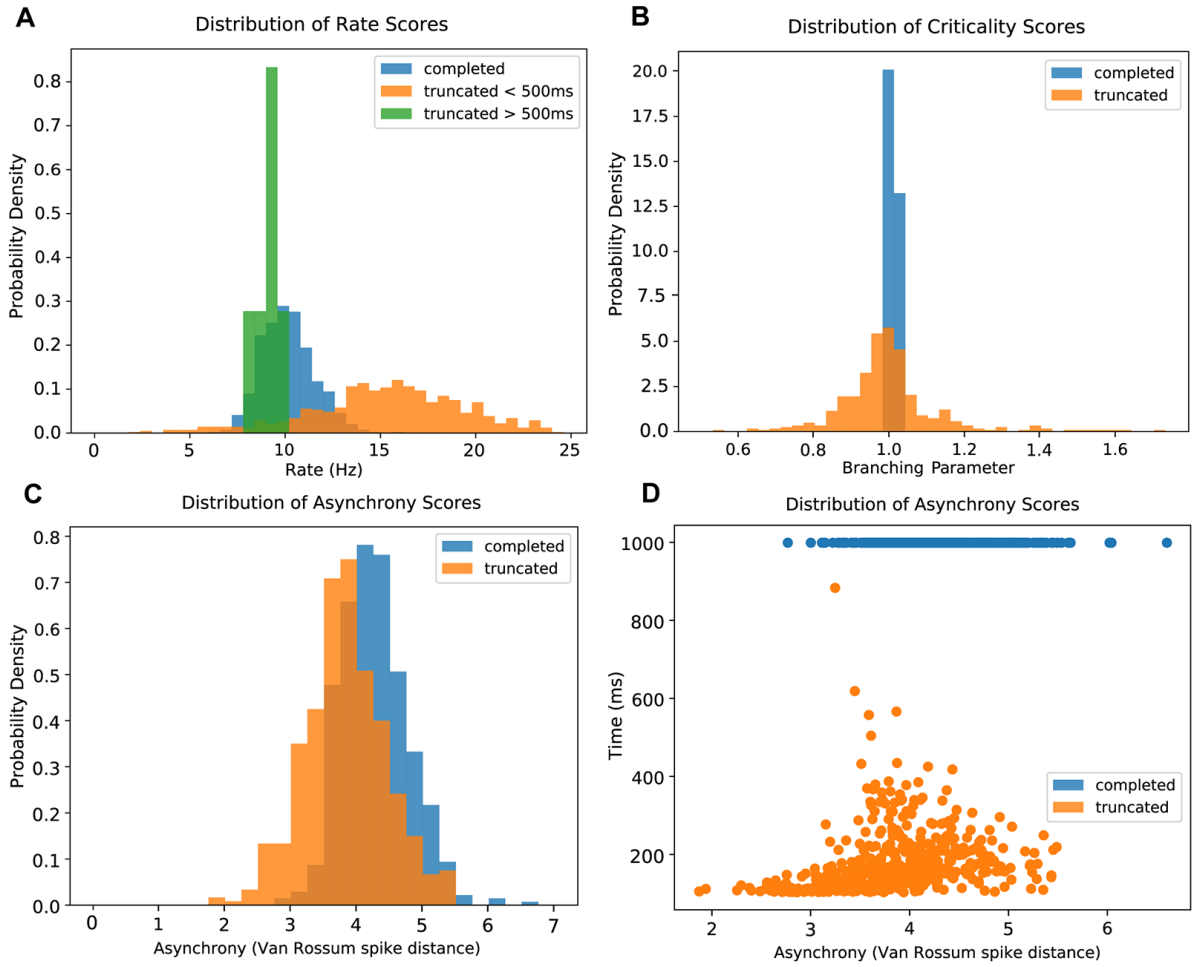

Figure 1.4: Score distributions for sustained and truncated runs

A. Distributions of spike rate scores for completed (blue), late-truncated (> 500 ms duration, green), and early-truncated (< 100 ms duration, orange) simulations.
B. Distributions of criticality scores (branching parameter) for completed (blue) and truncated (orange) simulations.
C. Distributions of asynchrony scores (Van Rossum spike distance) for completed and truncated simulations.
D. Same asynchrony score data as in C, with completed and truncated simulations now separated along the y axis by their total durations. Each dot indicates an individual simulation.

rate- and size-matched network of uncorrelated Poisson units would be approximately 9. The overlap index for truncated and completed runs was 0.68 (95% CI 0.61-0.73) (Fig 4C and 4D). The values of Van Rossum spike distance were, like rate and criticality, highly overlapping between truncated and sustained runs. Thus rate, criticality, and synchrony levels substantially overlapped and did not cleanly partition truncated and sustained run types.

2.4 Graph theory analysis of simulated networks

Having established that first order descriptions of network spiking failed to cleanly segment simulations into sustained and truncated runs, we next considered higher order descriptions. In previous work we have defined a taxonomy of active networks (Chambers & MacLean 2016). In that work we found that mechanistic insights underlying spiking network activity were provided by focusing on the subset of synaptic connections active during any one run. We refer to these networks as recruitment graphs and focus our analysis on these networks.

To do so, we begin with the structural connectivity matrix of our models as the synaptic graph (Fig 5A). We then constructed functional graphs using mutual information to quantify pairwise correlations between spiking neurons across each simulation. In order to generate a series of recruitment graphs, we identified the intersection of the functional graph with the synaptic subgraph according to the units which were active in each 10ms time step, resulting in one recruitment graph per time step. Weight values of functional and recruitment connections were calculated from mutual information and summarized in the functional graph, rather than taken from the synaptic weight matrix (see Methods). Due to our interest in the relationship between synaptic structure and functional spike maintenance, we focused our analysis on recruitment

graphs. Thus the following results, unless otherwise noted, describe actual synaptic connections which have functional significance because they are active.

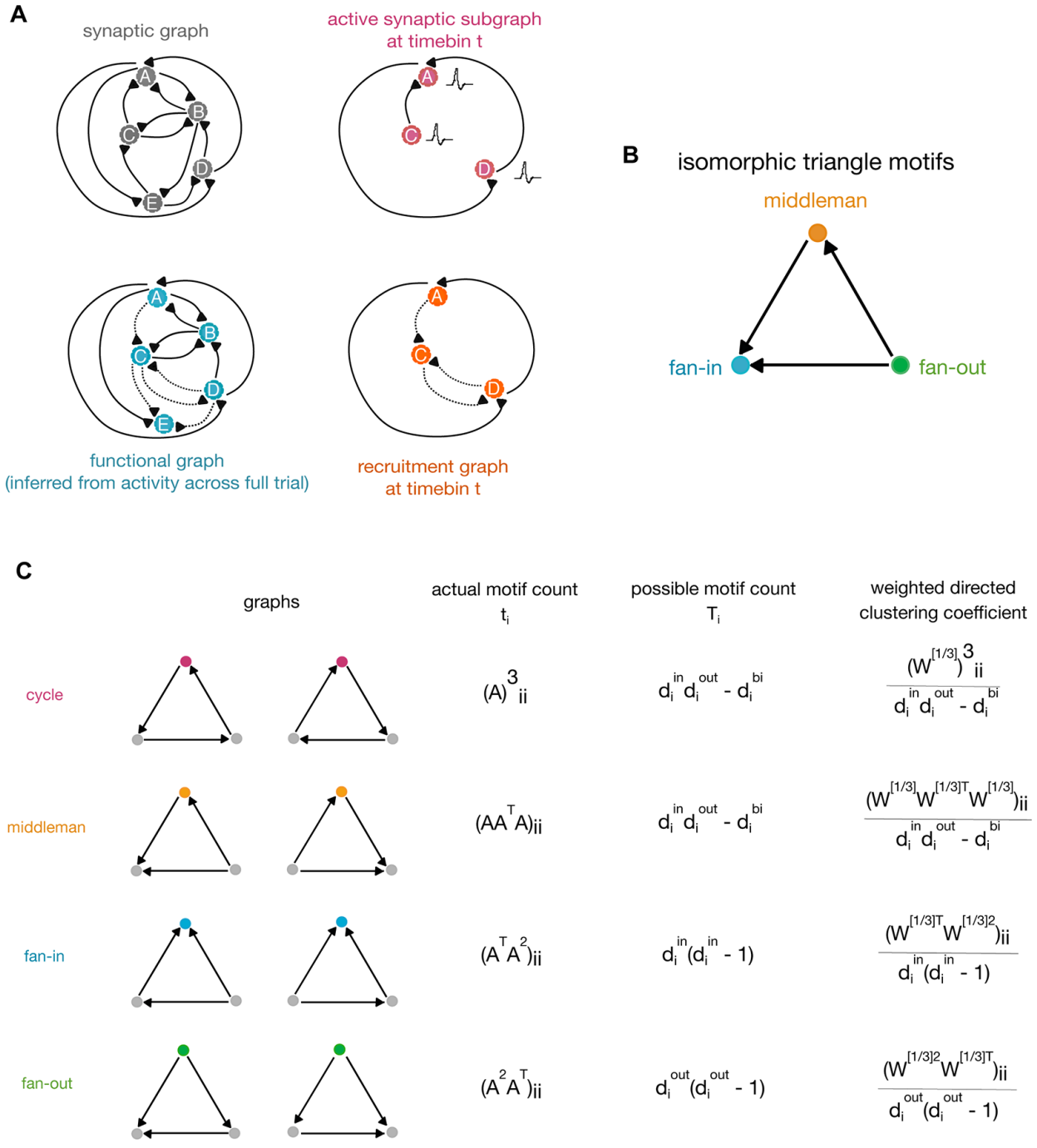

Figure 1.5: Graph and motif definitions

A. The synaptic graph is the ground-truth topology of our networks. Based on spiking activity during each simulation, we construct a series of active synaptic subgraphs—one for each time bin. These are graphs made of units which spiked in that bin, connected via the same edges as in the synaptic

graph. We infer a single functional graph from whole-trial spiking activity using confluent mutual information—these graphs represent the functional connectivity of the network for each simulation trial. The intersection of the functional graph with the active subgraph for a given time bin yields the recruitment graph for that time bin.

B. The three triangle motifs we examine—fan-in, fan-out, and middleman—are isomorphic by rotation. When calculating motif clustering, the choice of reference node is key.
C. Calculation of the clustering coefficients of the different triangle motifs on weighted directed graphs, as defined in Fagiolo 2007. The clustering coefficient is defined as the ratio of the actual to the possible motif counts.

2.5 Triplet motifs

The term 'motif' refers to a pattern formed by a group of units in a network. Previously we found that triplet motifs were informative of synaptic integration (Chambers & MacLean 2016) and also increased the power of in vivo encoding models (Dechery & MacLean 2018; Cayco-Gajic et al. 2015; Shi et al. 2015; Shahidi et al. 2019). Here we focused our analysis on similar patterns of connectivity in the recruitment network, involving groups of three units (Fagiolo 2007).

From the perspective of a single reference neuron, neighboring neurons can be arranged into four types of triplet motifs: fan-in, fan-out, middleman, and cycle. In isolating one triplet, the fan-in, fan-out, and middleman motifs are isomorphic by rotation, meaning that they only differ due to the choice of reference node (Fig 5B). The relative importance of a motif for a given neuron is measured by its contribution to that neuron's clustering coefficient (Fig 5C). The clustering coefficient is the weighted ratio of the actual over the possible counts of a particular triplet motif type in which that neuron participates. Individual reference nodes in a given triplet may yield different clustering coefficients due to their specific weights and connections (see Methods).

It was possible that each of the algorithmically generated networks had different connection densities and weight distributions, which would impact weighted motif clustering coefficient measures. A measure that incorporated weight and controlled for density would be especially relevant since the recruitment graph density evolves over time. Furthermore, comparison with density matches is important given that sparseness itself results in enhanced small-world clustering (Hlinka et al. 2012). We therefore used the measure of clustering propensity (Muldoon et al. 2016). Propensity is the ratio of the clustering coefficients of the recruitment graphs compared to the average clustering coefficients of graphs with the same connection structure but randomly assigned connection weights. The propensity measure allowed us to compare different networks despite different connection densities and also allowed us to assess the impact of specific edge weights on triplet motif clustering coefficients (Fagiolo 2007). A propensity value of 1 indicates that specific edge weights play a negligible role in clustering, since random edge weights would yield the same clustering coefficients (see Methods).

2.6 Density and reciprocity statistics

As reported above, synaptic networks were 21.1% connected, and 22.4% of connections were reciprocal. The functional networks of sustained runs, which were calculated using mutual information and were unique to each run, were more densely and also more reciprocally connected (Fig 6A, left). The functional networks averaged 32.6% (std: 0.6%) connectivity, of which 59.0% (std: 0.4%) were reciprocal. Recruitment graphs across time in sustained runs were sparser than the synaptic graphs, although only slightly less reciprocally connected (9.5% connected, std: 0.5%, and 16.7% recurrent, std: 0.5%) (Fig 6B, left). Functional and recruitment density and reciprocity did not differ significantly between sustained and truncated runs.

However, there were more limited ranges for and a tighter relationship between density and reciprocity of both functional and recruitment graphs of sustained runs. In contrast, the spread of density and reciprocity, and of their relation, was more diffuse in truncated runs (Fig 6A and 6B).

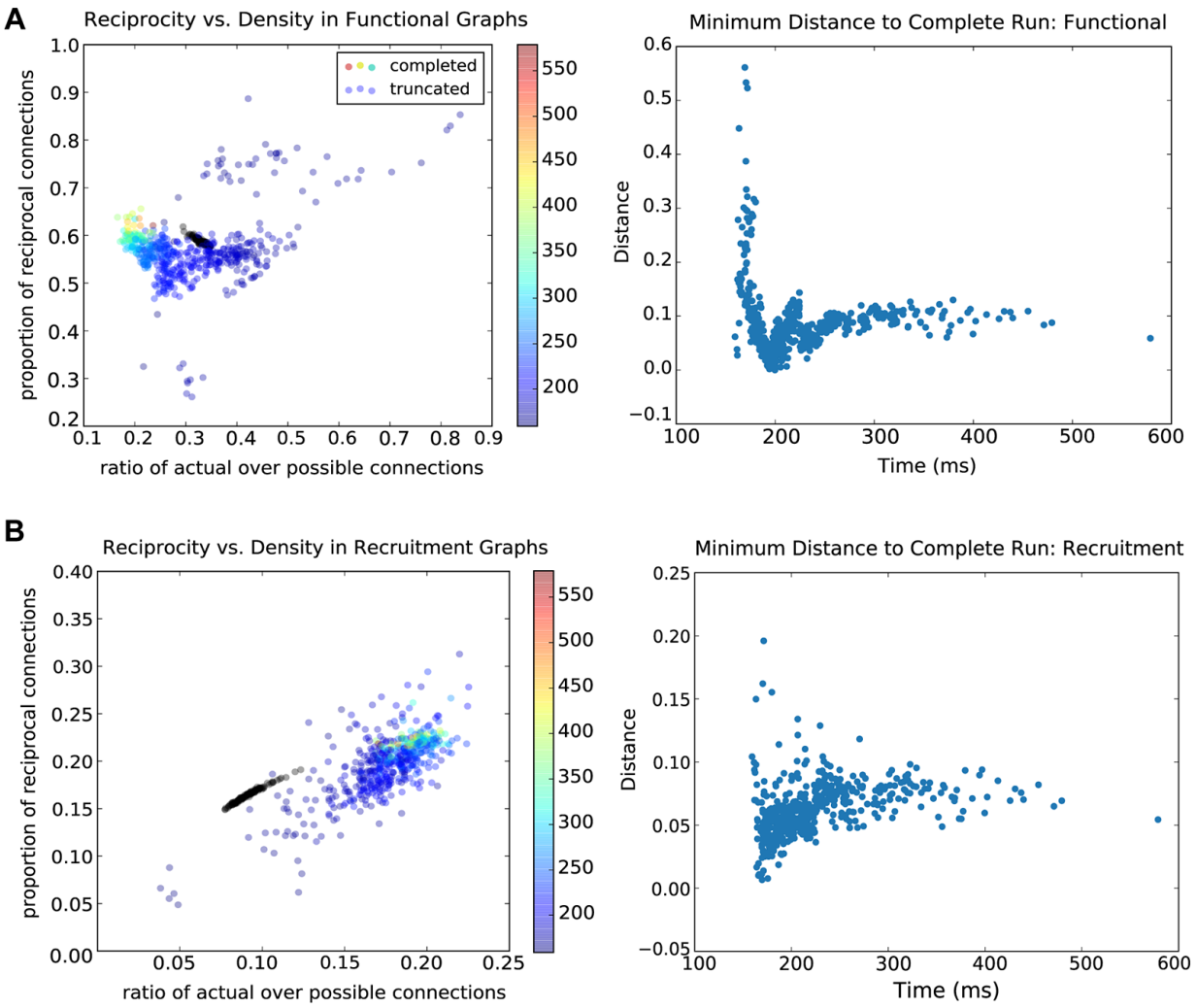

Figure 1.6: Standard network reciprocity

A. The left-hand panel shows the reciprocity (ratio of reciprocal connections to total connections) of **functional graphs** plotted as a function of their density (ratio of existing to possible connections). Data points for sustained runs are plotted in black and form a tight cluster, whereas those for truncated runs are varied. Truncated runs are colored by the ratio of inhibitory to excitatory spike rates. The right-hand panel shows the minimum distance (in reciprocity vs. density coordinate space for functional graphs) between each truncated run and a sustained run as a function of truncation time. Truncated runs which have greater than 200ms duration level off in their minimum distance. Thus, past a certain threshold, the difference between truncated and sustained runs' density and reciprocity is not related to the run duration.

B. The left-hand panel shows the reciprocity (ratio of reciprocal connections to total connections) of **recruitment graphs** plotted as a function of their density (ratio of existing to possible connections). Sustained runs are plotted in black and form a neat relationship between density and reciprocity and occur within a limited range of values. As in panel A, truncated runs are more diffuse. The color of each point indicates the ratio of inhibitory to excitatory spike rates. And also as in panel A, the right-hand panel shows the minimum distance between each truncated run and a sustained run (in reciprocity vs. density coordinate space) as a function of truncation time, this time for recruitment graphs. Truncated runs which have greater than 200ms duration level off in their minimum distance.

In addition to greater variance for truncated runs, there are differing trends in the variance, suggesting multiple modes of failure for a network simulation. The right-hand panels of 5A and 5B show, for functional and recruitment graphs respectively, the relationship between truncation time and the minimum distance between truncated and sustained runs, where distance is measured according to the 2D coordinate space of the left-hand panels (reciprocity vs. density). The minimum distance levels off for runs which exceed 200 ms in duration. Thus, the density and reciprocity measures of truncated runs do not approach those of sustained runs as duration increases. For runs which truncate prior to 200 ms, the minimum distances vary much more but are not strictly dependent on run duration. Thus the increased variance in truncated runs' density and reciprocity compared to those of sustained runs is not dependent on run duration. This is particularly the case at time points beyond 100—200 ms.

2.7 Triplet motifs in the different graph types

We found that the three isomorphic motifs showed equal clustering in the synaptic graphs. This is expected of graphs with random, albeit clustered, synaptic connectivity. Clustering propensity centered at 1.00 (std = $7.8 * 10^{-5}$, $7.9 * 10^{-5}$, $8.0 * 10^{-5}$ for middleman, fan-in, and fan-out) for all three isomorphic motifs (Fig 7A). A value of 1 indicates that specific edge weights in synaptic

graphs play a negligible role in clustering, since random edge weights would yield the same clustering coefficients. We found that in contrast to the static synaptic graph the dynamic

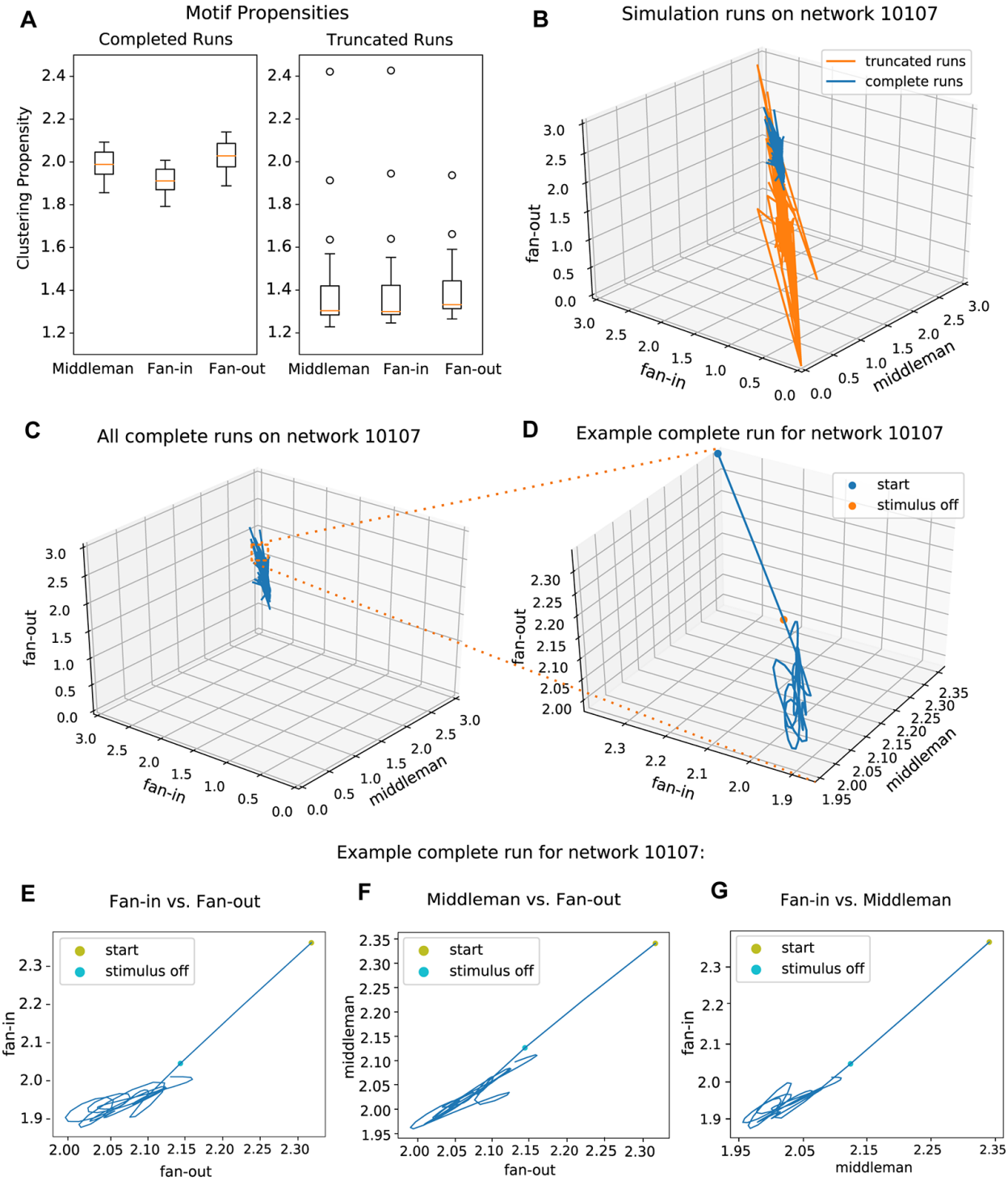

Figure 1.7: Standard network triplet motifs

A. Comparison of triangle motif clustering propensities of the three isomorphic motifs on sustained and truncated runs across all networks.
B. Trajectories of all runs on a sample network in 3-dimensional isomorphic motif space. Truncated runs have a larger spread of trajectories along with variation in the ratios of inhibitory to excitatory spike rates. However, sustained runs are consistent in their spike rate ratios.
C. Trajectories of all sustained runs alone, on axes of the identical scale as in panel B.
D. Example trajectory of a single run on the same network, now enlarged (from inset in panel C). The network begins away from the area of its eventual cyclic trajectory, and the 30ms of Poisson input at the beginning of the run drives it towards this region.
E. Example trajectory from panel D shown as fan-out propensity vs fan-in propensity.
F. Example trajectory from panel D shown as fan-out propensity vs middleman propensity.
G. Example trajectory from panel D shown as fan-in propensity vs middleman propensity.

functional and recruitment graphs were not random. The isomorphic motifs' dominance in the recruitment graphs, or the strength of each motif's contribution to overall clustering, varied over time in each trial. For sustained runs, motif clustering propensities for recruitment graphs (averaged across all time and all topologies) were 1.98 (std = 0.06), 1.91 (std = 0.06), and 2.03 (std = 0.07) for middleman, fan-in, and fan-out, respectively. Propensity values greater than 1, as these are, indicate that units in the recruitment graphs are more strongly clustered than would be expected in structurally-matched graphs with randomized weights. Motif clustering propensities also varied in recruitment graphs of truncated runs, with averages of 1.39 (std = 0.23), 1.40 (std = 0.23), and 1.43 (std = 0.24) for middleman, fan-in, and fan-out motifs (Fig 7A).

2.8 Cycling of triplet motifs

To evaluate how the three isomorphic motifs co-varied across time for both successful and truncated trials, we plotted motif clustering propensities at each point in time against one another. We visualized this for all runs from a sample synaptic network (Fig 7B), and then examined sustained runs in particular (Fig 7C and 7D). Additionally, we colored the trajectory of each run according to the ratio of inhibitory to excitatory spike rates (Fig 7C). Truncated runs varied in the

ratios of excitatory and inhibitory spike rates, consistent with the postulate that an imbalance between excitation and inhibition may have contributed to overall instability within the network. Sustained runs were consistent in their rate ratios. We further examined each two-dimensional projection of these motif transitions over time. 7E shows fan-in vs fan-out, 7F shows middleman vs fan-out, and 7G shows fan-in vs middleman for the complete run in (Fig 7D). As in the 3D case, clustering propensities formed a cyclic trajectory within a restricted region of 2D motif space. This indicates a systematic alternation between over- and under-representation of the three isomorphic motifs in the whole network relative to what would be expected in edge-matched networks. The cyclic trajectory within this region of motif space was consistent for all complete runs of low-rate, asynchronous, excitatory clustered networks we examined. We also found the same orderly temporal progression from one isomorphic motif to another when we considered clustering coefficient values as opposed to edge-normalized propensity. In contrast, truncated simulations were never restricted to this low variance cyclic alternation.

2.9 Motif cycling and sustained activity

The motif cycling trajectory was not present at the moment of first spikes in a simulation. Rather, the path started at a point in motif space as determined by the initial membrane voltages of all neurons in the network. Injection of Poisson input drove network activity towards its eventual trajectory (Fig 7D, 7E, 7F and 7G). We identified two distinct types of truncation—in the first and far more common (97.7%) of the two, the simulation trajectory never approached or entered the region in propensity motif space where sustained runs lay. Truncation occurred rapidly after input stimulus ceased. In the second, rarer case (2.3%), the simulation successfully entered the sustained regime, yet after several hundred ms the trajectory destabilized, resulting in truncated

activity. In this small subset of runs that exhibited trajectories prior to truncation, the trajectories did not follow a canonical path. Instead, motif dynamics during truncated runs transited in all directions away from the central region, demonstrating the multitude of ways in which activity structure can lose stability resulting in a failure of maintenance of an asynchronous spiking regime (Fig 7B). We examined whether the initial distance and input trajectory, which were determined by the initial conditions of the network and the Poisson stimulus, were determinants of successful activity maintenance. We found that even if the distance from the cycling region at the end of the stimulus period was minimal, some simulations still failed to enter into and stay within that regime. Others which were still distant from the region after the initial stimulus period continued on a successful trajectory and entered a stable cycling regime. These behaviors point to complex interactions between the network's internal state and how input onto precise units within that network can influence maintenance of asynchronous spiking.

2.10 Markov analysis

In order to quantify the cycling between isomorphic motifs, we constructed a Markov model for state transitions between dominant isomorphic motifs. We quantified cycling in this way because, while it does not give a sufficient account (the same markov chain could give different oscillatory behavior) of the observation, it does give us a necessary condition (a different markov chain could not generate the same observed pattern of oscillation). We described the network using a probabilistic voting scheme, as opposed to using analog propensity values. A unitary vote is cast by each unit for the motif type for which it has the highest propensity value at some time step. The proportion of total votes for each motif type is used to describe the relative dominance of that motif at that time step.

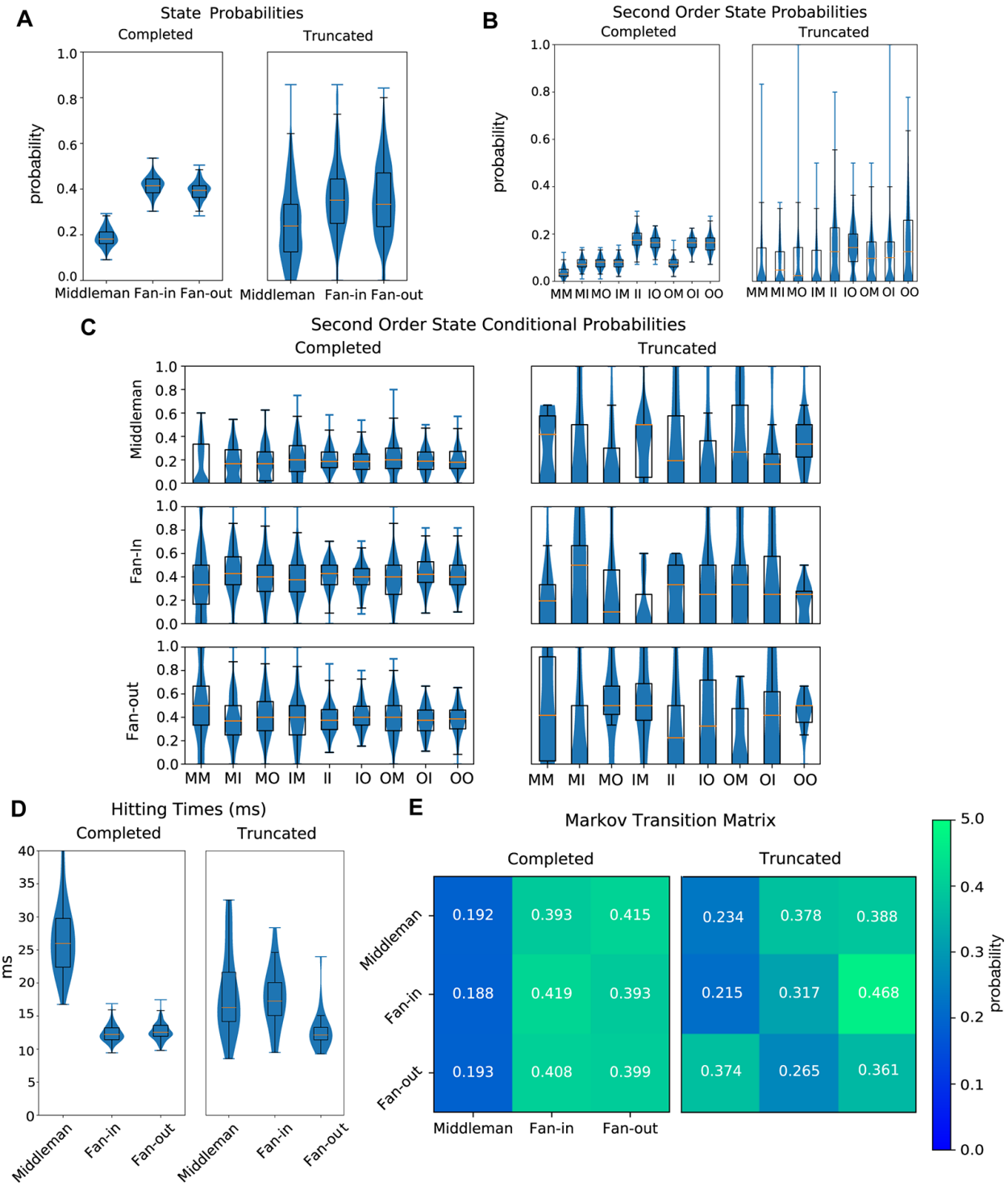

Figure 1.8: Markov comparisons between sustained and truncated runs on standard networks

A. Probabilities of state dominance of a triplet motif in sustained (left) and truncated (right) runs.
B. Second order state probabilities for sustained (left) and truncated (right) runs.
C. Second order conditional state probabilities for sustained (left) and truncated (right) runs.

D. Expectation of hitting time for Markov model of state dominance transitions in sustained (left) and truncated (right) runs.
E. Visualization of Markov matrix for state dominance in complete (left) and truncated (right) runs.

From this time series we constructed a Markov model transitioning between states. We found that the parameters characterizing the Markov process were canonical and low variance, such that successful cycling followed a specific reliable sequence between motifs. In contrast, the Markov parameters in simulations that truncated showed a failure to recruit this low-variance canonical sequence. First and second-order state probabilities and state transition probabilities significantly differed between sustained and truncated runs ($p < 1 \cdot 10^{-15}$, n = 1107, $p < 1 \cdot 10^{-15}$, n = 1107, $p < 1 \cdot 10^{-15}$, n = 884 respectively) (Fig 8A and 8B). Second-order state conditional probabilities also differed ($p \leq 0.029$, n = 227) (Fig 8C). State probability is the probability of a motif being dominant at a given time. Second order probability is the probability with which a sequence of two motifs will be dominant at some given time. Conditional probability is defined as the probability of a motif given history of previous two motifs.

Markov analysis also gave the time scale which characterized motif cycling via the mean time for recurrence. This is defined by the expectation of the hitting time for each motif, given the network is currently dominated by that motif. We define hitting time, t, as $H_i = inf\{n \geq 1 : S_n = i \mid S_o = i\}$ and our expectation of hitting time, t, as $E[t] = \sum_{n=0}^{\infty} n \cdot p(H_i = n)$. This gives a mean recurrence time for each motif. We find truncating middleman to have mean 18.59 ms (std: 6.49 ms), completing middleman to have mean 27.02 ms (std 5.99 ms), truncating fan-in to have mean 18.01 ms (std: 4.61 ms) completing fan-in to have mean 12.39 ms (std: 1.27 ms), truncating fan-out to have mean 12.95 ms (std: 3.14 ms), and

completing fan-out to have mean: 12.65 ms (std: 3.56 ms). Hitting times differed significantly between sustained and the small subset of truncated runs that entered this region of propensity ($p \leq 2.70 \cdot 10^{-6}$, n = 227) (Fig 8D).

2.11 Effects of connectivity weights

We hypothesized that the cycling between clustering propensities was necessary for sustained asynchronous activity due to the weak strength of the majority of individual synapses. Fan-in clustering has the highest probability of remaining in the state of fan-in clustering in the next time point which hints at the greater need for integration. But once integration is sufficient, the motif changes. For our model and for most of the synapses in neocortex, convergence of spikes from multiple sources must occur in order to evoke spikes in a receiving neuron (Chambers & MacLean 2016). Consequently we expected that as connection weights increased, the cycling between population-wide isomorphic motifs would lessen.

To test this, we strengthened all synaptic weights in the networks that previously scored well from 1.0x to 2.0x original values in increments of 0.1. Simulations were then re-run on these strengthened networks using the same stimulus and initial conditions. At 1.6 times the original weights, networks consistently displayed bursting activity. Consequently we restricted our analysis to networks with weights increased 1.5 times. All runs on these networks reached completion.

Increasing weights led to a decrease in all triangle motif propensities (Fig 9B), and also led to differences in the Markov characterization (Fig 9H). Motif state probabilities differed

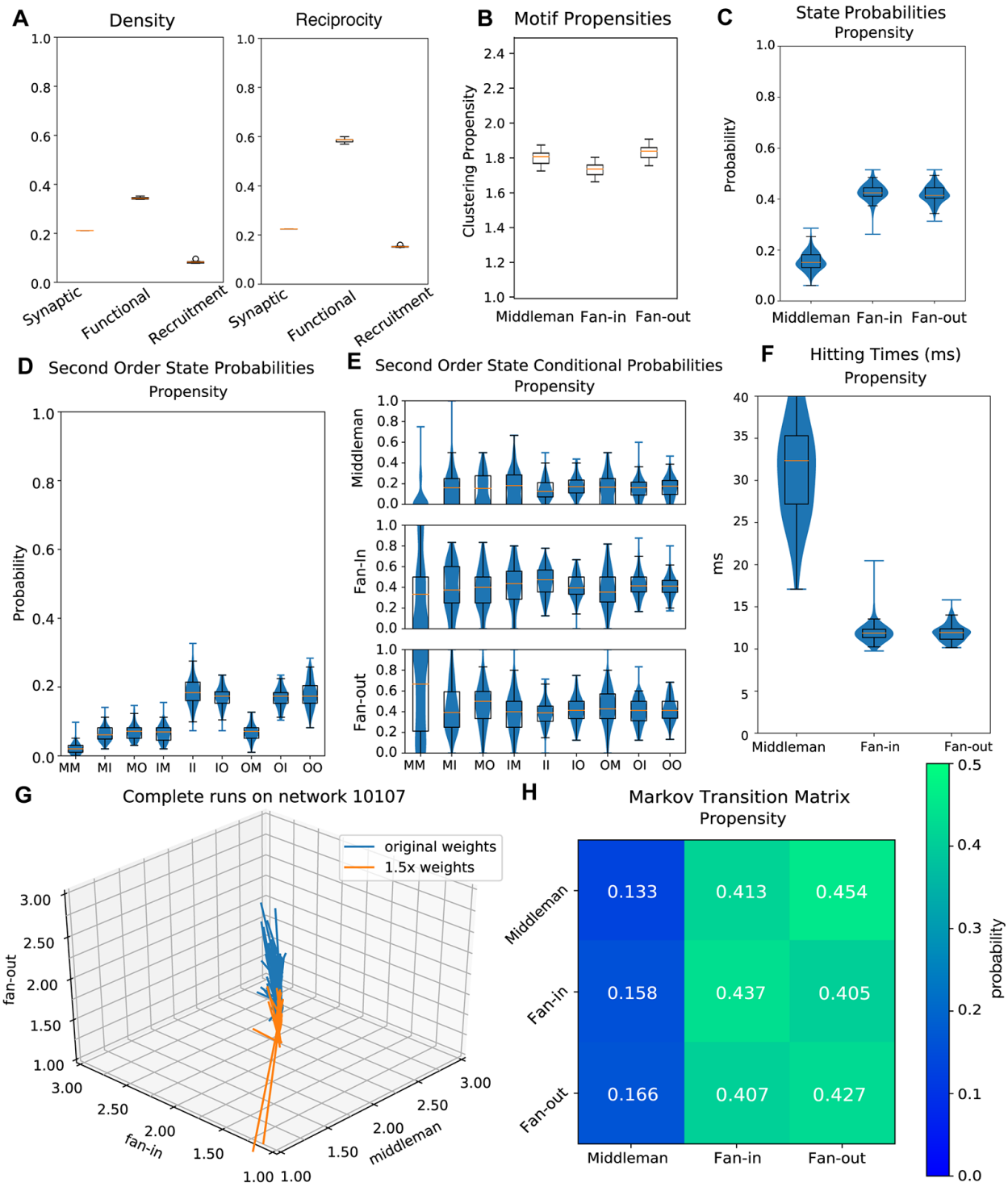

Figure 1.9: Networks with increased weights

Networks have the same structure as those seen in Figs 4 and 5, but all edge weights have been increased by 1.5 times their original values.

A. Left, density (ratio of existing to possible connections) for synaptic, functional, and recruitment graphs. Right, reciprocity (ratio of reciprocal to all existing connections) for synaptic, functional, and recruitment graphs.
B. Clustering propensity for isomorphic triangle motifs on increased-weight-graph simulations. The y-axis is scaled to match that of Fig 7A (clustering propensities on original graphs) and Fig 10B (clustering propensities on unclustered ER graphs).
C. Probabilities of dominance of each triangle motif. The dominant motif at a time point is given by the maximum of mean middleman, mean fan-in, and mean fan-out across units.
D. Second order motif state probabilities for progression of temporal recruitment graphs.
E. Probabilities for each motif to follow a given second order motif.
F. Hitting times for each state for the Markov process defined by motif transition probabilities.
G. Trajectories of all complete runs on a sample network in 3-dimensional isomorphic motif space. In blue are the runs on the network with its original weights, in orange are the runs on the same network with weights increased.
H. Markov Matrix for transition probabilities between motifs.

significantly between sustained runs on the original graphs and those on graphs with increased weights ($p \leq 0.00032$, $n = 296$) (Fig 9C). Second-order state probabilities, state conditional probabilities, hitting times, and state transition probabilities all differed significantly as well ($p < 0.01$, $p < 0.05$, $p < 0.005$, $p < 0.001$, Fig 9D, 9E, 9F and 9H) ($p \leq 0.0065$, $n = 296$, $p \leq .015$, $n = 267$, $p \leq 0.0014$, $n = 267$, $p \leq 0.011$, $n = 296$ respectively), demonstrating the interaction of synaptic strength on the necessity of this regime (Fig 9). However the trend remained and in all sustained runs a low variance transition from motif to motif occurred.

2.12 The dynamical motif solution is arrived at regardless of synaptic connectivity statistics

The networks on which we performed all our analyses have excitatory clusters of units. To test whether our results, including the motif cycling phenomenon, are dependent on this structure, we next examined non-clustered Erdős-Renyi (ER) graphs with $p_{i \to e} = 0.25$ and $p_{e \to i} = 0.35$. ER graphs had the same $p_{e \to e}$ and $p_{i \to i}$ values as the clustered networks. We found that transitions between motif types were also present in the activity of sustained runs on ER networks (Fig 10).

The relative increase in clustering in ER graphs when comparing synaptic to recruitment graphs is substantially greater than seen in our graphs with excitatory synaptic clusters. In the synaptic

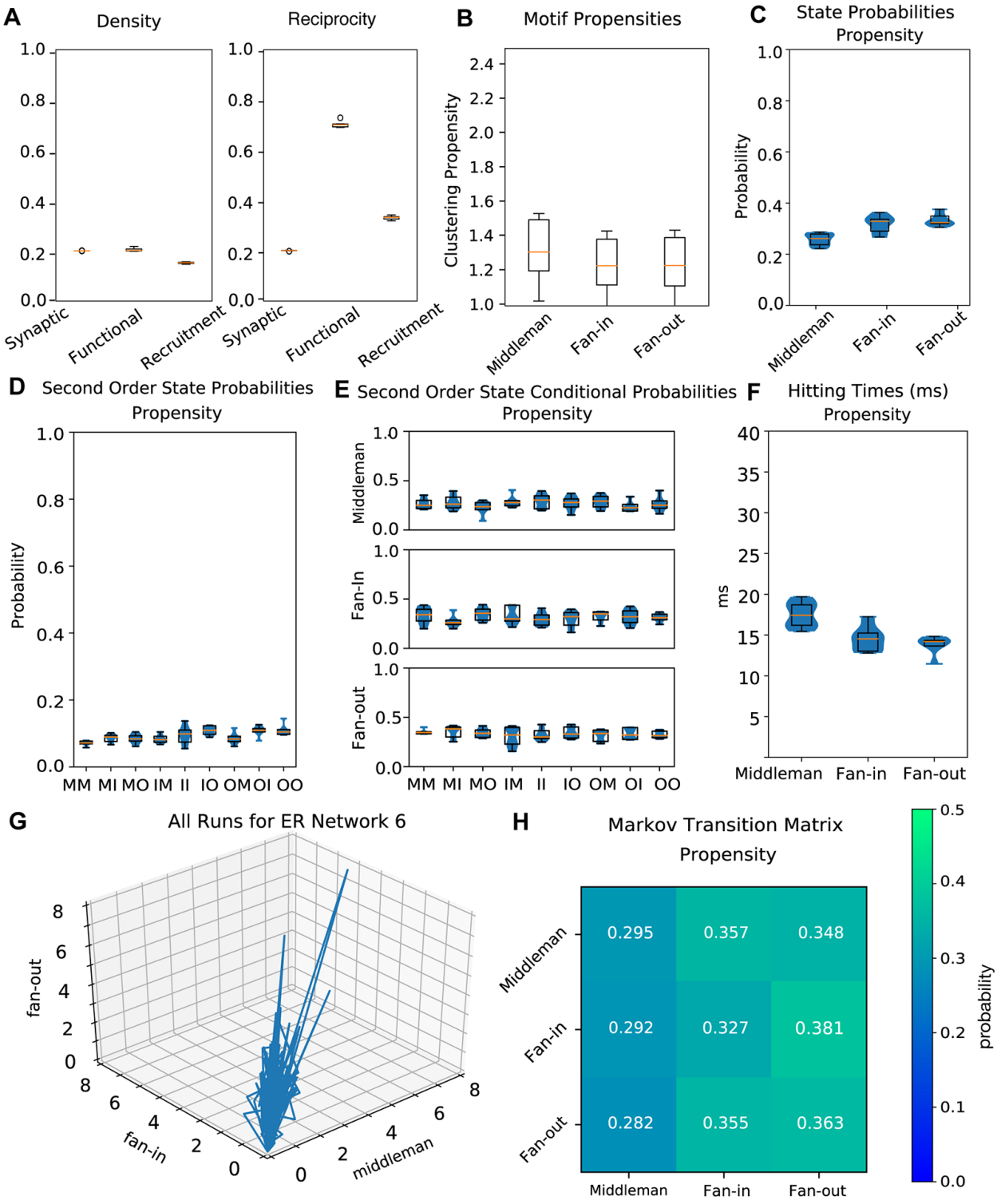

Figure 1.10: Unclustered (Erdős-Renyi) networks.

A. Left, density (ratio of existing to possible connections) for synaptic, functional, and recruitment ER graphs. Right, reciprocity (ratio of reciprocal to all existing connections) for synaptic, functional, and recruitment ER graphs.
B. Clustering propensity for isomorphic triangle motifs on ER graph simulations. The y-axis is scaled to match that of Fig 7A (clustering propensities on original graphs) and Fig 9B (clustering propensities on graphs with 1.5 times increased weights).
C. Probabilities of dominance of each triangle motif. The dominant motif at a time point is given by the maximum of mean middleman, mean fan-in, and mean fan-out across units.
D. Second order motif state probabilities for progression of temporal recruitment graphs.
E. Probabilities for each motif to follow a given second order motif.
F. Hitting times for each state for the Markov process defined by motif transition probabilities.
G. Trajectories of all runs on a sample ER network in 3-dimensional isomorphic motif space. All runs reached completion.
H. Markov Matrix for transition probabilities between motifs.

networks, triplet clustering coefficients average 0.11. However, this value increased to 0.20, 0.09, and 0.15 for fan-in, fan-out, and middleman motifs in the recruitment graphs. The propensity values for all isomorphic motifs were consistently lower than those in 1.5x networks, as well as original networks, with means centered at 1.25 (Fig 10B). We find that unclustered graphs and clustered graphs differ significantly in first and second-order state probabilities, state conditional probabilities, hitting times, and state transition probabilities (Fig 10C, 10D, 10E, 10F and 10H) ($p \leq 5.1 \cdot 10{-}12$, n = 111, $p \leq 0.00021$, n = 111, $p \leq 0.00017$, n = 106, $p \leq 1.1 \cdot 10^{-7}$, n = 106, $p \leq 6.2 \cdot 10^{-8}$, n = 111 respectively). As in the case with the increased weights however the qualitative cycling of motifs was present in sustained runs.

3 DISCUSSION

This work demonstrates that higher-order structure is crucial for sustained low-rate and asynchronous spiking in recurrent networks such as neocortex. Within the range of dynamics surveyed, we found that rate, criticality, and synchrony overlapped substantially between sustained and truncated runs, although high firing rates corresponded with short run times

consistent with previous work (Vogels & Abbott 2005). Thus first order statistics did not cleanly partition the stability of asynchronous spiking in our models. Our subsequent analyses of higher order structure revealed that there are many ways for network activity to 'fail' and only one specific way to 'succeed'. To succeed, spikes must traverse the synaptic network in a coordinated way, cycling iteratively between the global dominance of three triplet motifs. The transitions between fan-in, middleman, and fan-out motifs reveal the necessity of balance between distribution of output and convergence of input. The presence of these motifs in the recruitment graphs demonstrates the functional routing of activity through synaptic connections. When synapses become stronger and more reliable, overall triplet clustering decreases while the reliability of their transitions remains, demonstrating that these motifs tightly control synaptic cooperativity. Their presence is able to compensate for a prevalence of weak synaptic connections and to maintain the asynchronous spiking regime. Higher order motifs in the recruitment network thus provide a direct link between asynchronous spiking and the stability of activity in that network.

Our study exclusively examined networks which produced low-rate, critical, and asynchronous spiking, consistent with activity recorded in awake state in neocortex. As such, the cyclic motif transitions which support stability in this regime may not generalize to regimes with bursting or synchronous activity. The dynamical features of neocortex are undoubtedly interrelated with the stability of those dynamics. However, given the prevalence of this spiking regime, our results provide an explanation for the prominence of higher order motifs in real data. Elevated motif counts have been observed in synaptic connectivity and in recordings of clustered activity in vivo (Perin et al. 2011; Song et al. 2005; Litwin-Kumar & Doiron 2012; Sadovsky & MacLean 2014;

Rothschild et al. 2010; Pajevic & Plenz 2009; Orlandi et al. 2013; Shimono & Beggs 2015; Nigam et al. 2016; Dechery & MacLean 2018). Through mechanisms of learning in neocortex such as STDP, functional patterns may be further strengthened to enhance integration in cortex. We wish to draw attention to the fact that our study focused on the whole-network scale. Individual units spiked only sparsely, making it difficult to continuously track single-unit motifs across small epochs of time since interspike intervals were generally larger than the intervals we analyzed. Regardless, functional networks summarizing long recordings from neocortex also report the prevalence of these motifs, albeit without this dynamic component. The models we used were constructed to simulate neocortex. The network structures we employed closely match experimental observations (Perin et al. 2011; Song et al. 2005) and the model units capture many of the statistics of neocortical neurons (Zylberberg et al. 2017). Our results provide, first and foremost, an account of at least one of the roles of beyond-pairwise interactions in the brain. Yet the behavior of these models may reflect necessary features of weakly-connected networks in which integration from multiple sources is necessary for the system to succeed. In such systems it is likely that stability relies on higher-order patterns. For example, the spread of rumours in a social network relies on integrating interactions. Social networks are small-world networks characterized by clusters, a feature which is present in our model as well as many other systems (Barzel & Barabasi 2013; Milgram 1967). The “illusion-of-truth” effect in rumour spreading on a social network has the integrate-and-fire property, where an individual may need to hear a rumour from multiple sources before they reach a confidence threshold to repeat it to others (Moons et al. 2009).

The necessity of higher-order patterns for stable asynchronous activity has strong implications for neural coding. Previous work has already demonstrated that correlations enhance coding, with triplet correlations having an advantage over pairwise, as well as the limited role of motifs larger than three nodes (Zylberberg et al. 2017; Shimono & Beggs 2015; Dechery & MacLean 2018; Cayco-Gajic et al. 2015; Shi et al. 2015; Shahidi et al. 2019; Hu et al. 2018; Hu et al. 2013; Ocker et al. 2017; Curto et al. 2019). The neural code must rest upon a foundation of the maintenance of spiking, which we have shown in turn rests on higher-order motifs and coordinated synaptic integration in the awake dynamical state. Any two spikes must take place within some time interval for them to interact. The asynchronous and critical regime observed in vivo and in our models pushes the limits on what constitutes a cooperative event. In our model, the precise conditions for integration are dictated by the time constants we chose, while in neocortex the same time constants may vary and span some range. Neuromodulation, cognitive state, and a variety of other factors all dictate the requirements which need to be met for integration. Local connectivity certainly plays a large role as well. Consequently, the role of higher order interactions in coding and in coordinating synaptic integration may vary by brain region and state.

4 METHODS

4.1 Network structure

Our graphs are recurrent and sparsely connected networks of several thousand adaptive exponential leaky integrate-and-fire (AdEx) units with an extra poisson input term (Brette & Gerstner 2005). Synapses between all units are conductance-based. This enhances realism by

taking neuron-specific state features into account during synaptic integration (Brette & Gerstner 2005). Specifically we define our neuron Voltage, V, as

$$C\frac{dV}{dt} = -g_l\Delta_t exp\left(\frac{V-V_T}{\Delta_T}\right) - g_e(V - E_e) - g_i(V - E_i) - g_p(V - E_e) - w$$

adaptation current, w, as

$$\tau_w \frac{dw}{dt} = a(V - E_l) - w$$

excitatory conductance, g_e, as

$$\tau_e \frac{dg_e}{dt} = -g_e$$

inhibitory conductance, g_i, as

$$\tau_i \frac{dg_i}{dt} = -g_i$$

Poisson input conductance, g_p, as

$$\tau_p \frac{dg_p}{dt} = -g_p$$

A spike was said to occur if $V > V_t$, after which V was set to EL, w was incremented by b and g_e and g_i were incremented by synapse weight if downstream of the spiking neuron.

For information on parameters, see S1 Table. Each network is comprised of 1000 inhibitory and 4000 excitatory units. Precise wiring probabilities between excitatory and inhibitory populations were determined through grid search within biological constraints.

Network synaptic connectivity is heterogeneously clustered (Litwin-Kumar & Doiron 2012). For each network we defined 50 total clusters, with each excitatory unit randomly assigned to two clusters. Clusters thus vary in size and follow a binomial distribution. The wiring probability

between two units within the same cluster is twice that of units in different clusters. Network cluster sizes range from 111 to 207 excitatory units (mean = 158.40, std = 12.27). Inhibitory units are not clustered; their wiring probability is uniform across the graph.

Edge weights follow a heavy-tailed distribution (Fig 1B). Edge weights that originate from inhibitory units have conductances which are ten times greater than those which originate from excitatory units, in accordance with experimental results (Vogels & Abbott 2005).

4.2 Network simulation

Each simulation was recorded at 0.1-ms temporal resolution. A trial began with 30 ms of Poisson input stimulus onto 500 randomly chosen units. After 30 ms the stimulus would cease and activity would propagate naturally through the network. The simulation would continue for as long as spiking activity is sustained, up to a maximum of 1 second. If during a simulation no spikes occur across the network for 100 ms, the network is deemed inactive and the simulation trial is halted. We found that all network simulations which reached 1 second were also able to sustain activity up to 10 seconds. We therefore chose one second as the marker for a network's ability to sustain activity indefinitely, and as the definition of a successful run. Upon completion each simulation yields an output raster of spike times for every unit in the network. The Poisson input train, input units, network topology, and initial conditions of all units were recorded for each simulation. This enabled subsequent analyses and also allowed for re-use of a synaptic graph or re-instantiation of a simulation using some of the original settings while varying others.

4.3 Parameterization

Our models are constructed to parallel the features of biological neural networks, but are also constrained by considerations of computing resources. In a study which modeled cortex with high biophysical and anatomical detail, simplifying the neuron model did not lead to drastic differences in the network's behavior from the detailed model or from in vivo results. Most qualities remained unchanged, suggesting that in many cases extremely granular models are not necessary to yield experimental insights (Brette & Gerstner 2005). Instead, the most important feature for retaining qualitative correspondence are the rules of synaptic connectivity. Therefore we required our models' connectivity parameters to closely match those of biological neural networks.

The probabilities of wiring between excitatory (E) and inhibitory (I) populations in our models were taken directly from or bounded by the results of biological experiments. The wiring probabilities from E to other E units and from I to other I units in neocortex are well-studied, but there is less data on connections from E to I and from I to E. We therefore used an algorithmic approach to find the optimal values. Beginning within a biological range, we used grid search to find values of $p_{e\rightarrow i}$ and $p_{i\rightarrow e}$ that led to successful maintenance of activity at the lowest possible rates. We used these optimal wiring rules to construct all synaptic graphs in this study.

Two iterations of grid search were used to find the wiring parameters needed to maintain naturalistic spiking for the duration of a simulation (Fig 2A). We searched for the optimal probability of connection from excitatory to inhibitory units, $p_{e\rightarrow i}$, and the optimal probability of connection from inhibitory to excitatory units, pi→e, such that networks would sustain activity at the lowest possible rates. In the first iteration, we used a low resolution grid (space size 0.001) to

search for $p_{e\to i}$ within the range 0.16 to 0.24 and $p_{i\to e}$ within the range 0.29 to 0.37. These two ranges were taken from known wiring probabilities in neocortex. Each grid space was visited ten times to achieve an average measure of rate and completion. This isolated a region of interest where the rate was lowest, between $p_{e\to i}$ values of 0.210 and 0.230, and between $p_{i\to e}$ values of 0.300 and 0.320. We used a higher resolution grid (space size 0.0001) to explore this region.

For all subsequent simulations we used the best results obtained from grid search. The optimal probability of wiring for excitatory to inhibitory units, $p_{e\to i}$, was found to be 0.22, and the optimal value for $p_{i\to e}$ was 0.31. The values for $p_{e\to e}$ and $p_{i\to i}$ were taken from known wiring probabilities in neocortex, and were 0.20 and 0.30 respectively (Chambers & MacLean 2016). Based on these wiring rules, we constructed synaptic graphs of networks for simulations. Each synaptic graph is a matrix W where the value in w_{ij} denotes the weight of the directed connection from unit i to unit j.

4.4 Scores

To evaluate the biological realism of constructed networks, we computed several measures of network activity for both excitatory and inhibitory subpopulations. All measures were calculated based on spiking activity between 30ms after trial start and 50ms prior to trial truncation. Networks were evaluated on rate, defined as average spike frequency over the course of each trial. Networks were also evaluated on branching parameter as a measure of network criticality (Beggs & Plenz 2003). A branching value of 1 indicates that for every ‘ancestor’ unit that is active, there is an equal number of ‘descendant’ units active at the next time step. On average, the number of units active over the course of a trial in a critical network stays constant.

Branching was mathematically defined as:

$$\sigma = \sum_{d=0}^{n_{max}} d \cdot p(d)$$

$$p(d) = \sum_{avalanches} \left(\frac{n_{\Sigma a|d}}{\Sigma n_a}\right)\left(\frac{n_{max}-1}{n_{max}-n_a}\right)$$

where σ is the branching parameter, d is the number of descendants, n_{max} is the maximum number of active neurons, n_a is the number of ancestor neurons, n_d is the number of descendant neurons, $n_{\Sigma a|d}$ is the number of ancestor neurons in all avalanche events that involved d descendants, and $n_{\Sigma a}$ is the total number of neurons involved in avalanches. The branching parameter describes the network as a whole; it cannot be calculated for isolated units. For a given simulation, we calculated network branching at discrete, sequential time steps throughout. We used the same temporal resolution (5 ms) as used for determining the functional graph; all spikes at time t are ancestors, and all spikes from t + 5 to t + 20 ms are descendants. We then averaged the network branching parameter across all time steps to get the overall branching score for that simulation. Networks were further evaluated on their level of asynchrony, since biological networks display asynchronous activity. In order to evaluate asynchrony rapidly enough to make grid search feasible, a heuristic for synchrony was computed as the variance of mean voltage normalized by the mean variance of each neuron. An upper threshold of 0.5 was considered appropriate for network asynchrony. The threshold was evaluated empirically by examining a population of inhomogeneous Poisson neurons with underlying Gaussian firing rates where covariance across the underlying Gaussian processes was the varied parameter. The Van Rossum spike distance (van Rossum 2001) was used as the measure of asynchrony for all analyses outside of initial grid search. The Van Rossum spike distance was calculated as follows: each spike train was

convolved with an exponential kernel with time constant τ = 10 ms, we then took the distance to be the mean L2 norm between the resulting traces normalized by $\sqrt{\frac{1}{\tau}}$.

4.5 Triplet motifs

The clustering coefficients for the four triplet motifs are calculated in the following manner (Fagiolo 2007).

Let t_i denote the actual number of triplets of a motif type in the neighborhood of unit i, and T_i denote the maximum number of such triplets that unit i could form. We will build intuition by beginning with the case of a binary directed graph, or an unweighted connectivity matrix. Let A represent this graph, with a_{ij} = 1 indicating the presence of a directed connection from node i to node j. Raising the matrix A to the nth power yields the number of paths of length n between nodes i and j.

Let us first consider the cycle motif; in order for unit i to participate in a cycle, it must have an edge directed to a second unit, that second unit must have an edge directed to a third unit, and that third unit must have an edge pointing back to unit i. The path length is 3, and it both begins and ends at unit i. Thus we calculate A^3 and extract the values along the diagonal, or A^3_{ii}. This gives the number of actual cycle motifs unit i forms.

Counts of the three isomorphic motifs are calculated in a similar way, but they require the additional involvement of A^T. Taking the transpose of graph A reverses the directionality, so that connections from i to j are now those from j to i. We would like to trace a path of length 3 from i

back to i to form an isomorphic triangle, but exactly one of the steps must be against the true direction of that edge (Fig 5). Beginning with a middleman reference node, the first step is 'with the flow', the second step is invariably 'against the flow', and the final step back to i is again 'with the flow'. Therefore AA^TA_{ii} gives the number of actual middleman motifs unit i forms.

Since fan-in and fan-out motifs are isomorphic to middleman by rotation, we simply rotate which step is 'against the flow' to yield the count of fan-in and fan-out motifs. The number of actual fan-in motifs unit i forms is $A^TA^2_{ii}$, and the number of fan-out motifs is $A^2A^T_{ii}$.

Now that we can calculate the actual counts, the possible counts of each motif T_i are easily intuited as a combinatorics problem. Let us begin again with the cycle motif. To form a cycle, node i requires one edge directed towards it and one edge directed away from it. The number of possible pairs of in and out edges from node i is calculated by multiplying the out-degree of node i with the in-degree of node i. In-degree and out-degree refer simply to the number of edges that are directed in or out of a given node. Some edges may be bidirectional—these cannot be part of a true cycle motif. The number of bidirectional edges is subtracted from the product of in- and out-degrees. The final T_i for the cycle motif is

$$T_i = d_i^{in} d_i^{out} - d_i^{\leftrightarrow}$$

The Ti for middleman is in fact equal to that for cycle, since forming a middleman has the same requirements—one edge directed inward paired with one edge directed outward.

A fan-in motif requires two edges directed in towards the reference node. There are d_i^{in} number of choices for the first inward edge. Once that choice has been made, there are $d_i^{in} - 1$ choices remaining for the second inward edge. Thus we multiply the two to yield T_i for the fan-in motif.

$$T_i = d_i^{in}(d_i^{in} - 1)$$

Fan-out is similar—we simply substitute in-degrees with out-degrees since a fan-out motif requires two edges directed out from the reference node. T_i for the fan-out motif is thus

$$T_i = d_i^{out}(d_i^{out} - 1)$$

Now that we have both the actual and possible counts for each motif type, the triplet clustering coefficients of node i are simply their ratios. That is,

$$C_i^* = \frac{t_i^*}{T_i^*}$$

If we were interested in binary graphs, we would end here. However, our graphs of interest have weights associated with each directed edge. There are multiple ways to account for edge weights when calculating clustering coefficients. One way is to consider only the weights of the two edges that are incident to reference node i. Alternatively, the weights of all three edges in a triplet can be taken into consideration. The latter is the chosen method, since we desire a measure of central tendency. The total contribution of a triplet to the clustering coefficient is thus the geometric mean of its weights.

Let W denote our weighted directed graph. For a triplet in this graph with edge weights w_{ij}, w_{ih}, and w_{jh}, the geometric mean is $(w_{ij} \cdot w_{ih} \cdot w_{jh})^{\frac{1}{3}}$. We can extend this to the entire graph by,

Instead of using a binary graph as matrix A in the calculation of t_i, using $A = W^{\frac{1}{3}}$, which is the matrix that results from taking the cubic root of every entry in W. We also note that this formulation is invariant to the choice of reference node in a triplet. Incorporating weights only modifies the value of t_i. It remains a measure of the actual triplets present—instead of counts it is now a weighted measure. The denominator T_i still refers to maximum possible counts. It follows that the clustering coefficient for node i can only be 1 (maximum) when its neighborhood truly contains all triplets that could possibly be formed and every edge in each triplet is at unit (maximum) weight. The complete clustering coefficient formulas of weighted directed graphs are given in Fig 5.

4.6 Active subgraphs

For any small span of time in a trial, only a subset of all units in the graph will be active. The subset of units which spike in some defined time window form the active subgraph for that time window. We binned spikes into a temporal resolution of 10 ms, so that each complete 1-second simulation resulted in 99 time bins. For each time bin t we defined an active subgraph. If a unit spiked within time bin t, that unit will be part of the active subgraph for time bin t. All units which did not spike within that particular time bin are not included in that particular active subgraph. Since there are 99 time bins for a complete 1-second simulation, there are also 99 active subgraphs in sequence. Edge weights between units in an active subgraph are equal to those from the corresponding edges (between active units) in the ground truth synaptic graph.

4.7 Functional subgraphs

We calculated motifs in the underlying synaptic graphs and found that all four clustering coefficients were equivalent when averaged across each graph, as expected.

To apply motif analysis to activity, we needed to infer functional graphs from spiking activity to summarize network dynamics. Directed edge weights in a functional graph represent the likelihood of a functional relationship in the activity between every pair of units.

We used mutual information (MI) to infer functional graphs from all spikes across the course of a trial, regardless of the trial's duration (complete or truncated). This results in a single functional graph for each trial. We chose to perform functional inference using the full spike set because this yields functional graphs with higher fidelity and greater sparsity.

The MI method we used is the confluent mutual information between spikes. At a conceptual level, an edge inferred from unit i to unit j using confluent MI means that unit j tends to spike either in the same time bin or one time bin after unit i spikes. Since spikes are binned at 10ms resolution, this method encompasses a delay of 0 to 20 ms. This delay is appropriate because we found that presynaptic spikes yielded a maximal response from all postsynaptic neurons at a delay of 5 to 20 ms.

Mathematically, we defined an indicator function on the spike train of neuron j, s(j) evaluating to 1 in the case where there is a spike at time t or t + 1, an indicator function on the spike train of neuron i, t(j), evaluating to 1 in the case where there is a spike at time t, and considered the

mutual information between them. The resulting networks were further processed by removing weights corresponding to neurons with negative pairwise correlations. Networks were then re-expressed to minimize skewness, and background signals were removed by accounting for background signal and considering weights as the residual resulting from linear regression on background strength. Finally we considered the z-normalized residual graph to account for heteroskedasticity (Chambers et al. 2018). This yields weighted values, for which we establish 0 as a threshold. All positive normed residual MI values are included in the full functional graph.

4.8 Recruitment graphs

The recruitment graph represents both the activity and the underlying connections of a network. A recruitment graph is defined separately for each 10 ms time bin of a given trial, thus yielding a temporal sequence of graphs. Each graph is calculated as the intersection of the functional graph, which is unique to every trial, and the active subgraph, which is unique to every 10 ms time bin. All edges in the recruitment graph come from underlying synaptic wiring, contained in the active subgraph, while edge weight values come from the inferred functional graph. In other words, for all edges $i \rightarrow j$ where $w_{functional,ij} > 0$ in the confluent MI functional graph and $w_{synaptic,ij} > 0$ in the active subgraph of time bin t, the edge in the recruitment graph for time bin t takes on the value of $w_{functional,ij}$. All other recruitment graph edges have value 0.

Just like the sequence of active subgraphs, there are 99 sequential recruitment graphs at 10 ms temporal resolution for every complete 1-second simulation trial. Triplet clustering coefficients were calculated for every unit on each 10 ms recruitment graph, then averaged across the

population to yield the whole-network clustering coefficients for that 10 ms time window. These methods allow us to observe how motif clustering changes in the recruitment graphs across time.

4.9 Clustering propensity

Networks may have very different connection densities, which would impact motif clustering coefficients. This is especially true as the active subnetwork changes in time, and for ER networks in comparison to clustered model graphs. We therefore used weighted and unweighted clustering propensity which enables meaningful comparisons between networks with different connection densities.

Our measure of weighted propensity begins with calculating the triplet motif clustering coefficients for each unit in the recruitment graph of every time bin. Then, for each time bin t we generate ten simulated graphs. These graphs have the same edges as the original recruitment graph at time t, with edge weights randomly sampled from the underlying distribution of functional edge weights. Motif clustering coefficients are calculated for units in each of the simulated graphs, then averaged for each unit and each motif type. The clustering coefficients of the units in the original graph at time t are normalized by the average of the ten simulated graphs' clustering coefficients, yielding the unit-wise clustering propensity at time t for each triplet motif. We used these values to perform all unit-wise motif analysis. In order to examine motifs at a whole-network level, for each motif type at time t we average across all units with nonzero clustering propensity values for that motif type.

Unweighted propensity is calculated similarly, considering the functional networks' unweighted directed clustering to that expected in both an ER graph as well as a small world graph. Thus, in addition to controlling for density, weighted propensity also measures the extent to which the specific edge weights in the recruitment graph impact triplet motif clustering, while unweighted propensity measures the same for specific structure of the recruitment graph. A weighted propensity of 1 indicates that specific edge weights play a negligible role in clustering, since random edge weights would still yield the same clustering coefficients, while an unweighted propensity of 1 indicates that the specific structure of the network is not important for clustering.

4.10 Erdős-Renyi graph simulations

Unclustered ER Graph simulations were performed using networks consisting of 1000 excitatory neurons, 200 inhibitory neurons and 50 Poisson input neurons. These populations were connected with $p_{e\to e} = 0.2$, $p_{i\to i} = 0.3$, $p_{i\to e} = 0.25$ and $p_{e\to i} = 0.35$. Synaptic weights relative to leak conductance were drawn from a log normal distribution (mean = 0.60, variance = 0.11), with i to e connections scaled up 50% (Chambers & MacLean 2016).

4.11 Overlap index

Overlap Index was used to measure the degree of overlap between two probability distributions. It is defined as $O = \sum_i min\, p_{i1}, p_{i2}$, where i is histogram bin index. If two distributions do not overlap at all they will have an overlap index of 0, if they are identical they will have an overlap index of 1.

4.12 Probability vectors

To quantify the cyclic transitions between relative prominence of motifs over time, we examined the dominant motif of the network for a given recruitment graph. We define the dominant motif of a graph as the maximum of the demeaned propensities for middleman, fan-in, and fan-out. We consider the demeaned values of each motif in order to account for the different relative magnitudes of motifs without affecting scaling in the cycle structure. Examining first order probabilities, which is the probability of a motif dominating a recruitment graph in a given run, on the time series of recruitment graphs from sustained and truncated runs shows that there is a significant difference between the distributions defining these values across each type of run.

To further characterize the transitions between different dominant motifs we fit a Markov model to the series of dominant motifs across recruitment graphs in both truncated and sustained networks. We again find a significant difference ($p < 1 \cdot 10^{-15}$, $n = 884$ for all Markov parameters). This all suggests that the failure to propagate found in some networks is tied to the inability to recruit the cyclic structure that we find to be a hallmark of sustained activity.

4.13 Statistical testing: Two-sample chi squared test

P values for comparisons between distributions of different types of network activity were done by two sample chi square test.

Parameter	Name	Value
C	membrane capacitence	281 pF
g_L	leak conductance	30 nS
E_L	leak reversal potential	-70.6 mV
E_E	excitatory reversal potential	0 mV
E_I	inhibitory reversal potential	-75 mV
V_T	spike threshold	-50.4 mV
Δ_T	slope factor	2 mV
τ_w	adaptation time constant	144 ms
τ_e	excitatory time constant	10 ms
τ_i	inhibitory time constant	3 ms
τ_p	poisson time constant	3ms
a	subthreshold adaptation	4 nS
b	spike triggered adaptation	.0805 nA

Table S1: Neuron parameters
Parameters used for simulation of adaptive exponential integrate and fire neurons.

ACKNOWLEDGEMENTS

We thank Isabel Garon for her expertise and aid in synchrony analysis. We thank Elizabeth de Laittre, Maayan Levy, Graham Smith, and Barbara Peysakhovich for helpful comments on our manuscript.

CHAPTER 2

Emergence of cross-tuned inhibition promotes task solution in trained spiking neural network models

This work will be published as a preprint in Summer 2023: Zhu, Y., Smith, C. M. B., Tang, M., Scherr, F., MacLean, J. N. (2023). Emergence of cross-tuned inhibition promotes task solution in trained spiking neural network models. *BioRxiv*.

ABSTRACT

Neocortex is composed of spiking neuronal units interconnected in a sparse, recurrent network. Spiking activity in such a network of neurons transforms sensory inputs to appropriate behavioral outputs. In this study, we use biologically realistic, task-optimized spiking neural network (SNN) models to evaluate how recurrent networks of spiking units change to achieve task computations. Networks are composed of excitatory and inhibitory units randomly interconnected with likelihoods and strengths matched to mouse neocortex. We employ a task that requires a binary report of moving visual stimuli, analogous to tasks that mice perform. We find that, through training, SNNs selectively adjust firing rates in response to the stimulus input, and that excitatory and inhibitory connectivity between input and recurrent layers change in accordance with rate modulation. Input channels that exhibit bias to one specific input developed stronger connections to recurrent excitatory units during training, while channels that exhibit bias to the other input developed stronger connections to inhibitory units. Furthermore, recurrent inhibitory units which were tuned to one input strengthened their connections to recurrent units of the opposite tuning. The convergence of trained network models on the specific pattern of cross-tuned inhibition highlights the significance of interneurons and their connectivity pattern in local circuit computations within neocortex. Overall, by combining realistic network features and optimizing models on ethologically meaningful tasks, our approach aims to bridge the gap

between neural network models and the neocortex, strengthening existing theories and revealing new ones on cortical behavior.

1 INTRODUCTION

Neocortex is a network of spiking neurons. These networks perform computations which are ultimately responsible for purposeful animal behavior. While definitions of "computation" vary, ranging from precise information calculation to metaphor, one way to study it is as the input / output transformation which underlies the generation of behaviors appropriate to sensory inputs. Through the use of task optimized spiking neural network (SNN) models, we investigated how networks of neurons in neocortex execute computations, and how these networks differ from networks that are not performing computations.

SNN models have been used to enrich our understanding of synaptic and cellular mechanisms that underlie neocortical activity in the absence of explicit computations. In particular, they have yielded mechanistic insights into structure-function relationships in spiking networks. Theoretical work has also produced compelling theories as to the relationships between neocortical structure, dynamics, and computation as proxied through measures such as information capacity (e.g. in Brunel 2016), dynamic range (e.g. in Shew et al. 2009), information transmission (e.g. in Mejias & Longtin 2012), and decodability (e.g. in Cohen et al. 2020). For example, asynchronous spiking activity, which is observed in neocortex, has been shown to enhance coding capacity (Kohn et al 2016; Lankarany & Prescott 2015). Recent advances in machine learning using spiking units (Lee et al. 2016; Huh & Sejnowski 2018; Bellec et al. 2020;

Zenke & Vogels 2021) have opened the possibility for neuroscientists to apply the same mechanism-uncovering approaches to task-optimized SNNs.

Trained SNNs enable expansive fidelity between model and biology, as they use spike-based computations and learning rules (Brette 2015; Verzi et al. 2018) and can capture biological network nonlinearities (Brette & Gerstner 2005). Spikes are a hallmark of neurons within nervous systems, and are potentially crucial in neocortical information processing. The relative timing of spikes has been demonstrated to carry information about stimulus features in neocortex (deCharms & Merzenich 1996). The discrete and sparse nature of spikes can help mitigate the impact of noisy inputs or other fluctuations (Calaim et al. 2022), especially when neuronal units themselves are also leaky (Sharmin et al. 2020). This makes SNNs especially suitable for modeling neural systems operating under realistic, noisy conditions. Individual spikes also have genuine efficacy, as they are used by animals to drive their own behaviors. The earliest stimulus-evoked spikes in mouse V1 are preferentially weighted for guiding behavior (Day-Cooney et al. 2021), and mice are capable of performing a visual discrimination task within a narrow time window during which the majority of V1 neurons involved in the task fire either one or no spikes (Resulaj et al. 2018).

Guided by experimental observations, we built biologically realistic SNNs with realistic structural and dynamic features and trained them to perform a task that mice are capable of. SNNs were composed of distinct excitatory and inhibitory adapting units that obeyed Dale's law, had sparse recurrent connections according to mouse visual cortical statistics (Billeh et al. 2020), long-tailed weight distributions (Song et al. 2005), and low spike rates (Koulakov et al. 2009;

Roxin et al. 2011). SNNs were trained with modified backpropagation through time (BPTT; Bellec et al. 2020) to give a binary report in response to the coherence level (low or high) of drifting dot videos.

Over the course of training, we found that networks solved the task by elevating firing rates in response to one coherence level and suppressing firing rates to the other. Specific patterns of connectivity emerged in support of this solution. In particular, recurrent units that have elevated firing–or tuning–to one coherence level developed strong inhibitory connections to units of the opposite coherence tuning. The emergence of this pattern of connectivity supports the theory that intracortical inhibitory feedback defines excitatory selectivity in neocortex by refining thalamic inputs (Carandini & Ringach 1997; Katzner et al. 2011), and points to the specific role of parvalbumin-expressing (PV) interneurons. Experimental and theoretical results suggest that PV neurons set the identity of the neuronal population associated with a given behavior by suppressing other neurons not associated with the behavior (Morrison et al. 2016; Bos et al. 2020; Lagzi et al. 2021).

2 RESULTS

2.1 Building and training biologically realistic spiking neural network models

We constructed each recurrent spiking neural network (SNN) model with adaptive leaky integrate and fire (ALIF) model neurons, or "units". Units are connected to one another via weighted, directed edges which simulate current-based synapses. Networks contained a total of 300 neuronal units; 240 were excitatory (e) and 60 were inhibitory (i), matching the 4:1 e:i ratio observed in neocortex (Figure 1A).

Models were initialized with additional neocortical structural properties. Initial excitatory weights followed a long-tailed distribution (Song et al. 2005), where $\mu = -0.64$ nA, $\sigma = .51$ nA, corresponding to a mean of 0.6005 nA and a variance of 0.1071 nA (Bojanek & Zhu et al. 2020) (Figure 1B). Inhibitory weights followed a similar distribution, but with weight values multiplied by -10 (Litwin-Kumar & Doiron 2012). A given excitatory neuron could only have positive outgoing weights, and a given inhibitory neuron could only have negative outgoing weights. Positivity and negativity of edges, i.e. the excitatory or inhibitory identity of each neuron, was maintained during training consistent with Dale's law.

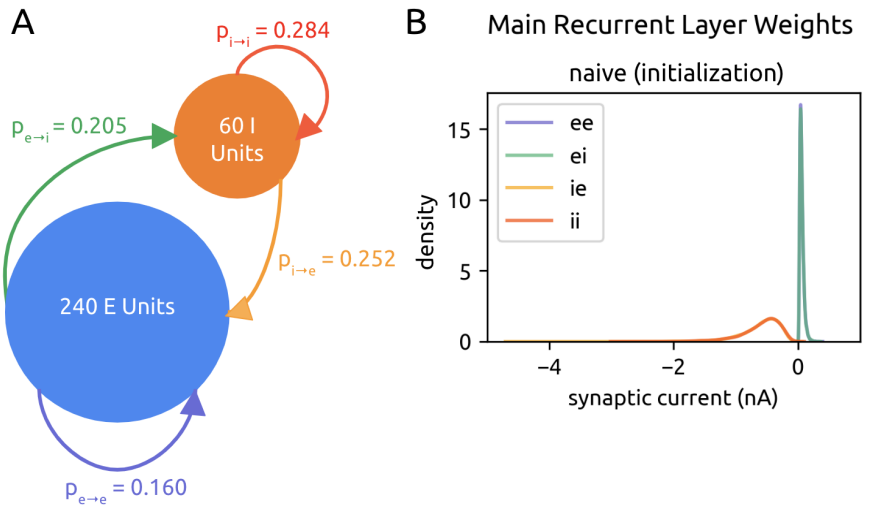

Figure 2.1: Network architecture

A. The main recurrent SNN is made of 240 excitatory (E) units and 60 inhibitory (I) units, connected within and between themselves with probabilities of connectivity taken experimentally from mouse neocortex (Billeh et al. 2020).

B. Distribution of naive (upon initialization) recurrent weights of e→e, e→i, i→e, and i→i connections, pooled across all experiments. Inhibitory weights are initialized to be -10x stronger than excitatory weights.

All units were sparsely and recurrently connected; the precise probabilities of connection within and between e and i populations are taken from mouse visual cortex ($p_{e\to e}$: 0.160, $p_{e\to i}$: 0.205, $p_{i\to e}$: 0.252, $p_{i\to i}$: 0.284, Billeh et al. 2020; Jabri and MacLean 2022) (Figure 1A). Precise connectivity of e and i populations was permitted to change during training. However, overall

sparsity was maintained according to the DEEP R algorithm (Bellec et al. 2018), details in Methods.

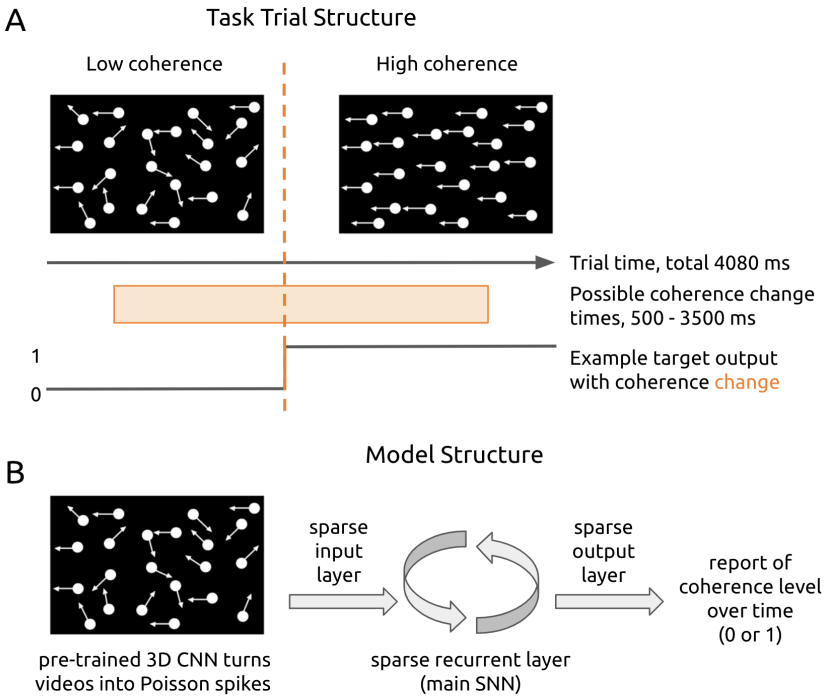

Figure 2.2: Task and model structure

A. In each trial, the model is presented with a 4080-ms video of drifting dots. Dots move in 1 of 4 global directions at low (15%) or high (100%) coherence. The task is to report the dots' coherence level (high or low) over time. In half of all trials, the coherence level changes at a random time between 500 to 3500 ms. The change can be either from high to low or vice versa.

B. The main SNN receives video input from 16 input channels in the form of Poisson spikes. These 16 input channels convey the activation of a velocity-trained 3D CNN (see Methods and Figure S1) in response to video input. The output of the SNN is a vector of '0's and '1's over time to signal the coherence level at each ms. Whether '0' or '1' indicates high coherence or low is toggled in different experiments.

Models were trained on a visual coherence change detection task of moving dot patterns, which mice are able to learn (Douglas et al. 2006; Kirkels et al. 2018; Marques et al. 2018) (Figure 2A). The SNN was presented with spiking output from a model of retina and thalamus (McIntosh et al 2016; see below and Methods) evoked by videos of drifting dots moving in one of 4 global directions (0°, 90°, 180°, 270°) and at two coherence levels (100% or 15%). Each trial had a total duration of 4080 ms. Half of all trials had a change in coherence which occurred at a random time between 500 and 3500 ms within the trial. The model is tasked to report the coherence level at all timepoints of the trial instantaneously as

output (Figure 2B). The target output sequence is composed of 0's and/or 1's (Figure 3). The numerical label assigned to each coherence level is randomly swapped in different experiments.

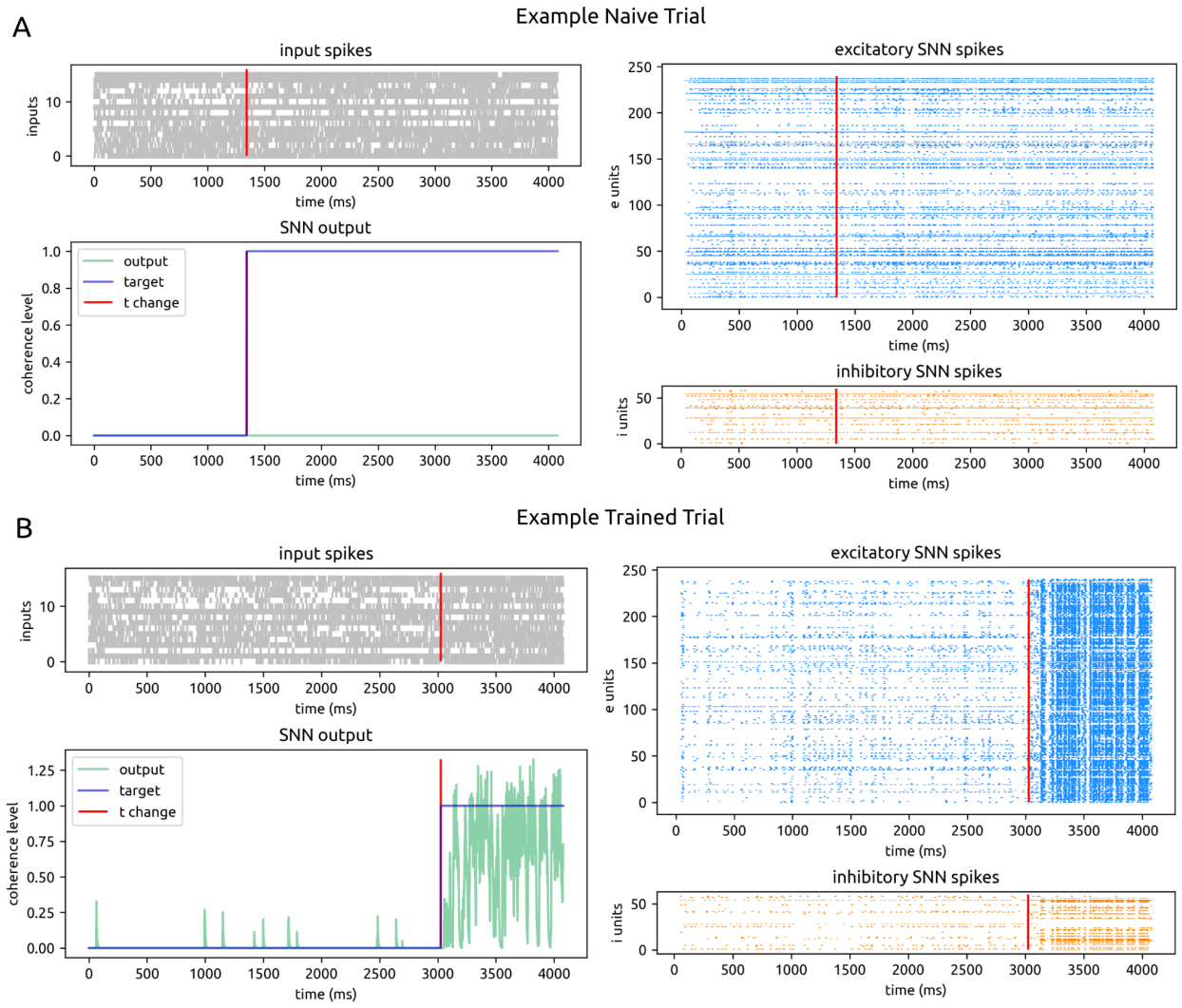

Figure 2.3: Example model activity

A. An example of a naive trial in which a coherence change occurred. All plots are displayed over time (4080 ms trial duration) on the x-axis. The red vertical line shows the time of coherence change. Top left: spikes from 16 input channels. Bottom left: model output for coherence level in green; true target coherence level in violet. Top right: spikes from 240 excitatory units in the main recurrent SNN. Bottom right: spikes from 60 inhibitory units in the main recurrent SNN.

B. Same as (A) for an example trained trial. Note the higher firing rates for the coherence level '1' and lower rates for '0', which we report on further in section 2.2 and summarize in Figure 6.

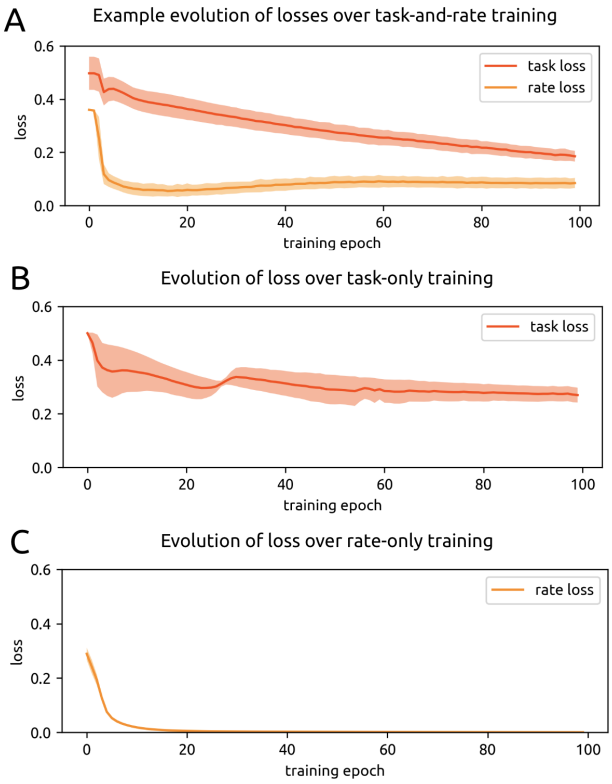

Figure 2.4: Loss

A. Losses over training for an example experiment in which the model was trained to minimize both task (red) and rate (orange) loss. Shaded areas are the standard deviation of the losses across trials in each epoch.
B. Loss over training for all experiments in which the model was trained to minimize only task loss. Shaded area is the standard deviation of the epoch task loss across separate experiments.
C. Same as (B) for only rate loss training.

To make input videos interpretable to the main recurrent SNN, a 3D convolutional neural network (CNN) model (McIntosh et al. 2016) was used to turn videos into spike sequences (Figure 3, Figure S1A). This CNN was pre-trained to report the velocity (x,y) of global motion in a combined dataset of drifting dot videos and black-and-white natural motion videos (see Methods, Figure S1A). The output activations of the 16 units in the CNN's next-to-last layer are interpreted as firing rates, from which Poisson spikes were newly generated for each trial. These 16 units were the SNN's input channels.

The MSE between the model's coherence report and the target coherence label ('0' or '1') is the task loss. To maintain naturalistic spiking dynamics, a target spike rate was specified (20 Hz, or 0.020 spikes/ms), and the MSE between the recurrent SNN's actual spike rate from the target rate could be added to the task loss (Zhu et al. 2020). Networks were trained in three ways: to minimize task loss, to minimize rate loss, and to minimize both task and rate loss (Figure 4). We report results primarily for the case

of training on both task and rate loss; results from training on only rate and only task loss are reported as controls / points of contrast.

Since networks were all initialized with the same protocol regardless of the class of loss optimization, the naive starting losses across all training sessions (n = 69) were similar, with naive task loss at 0.498 ± 0.073 and naive rate loss at 0.394 ± 0.080. Across all task-and-rate training sessions (n = 20), models achieved 0.271 ± 0.127 task loss and 0.189 ± 0.174 rate loss after training over 10,000 batch updates, or 100 epochs (Figure 4A). After training on task alone (n = 16), models achieved 0.273 ± 0.021 task loss. After training on rate alone (n = 33), models achieved 0.0007 ± 0.0002 rate loss. Thus training on rate alone leads to better rate performance than when training to minimize both rate and task, but training on task alone does not lead to better task performance than when training to minimize both rate and task loss.

Weights of all layers (input, main recurrent SNN, and output) were permitted to change during each class of training. We used the Adam optimizer, which adapts the learning rate for every variable over the course of training (details in Kingma & Ba 2015). SNNs were trained using backpropagation-through-time (BPTT) with modifications for spiking (see Methods, Huh & Sejnowski 2018; Bellec et al. 2020; Zenke & Vogels 2021).

SNN input and output layers were sparse and matched to the connectivity of the main recurrent and output layers (Figure 5). We also specified that no recurrent units which received input could directly project to output. We arrived at this set of constraints through the series of investigations reported in section 2.6. By stipulating these architectural and training details, we ensured that

weight changes which supported training preferentially took place in the recurrent SNN layer, and thus the recurrent SNN layer would play the most important role in solving the task.

2.2 Models maintain a long-tailed weight distribution

Synaptic weights are long tailed in neocortex and are initialized as such in our models. This distribution is not enforced during training, yet we observed that a long tailed distribution is preserved following all varieties of training (Figure 1C). This result supports the hypothesis that the long-tailed distribution of synaptic strengths observed in neocortex may be beneficial for performing computations.

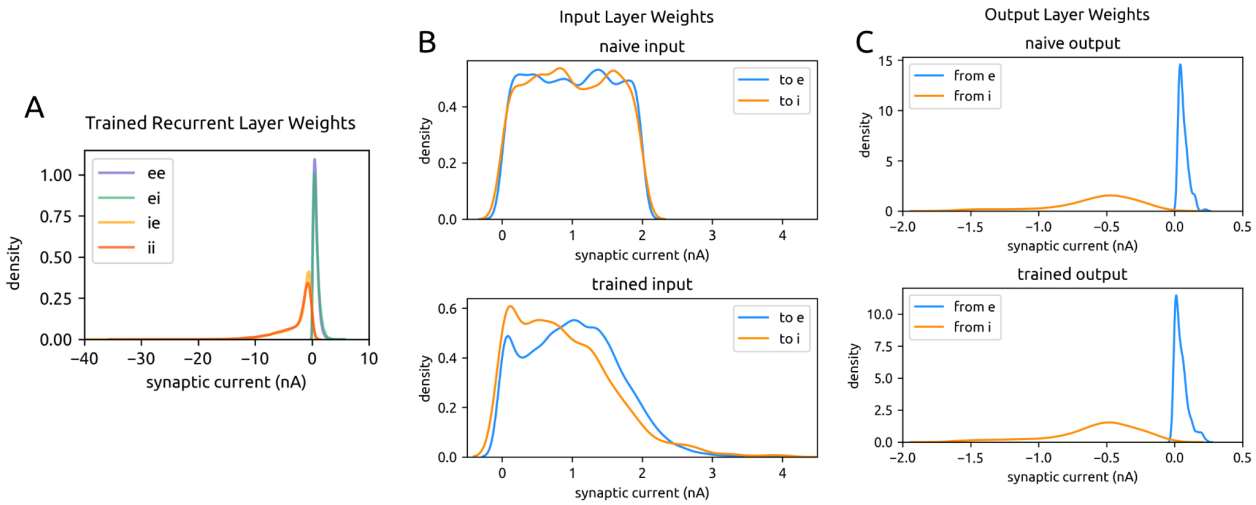

Figure 2.5: Weight changes over task-and-rate training

A. Trained weight distributions for the main recurrent SNN. Connections are separated by e→e, e→i, i→e, and i→i type, and weights are pooled across all experiments. Naive recurrent weights are shown in Figure 1B. Note that the long-tailed weight distribution is maintained through training, but scaled roughly 10x.
B. Naive (top) and trained (bottom) input layer weights to main SNN excitatory units (blue) and inhibitory units (orange), pooled across all experiments. Input weights were initialized with a uniform distribution [0, 2] and developed skewness with a longer positive tail over the course of training.
C. Same as (A) for the output layer; in blue are the weights from main SNN excitatory units and in orange are the weights from inhibitory units. Output layer weights statistically did not change over the course of training.

After training on task and rate, weights in the recurrent layer became approximately 10x stronger overall (Figure 5A). Separating the recurrent network by excitatory and inhibitory units, weights change as follows: e→e: 0.001 ± 0.026 naive to 0.113 ± 0.358 trained; e→i: 0.012 ± 0.028 naive to 0.081 ± 0.341 trained; i→e: -0.150 ± 0.307 naive to -0.790 ± 2.131 trained; i→i: -0.169 ± 0.318 naive to -0.768 ± 2.078 trained; $p \approx 0.0$ for all naive and trained recurrent weight distributions. Weights in the input layer underwent moderate changes (in→e: 0.325 ± 0.574 naive to 0.432 ± 0.999 trained, $p = 2.761 \cdot 10^{-152}$; in→i 0.288 ± 0.549 to 0.346 ± 0.945 trained, $p = 7.923 \cdot 10^{-24}$) (Figure 5B). Weights in the output layer showed the least difference between naive and trained states; outputs from inhibitory units to outputs do not change at all over training (e→out: 0.010 ± 0.027 naive to 0.006 ± 0.023 trained, $p = 6.268 \cdot 10^{-15}$; i→out: -0.148 ± 0.302 naive to -0.145 ± 0.298 trained, $p = 1.0$) (Figure 5C). Weight changes for rate-only-training and task-only-training are similar to the above and are reported in Methods.

We investigated whether the edges which began with the strongest naive weights were also the edges with the strongest trained weights. We identified the recurrent edges which had the top decile of starting absolute weights and tracked them over the course of task-and-rate training. After completing training, the percentage of those edges which still contained the top decile of weights was only 3.909 ± 1.377% (compared to 100% if all edges beginning in the top decile remained in the top decile). When we broadened to tracking the starting top quartile, after training only 10.028 ± 2.857% of those edges remained in the top quartile. Therefore, starting weights do not pre-determine final weights in the recurrent network.

2.3 Models modulate firing rates to solve the task

We found that models increasingly modulated their firing rates according to the output label associated with each coherence level over the course of training. Network models came to increase firing rates in response to the coherence labeled ‘1’ and decreased firing rates to the coherence labeled ‘0’ (Figure 3B; Figure 6). It did not matter whether ‘1’ was attached to low or high coherence–networks trained with swapped labels (when ‘1’ means 100% coherence vs when ‘1’ means 15% coherence) exhibited the same behavior.

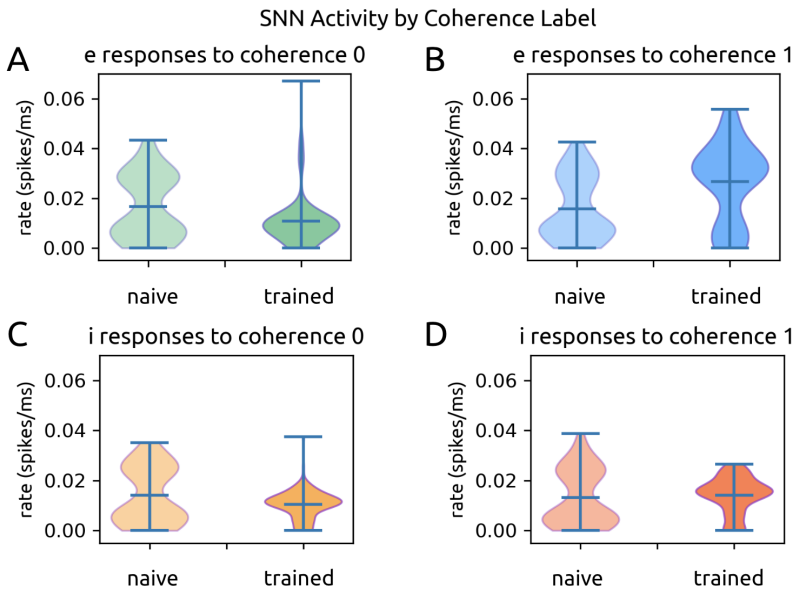

Figure 2.6: SNN activity in response to coherence levels

A. Excitatory units’ firing rates in response to coherence level ‘0’ in the naive (left) and trained (right) states. Rates are pooled over all task-and-rate training sessions.
B. Same as (A) for coherence level ‘1’.
C. Same as (A) for inhibitory units.
D. Same as (C) for coherence level ‘1’.

In the naive state across all training sessions (n = 20), excitatory units responded with 0.017 ± 0.013 spikes/ms to coherence ‘0’ and 0.016 ± 0.013 spikes/ms to coherence ‘1’, with p = 0.376 comparing responses to the two coherence level labels (Figure 6A;B). Inhibitory units responded with 0.014 ± 0.011 spikes/ms to coherence ‘0’ and 0.013 ± 0.011 spikes/ms to coherence ‘1’, with p = 0.040 (Figure 6C;D). Thus in the naive state, excitatory and inhibitory units responded similarly to both coherence levels.

In the trained state (following 10,000 batch updates, or 100 epochs) across all sessions, excitatory units responded with 0.011 ± 0.010 spikes/ms to coherence '0' and 0.027 ± 0.016 spikes/ms to coherence '1', $p = 6.661 \cdot 10^{-16}$ (Figure 6A;B). This demonstrates both a decrease and increase in firing rates following training according to label, resulting in approximately a 2.5x increase in firing rates to the higher coherence label. Similarly, inhibitory units decreased firing rates to 0.011 ± 0.006 spikes/ms to coherence '0' and increased firing 0.014 ± 0.007 spikes/ms to coherence '1', $p = 1.147 \cdot 10^{-13}$ following training (Figure 6C;D). We next investigated how the models' connectivity changed over the course of training in order to support this reliable behavior and find that both the inputs and the recurrent layer of the SNN change inhibitory connectivity to achieve task accuracy.

2.4 Input channels strengthen connections to excitatory or inhibitory recurrent units according to their own tuning to coherence levels

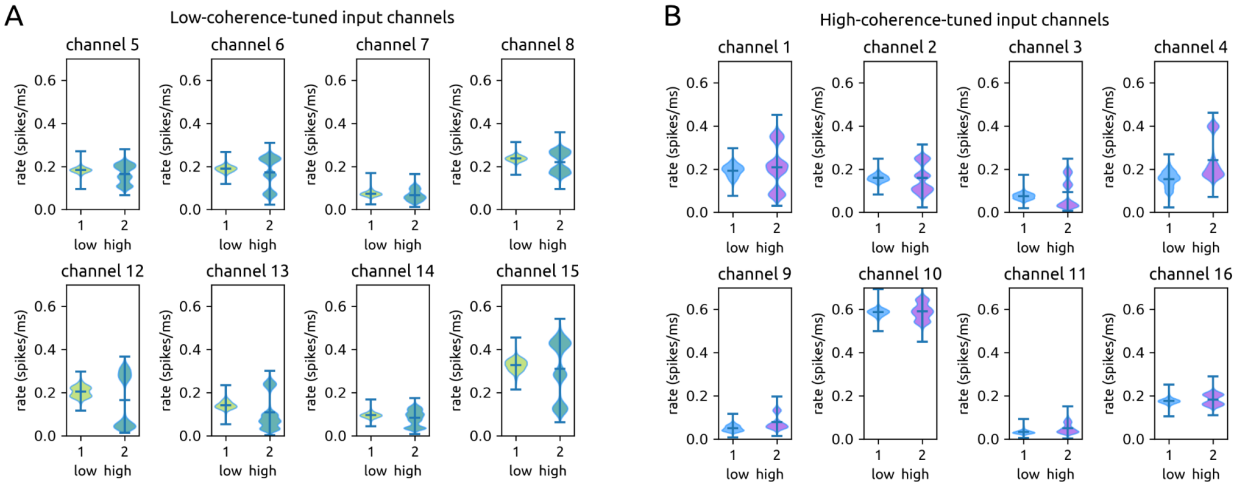

Figure 2.7: Tuning of input channels

A. Violin plots of input channels' firing rates in response to low (15%, left violins) and high (100%, right violins) coherence input. (A) shows the 8 channels that have higher mean firing rates for low coherence.
B. Same as (A) for the 8 channels that have higher mean firing rates for high coherence.

All 16 input channels had different mean firing rates to one coherence level versus the other (all $p \approx 0.0$). However, the distance (absolute difference) between channels' mean responses to one coherence vs the other was small, measuring 0.021 spikes/ms on average. The difference between mean rates (high coherence minus low coherence) are as follows for all 16 channels: 0.0163, 0.0002, 0.0192, 0.0874, -0.0176, -0.0163, -0.0063, -0.0170, 0.0284, 0.0033, 0.0180, -0.0395, -0.0336, -0.0118, -0.0166, 0.0071 spikes/ms.

Notably half of all channels responded with elevated mean firing during periods of low coherence (mean rate to low coherence: 0.181 ± 0.078 spikes/ms; rate to high coherence 0.162 ± 0.108 spikes/ms; $p \approx 0.0$) (Figure 7A). The other half responded more to high coherence (mean rate to low coherence: 0.178 ± 0.165 spikes/ms; rate to high coherence: 0.201 ± 0.160 spikes/ms; $p \approx 0.0$) (Figure 7B). We will refer to these as low-coherence-tuned and high-coherence-tuned input channels respectively.

In response to high coherence input, input channels exhibited multimodal rate distributions. However, in response to low coherence input, the input channels' rate distributions were largely unimodal. When we consider medians instead of means, 10 channels had greater median rates in response to low coherence input, and 6 channels had greater median rates in response to high coherence input. The average distance between all medians was 0.023 ± 0.029 spikes/ms. When we consider modes, 5 channels had greater modes in response to low coherence input, and 6 channels had greater modes in response to high coherence input. 5 channels had the same modes to both low and high coherence, further illustrating the similarity of input responses to the two coherence levels. The average distance between all modes was 0.045 ± 0.048 spikes/ms. We

chose to use the mean as the measure of central tendency to define tuning, as the majority of high coherence rate distributions were trimodal, making the mean the best measure for comparison across all distribution shapes.

Each input channel synapses onto subpopulations of units in the recurrent layer, including both excitatory and inhibitory units. In the naive state, low-coherence-tuned input channels and high-coherence-tuned input channels had similarly weighted connections onto both excitatory (low-coherence-tuned: 0.198 ± 0.009; high-coherence-tuned: 0.204 ± 0.004) and inhibitory units (low-coherence-tuned: 0.187 ± 0.010; high-coherence-tuned: 0.172 ± 0.011) in the recurrent layer. Over the course of training, the two coherence-tuned input channel groups diverge in the strength of their synapses onto excitatory and inhibitory recurrent units (Figure 8A). The recurrent layer then amplifies this effect, as we report in a later section.

When low coherence is labeled as output ‘0’ and high coherence is labeled as output ‘1’ during training, low-coherence-tuned input channels increase weights onto inhibitory units (0.249 ± 0.032) while decreasing weights onto excitatory units (0.212 ± 0.014). On the other hand, high-coherence-tuned input channels develop stronger weights onto excitatory units (0.263 ± 0.027) and weaker weights to inhibitory units (0.135 ± 0.007). In this way, when a low coherence input is presented, the network becomes inhibition dominated, leading to lower spike rates and a report of the lower output value. Conversely when a high coherence input is presented, the network becomes driven by excitation and reports the higher output value.

If the labels are swapped, the opposite changes in connectivity occurred. Low-coherence-tuned

input channels developed stronger weights onto excitatory units and high-coherence-tuned input channels developed stronger weights onto inhibitory units. Thus, over the course of training, input channels that preferentially responded to a particular coherence level–regardless of the magnitude of difference– became more strongly connected to either excitatory or inhibitory units resulting in either high or low firing states to match the desired output label.

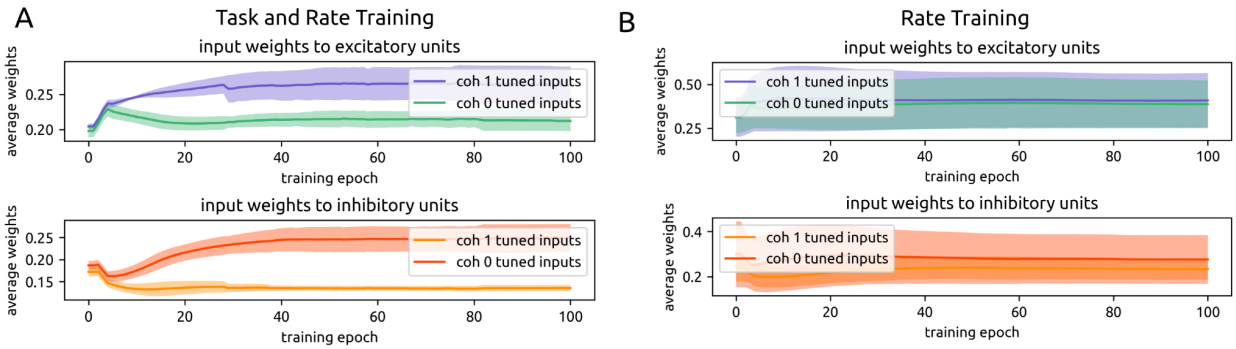

Figure 2.8: Selective strengthening of input layer weights according to tuning

A. Over the course of task-and-rate training, input channels tuned to coherence 1 develop stronger connections to excitatory units (top), while input channels tuned to coherence 0 develop stronger connections to inhibitory units (bottom).
B. This pattern is not observed during only rate training.

This result can be summarized as a ratio of 1-tuned input weights to 0-tuned input weights onto inhibitory and excitatory SNN units. Again, 0-tuning can refer to either low or high coherence-tuned; what is important is which particular coherence is given the '0' label for that experiment. For inhibitory units, this ratio begins at 0.921 in the naive state and becomes 0.543 after training. This means that through training inhibitory units receive approximately twice as much input from 0-tuned input channels. For excitatory units, this ratio begins at 1.031 and becomes 1.239 after training. Thus following training, excitatory units receive moderately more input from 1-tuned input channels.

In versions of the model trained only on rate, we do not observe this separation of input channels onto excitatory or inhibitory recurrent populations (Figure 8B). For inhibitory units, the average ratio of 1-tuned input weights / 0-tuned input weights begins at 0.970 and becomes 0.821. For excitatory units this ratio begins at 1.010 and becomes 1.054. This is expected, as there is no imperative to report '1' or '0' in response to any feature of the input in rate training.

The input layer is not the only part of the model that changes to enable rate modulation–the recurrent layer itself plays a role in amplifying this behavior.

2.5 Recurrent inhibitory units strengthen connections to recurrent units of opposite tunings

Similar to input channels, recurrent units also demonstrated tuning to one or the other coherence level. We defined the tuning of recurrent units as the coherence label to which units responded with greater mean firing rates in the network's final trained state.

In the task-and-rate-trained SNN, the majority of excitatory units respond with elevated mean firing to the coherence level labeled as '1' (193.8 ± 60.0 out of 240 total excitatory units). A smaller proportion (27.1 ± 15.7 excitatory units) respond with elevated firing to the coherence level labeled as '0'. Similar numbers of inhibitory units respond with elevated firing to '1'-labeled (21.6 ± 7.5 out of 60 total inhibitory units) as to '0'-labeled (21.3 ± 8.2 inhibitory units) coherence input.

We plotted the mean weights within and between '0'- and '1'-tuned populations for e→e, e→i, i→e, and i→i connections over the course of task-and-rate training (Figure 9). Again, these

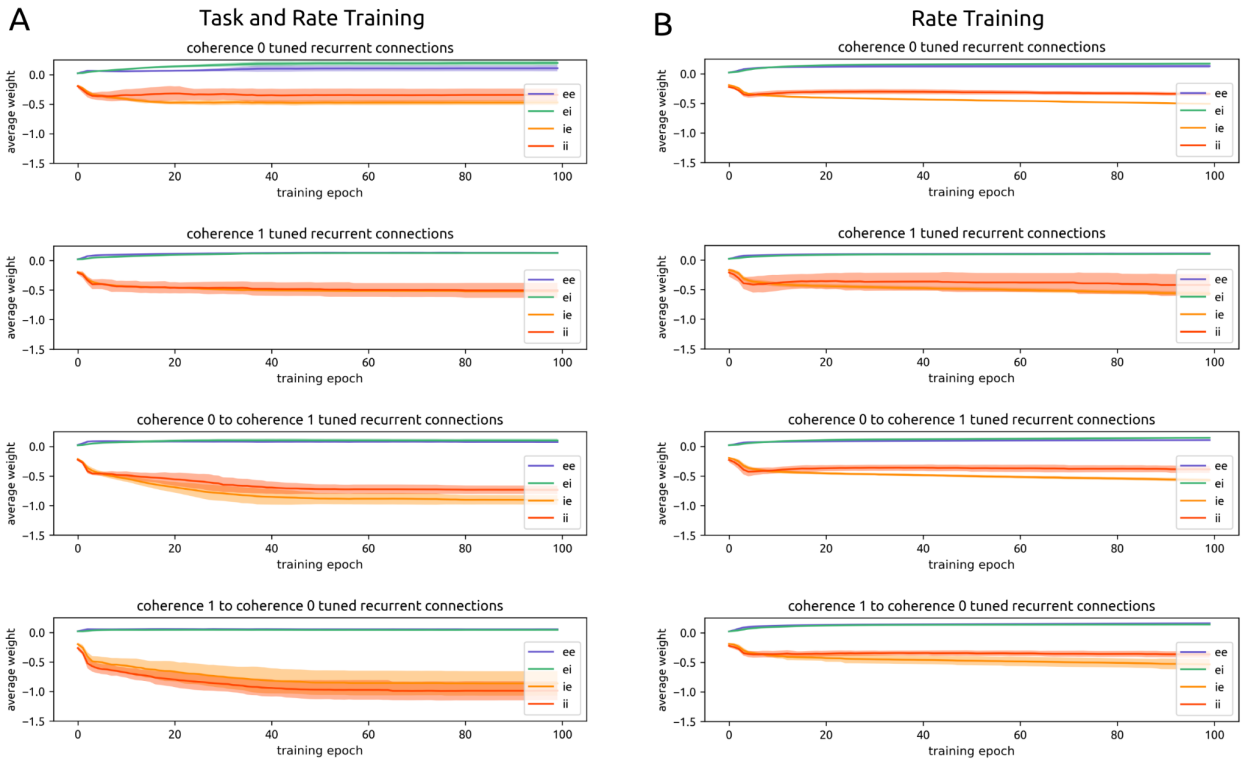

Figure 2.9: Selective strengthening of recurrent cross-tuning inhibition

A. Over the course of task-and-rate training, inhibitory recurrent units develop the strongest connections to recurrent units of the opposite tuning (bottom two plots).
B. This pattern is not observed when models are trained on only rate.

populations are defined according to their tuning at the end of training, so plots track retrospectively how these units' weights evolved over training in conjunction with the emergence of tuning. We found that differences arise in the connectivity between recurrent units according to the units' final coherence tuning. In particular, recurrent inhibitory units send the strongest connections to units of the opposite tuning to themselves (Figure 9A).

Connectivity for excitatory units to other excitatory units is as follows: '0'-tuned to '0'-tuned: 0.021 ± 0.007 naive to 0.104 ± 0.052 trained; '1'-tuned to '1'-tuned: 0.025 ± 0.009 naive to 0.128 ± 0.011 trained; '0'-tuned to '1'-tuned: 0.024 ± 0.007 naive to 0.076 ± 0.015 trained; '1'-tuned to '0'-tuned: 0.022 ± 0.007 naive to 0.056 ± 0.013 trained. Over training, recurrent

excitatory weights increase more within tuning (approximately 5x) than across tuning (approximately 3x).

Connectivity for excitatory units to other inhibitory units is as follows: ‘0’-tuned to ‘0’-tuned: 0.030 ± 0.014 naive to 0.191 ± 0.046 trained; ‘1’-tuned to ‘1’-tuned: 0.027 ± 0.012 naive to 0.125 ± 0.020 trained; ‘0’-tuned to ‘1’-tuned: 0.025 ± 0.010 naive to 0.097 ± 0.033 trained; ‘1’-tuned to ‘0’-tuned: 0.024 ± 0.010 naive to 0.045 ± 0.004 trained. Once again, over training, excitatory weights onto inhibitory units increase more within tuning (approximately 5x) than across tuning (approximately 3x).

Connectivity for inhibitory units to excitatory units is as follows: ‘0’-tuned to ‘0’-tuned: -0.226 ± 0.075 naive to -0.523 ± 0.102 trained; ‘1’-tuned to ‘1’-tuned: -0.243 ± 0.090 naive to -0.605 ± 0.138 trained; ‘0’-tuned to ‘1’-tuned: -0.266 ± 0.100 naive to -0.939 ± 0.158 trained; ‘1’-tuned to ‘0’-tuned: -0.267 ± 0.118 naive to -0.978 ± 0.224 trained. In contrast to excitatory units, over training, inhibitory weights onto excitatory units increase moderately within tuning (approximately 2x) and increase more strongly across tuning (approximately 4x).

Connectivity for inhibitory units to other inhibitory units is as follows: ‘0’-tuned to ‘0’-tuned: -0.233 ± 0.067 naive to -0.411 ± 0.149 trained; ‘1’-tuned to ‘1’-tuned: -0.235 ± 0.059 naive to -0.519 ± 0.138 trained; ‘0’-tuned to ‘1’-tuned: -0.280 ± 0.106 naive to -0.855 ± 0.248 trained; ‘1’-tuned to ‘0’-tuned: -0.327 ± 0.135 naive to -1.097 ± 0.266 trained. Over training, inhibitory units moderately increase weights to other inhibitory units of the same tuning (approximately 2x)

and more strongly increase weights to other inhibitory units of the opposite tuning (approximately 3x).

Thus, over the course of training, excitatory units focus on increasing weights to other units of the same tuning, while inhibitory units focus on increasing weights to units of the opposite tuning. Two-sample KS-testing confirms that the weight distributions are different for within- and across-tuning excitatory connections (p = comparing e→e within- vs. across-tuning; p = comparing e→i within- and across-tuning). This is also true for within- and across-tuning inhibitory connections ($p = 6.323 \cdot 10^{-14}$ comparing i→i within- vs. across-tuning; $p = 5.409 \cdot 10^{-11}$ comparing i→e within- vs. across-tuning).

This last result can be summarized as a ratio: mean inhibitory weight across-tuning / mean inhibitory weight within-tuning. For naive networks, this ratio is 1.196. After training, this ratio increases to 1.904. Inhibitory units in trained networks send connections that are approximately twice as strong to opposite-tuned recurrent units compared to same-tuned recurrent units.

Because inhibitory weights are initialized to be 10x stronger than excitatory weights, over the course of training they become proportionally stronger still. Through their strong cross-tuning connectivity, inhibitory units enable the appropriate modulation of high and low recurrent firing rates in response to ‘1’ and ‘0’ labeled coherences.

When networks are trained on rate alone, this pattern of stronger cross-tuning inhibition is not observed (Figure 9B). The average ratio of inhibitory weights across-tuning / within-tuning is

1.058 for naive networks and 1.006 for trained networks. A two-sample KS-test confirms that the weight distributions are the same for within- and across-tuning inhibitory connections for rate-trained models ($p = 0.239$ comparing i→i within- vs. across-tuning; $p = 0.156$ comparing i→e within- vs. across-tuning).

2.6 When unconstrained, models find task solutions that bypass the recurrent SNN

Through stipulation of architectural and training details, such as not permitting recurrent units that receive input to also project to output, we biased the weight changes which enabled training to preferentially take place in the recurrent layer. This was desirable to us so that we could study how a recurrent network of spiking units changes to yield computation. We arrived at our set of constraints–which we used for all models described above–following the observation that, in the absence of these constraints, models tended to bias most of their weight changes to occur in the input and output layers.

We began with a version of the model in which only the main recurrent layer had neocortical structural constraints. Input and output layers were not constrained upon initialization nor during training. They were instead densely connected to the recurrent layer and initialized with uniform weight distributions (see Methods). In this version of the model, output layer weights were observed to change the most (0.112 ± 0.052 absolute mean difference between all naive and trained weights), relative to input (0.005 ± 0.003) and recurrent (0.003 ± 0.002) weights during training ($n = 13$ sessions) by approximately 22x and 37x respectively. We interpreted this to mean that the recurrent dynamics corresponding to a particular coherence-level input became

"mapped" by the output layer to the correct label over the course of training. There was less need to alter those recurrent dynamics through updating recurrent or input layer weights.

With the goal of coaxing the recurrent SNN to be truly involved in solving the task, we sparsified the output layer to match that of the main recurrent layer and maintained this sparsity over training (n = 61 sessions). However, the model still utilized a strategy that bypassed the recurrent layer.

Since there were no restrictions on direct connections from input-receiving units in the recurrent layer to the output layer, the network made use of this shortest path to strongly route the input to the output. Specifically, excitatory units in the recurrent layer which received the strongest sum of input weights also sent the strongest sum of output weights (Figure 10A). Inhibitory units exhibited a complementary pattern: those which received the weakest sum of input weights sent the strongest inhibitory outputs (Figure 10B). Thus over the course of training, the model removes inhibitory drive onto the output so that inputs can most directly exert their effects through single-synapse excitatory connections.

Once again, we desired to push the computation of task solutions onto the recurrent layer. We therefore sparsified the input layer to match the connectivity of the main recurrent and output layers. We also specified that no recurrent units which receive input could also project to output. We used this version of the model for the above analyses, and these are the models in which we observed the specific connectivity changes that arose in conjunction with high and low rates to match coherence level labels.

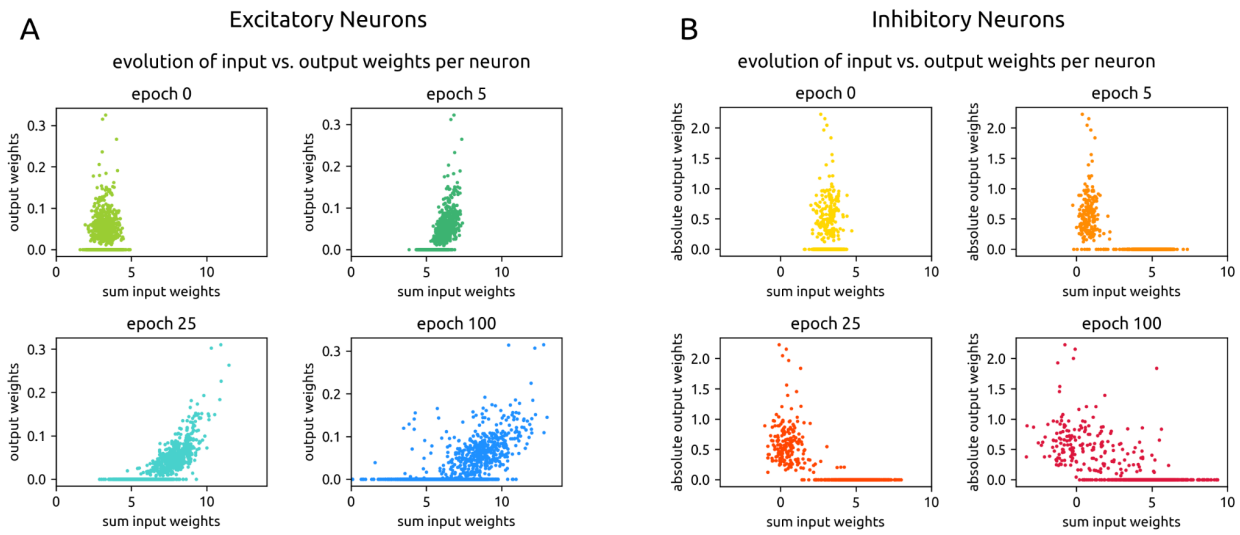

Figure 2.10: Weight changes when input / output layers are unconstrained

A. In a prior version of the model with fewer connectivity constraints, the excitatory units which received the strongest sum of inputs also sent the strongest outputs after training. In this way, the model could most efficiently let the input drive the output. We see this solution emerge over the course of training, in which there is no relationship between a unit's input and output weights in the naive state (epoch 0, top left), but the positive relationship emerges over time to become established in the trained state (epoch 100, bottom right).

B. Same as (A) for inhibitory units. We observe the opposite pattern emerge for inhibitory units: those which receive the weakest sum of inputs send the strongest inhibitory outputs. Thus over the course of training, the model removes inhibitory drive to the output so that excitatory units can most directly exert their effects.

3 DISCUSSION

Using biologically realistic, task-optimized SNN models, we found that recurrent networks of spiking units selectively elevate or depress firing rates in response to a coherence-report task. We achieved a level of interpretability, finding that excitatory and inhibitory connectivity of the input and main recurrent layers changed in conjunction with this rate modulation.

Input channels that responded preferentially to the coherence level labeled as '1' (numerical labels attached to low or high coherence were swapped in certain experiments) developed stronger connections to excitatory units in the main recurrent SNN. In contrast, input channels that preferred coherence level '0' strengthened connections to inhibitory units. This was the first step leading to excitation-dominated or inhibition-dominated SNN activity. This effect was further amplified through connectivity patterns in the main recurrent SNN. Recurrent inhibitory units that were strongly tuned to one coherence label at the end of training demonstrated a strengthening of connections to both excitatory and inhibitory units of the opposite tuning. Thus when one coherence level was presented, inhibitory units would suppress other units that were not part of the appropriate coherence tuning. This arose over the course of training. In such a way, small but significant differences in the input statistics are exploited, and the recurrent network employs strong cross-tuning inhibition to further adjust its responses to be appropriate to the input.

A biological parallel to the emergence of this pattern–in which units tuned to one coherence level came to inhibit units of opposite tuning–can be found in the rich literature of cross-orientation suppression of V1 neurons (Morrone et al. 1982; Eysel et al. 1990; DeAngelis et al. 1992; among others) and of the role of interneurons in establishing V1 excitatory selectivity to stimulus orientation and direction. While the relative contributions of thalamic (feedforward) and intracortical (feedback) connections to establishing V1 tuning is under continued study (Ferster & Miller 2000; Alitto & Dan 2010; Katzner et al. 2011), one theory is that selectivity arises through a combined effect of initially broad tuning via LGN inputs that is then sharpened through intracortical feedback (Carandini & Ringach 1997). Our modeling results support this

theory in that the strict dichotomy between thalamic and intracortical contributions is false. We find that intrinsic tuning from our input channels, which can be interpreted as modeling thalamic inputs, was refined by recurrent connections in our main SNN, which can be interpreted as a model of V1. Over the course of task learning, local recurrent circuitry changed to amplify the input tuning. Under training conditions that are agnostic about the specifics of excitatory and inhibitory wiring, the solution was for learning to drive the formation of recurrent cross-tuned inhibition.

Intracellular data and modeling work suggests that this local intracortical feedback takes the form of inhibitory connections from interneurons that have the same–rather than broader–selectivity as excitatory targets (Katzner et al. 2011). By matching excitatory selectivity, interneurons can precisely keep responses to undesired stimuli below the spike threshold. This is supported by the variety of tuned connectivity observed in our trained models, and the mechanisms underlying this behavior are aligned in both neocortex and the model, since the activity of our model neurons is also fundamentally based on thresholding.

A potential candidate responsible for establishing excitatory selectivity in neocortex is parvalbumin-expressing (PV) interneurons. PV neurons synapse onto the soma or axon of synaptic partners (Kepecs & Fishell 2014), allowing them to tightly control spiking in postsynaptic neurons. We used a generic inhibitory neuron in our models, but this property of direct control is a feature of our inhibitory model units. This property makes PV neurons a candidate for selecting neurons that are involved in task-relevant assemblies. Theoretical work suggests that PV neurons stabilize new groups of task-associated excitatory neurons (Bos et al.

2020, Lagzi et al. 2021), which is consistent with experimental work in associative learning (Morrison et al. 2016).

While there were various possible network configurations that could have achieved tuning, such as strengthening excitatory connections, it is noteworthy that our networks converged on this specific pattern of cross-tuning inhibition through training. This result may underscore the significance of PV neurons and their connectivity pattern in establishing local circuit computations in the neocortex. A natural extension of this work is to diversify neuronal subtypes in future models, as each exhibits unique connectivity properties in local circuits, which may point to distinct computational roles as well (Kepecs & Fishell 2014; Cone et al. 2019).

In order to reach the cross-tuning inhibition solution, learning took place in the form of structural changes in the model's recurrent layer–the SNN itself. By contrast, when we built models without certain architectural constraints, we found that they tended to arrive at solutions that did not change recurrent connectivity. Instead, these solutions exploited the SNN as a reservoir. The model either strengthened the shortest path between input and output, or the output layer weights changed the most over the course of training. Rather than altering recurrent dynamics through changing recurrent weights, the model solved the task by learning to map the pre-existing, naive dynamics–which occur in response to a particular input–to the correct output label. This is not dissimilar from a liquid state machine (LSM), a type of reservoir computer that makes use of the spatiotemporal dynamics of recurrent spiking units and linear readouts to achieve a wide variety of tasks (Buonomano & Maass 2009).

Through this work, we provide two circuit mechanisms–one based on input and another based on recurrent connectivity changes–that underlie task learning. Moreover, we achieve this result in a spiking network, making the circuit mechanisms more directly applicable to networks of spiking neurons in neocortex. We believe that task-trained SNNs demonstrate promise for advancing the understanding of network computation and applying that understanding to neocortex, and look forward to future work that adopts this method for this goal.

4 METHODS

4.1 Model construction

The models we are building and training are recurrent spiking neural networks (SNNs). Each model is built with spiking units which represent individual neurons. Units are connected with one another via weighted, directed edges which represent synapses. The weight of an edge W_{ij}^{rec} is a numerical value that indicates the strength of the connection from presynaptic unit i to postsynaptic unit j within the recurrent SNN.

In a network of 300 neuronal units, 240 are excitatory and 60 are inhibitory. They are built this way to maintain the 4:1 e:i ratio, which is a ratio observed in neocortex. All synaptic edges originating from an excitatory unit have positive weight values, and all from an inhibitory unit have negative weight values. Positivity and negativity of each edge is maintained during training, although values may change (Zhu et al. 2020).

4.2 Input

Input to the main SNN will vary depending on the task of interest. The task and input preprocessing are described in detail in the "Task" section. As an overview, input is delivered in the form of spike sequences onto a subset of the main SNN units. Input spikes are also weighted (e.g. W_i^{input} is the input weight to SNN unit i), and the weight distribution can be initialized as desired and permitted to change or be fixed during training. For the majority of our experiments, input weights were initialized with a sparse uniform distribution ($min = 0.0,\ max = 0.4$) and all weights were permitted to change during training while overall sparsity of connectivity was maintained (details in "Sparsity and rewiring" section).

4.3 Time

SNN trials, whether they do or do not involve training, all take place over time. We use a discrete time step $\delta t = 1ms$ for all our SNN work. Input (x) is given to the SNN ms-by-ms, and output (y) is read ms-by-ms. Dynamics such as all neurons' membrane potentials (v) and spikes (z) change over ms as well. The current ms time point is denoted as t, the next time point as $t + 1$, and so on.

4.4 ALIF model neuron

The spiking neuron model used for all units in the recurrent SNN is the adaptive leaky-integrate-and-fire (ALIF) model. ALIF units contain two hidden state variables – one for the membrane potential (aka voltage) v and one for the variable a which governs the adaptive spike threshold A. Together they determine whether a unit i spikes ($z_i^t = 1$) or does not spike ($z_i^t = 0$) at time point t.

Each unit's membrane potential evolves over time according to the equation:

$$v_i^{t+1} = \alpha(v_i^t - E_m) + \sum_{i \neq j} W_{ji}^{rec} z_j^t + \sum_i W_i^{input} x_i^{t+1} - z_i^t(v_{th} - E_m)$$

where the resting membrane potential E_m = -70. 6mV,

the baseline threshold v_{th} = -50. 4mV,

the decay factor $\alpha = e^{-\delta t/\tau_m}$, and $\tau_m = 20ms$ is the membrane time constant.

All synaptic connections and inputs in our model are current-based. At time point $t + 1$, SNN unit i receives recurrent spiking input from its presynaptic units j which just spiked (z_j^t, weighted according to W_{ji}^{rec}), and a subset also receive stimulus input (x_i^{t+1}, weighted according to W_i^{input}). This is summed across all presynaptic units and all stimulus input sources, so that the input current into unit i at time $t + 1$ is given as:

$$I_i^{t+1} = \sum_{i \neq j} W_{ji}^{rec} z_j^t + \sum_i W_i^{input} x_i^{t+1}.$$

Finally, the term $z_i^t(v_{th} - E_m)$ reduces a unit's membrane potential by a constant value after neuron i spikes (Bellec et al. 2020). $z_i^t$is further fixed to be 0 for a refractory period of 4ms following unit i spiking.

By the above equations, all units' membrane potentials evolve over time. When a unit's membrane potential exceeds the spike threshold, that unit will emit a spike. For ALIF neurons, the spike threshold is adaptive, meaning it also evolves over time.

The adaptive threshold increases following a spike and decays exponentially to the baseline threshold v_{th}. This can be described by the equations:

$A_i^t = v_{th} + \beta a_i^t$, where $\beta = 0.16$,

$z_i^t = H(v_i^t - A_i^t)$, where H is the Heaviside step function,

$a_i^{t+1} = \rho a_i^t + z_i^t$,

$\rho = e^{-\delta t/\tau_a}$, where $\tau_a = 100ms$, which can be altered to fit timescales relevant to the task of interest (Bellec et al. 2020).

Essentially, if a unit's membrane potential at time t exceeds its adaptive threshold at time t, the unit will emit a spike.

If $v_i^t > A_i^t$, then $z_i^t = 1$.

At the very beginning when $t = 0$, voltages are initialized with a random normal distribution ($\mu = -65$ mV, $\sigma = 5$ mV).

4.5 Structure

SNNs are initialized with neocortical structural properties, some of which continue to be enforced during training. Initial excitatory weights follow a long-tailed, log-normal distribution (Song et al. 2005), where $\mu = -0.64$ nA, $\sigma = .51$ nA, corresponding to a mean of 0.6005 nA and a variance of 0.1071 nA (Bojanek & Zhu et al. 2020). Inhibitory weights follow the same distribution but with values 10x stronger than excitatory. All units are sparsely and recurrently

connected; the precise probabilities of connection within and between e and i populations are taken from neocortical experiments (Billeh et al. 2020). Weight values and connection probabilities of e and i populations are permitted to change during training. However, overall sparsity is maintained (details in "Sparsity and rewiring" section).

4.6 Dynamics

In addition to structural constraints, models are also constrained to exhibit spiking dynamics that match neocortical dynamics during training. For example, spiking in neocortex is sparse, asynchronous, and near-critical (Brunel 2000; Renart et al. 2010; Zerlaut et al. 2019), and each of these features can be maintained in the SNNs' activity throughout training. Further details are provided in the "Training" section.

4.7 Output

Output from the main SNN will vary depending on the task. A subset of main SNN units sends weighted connections to the output. Thus, $W_i^{output} z_i^t$ is the output activation from SNN unit i at time t. Output connectivity is initialized using the same statistics as the recurrent SNN's connectivity, and sparsity is maintained during training in the same manner.

4.8 Training

SNNs are trained using backpropagation-through-time (BPTT) with modifications for spiking. The goal of training is to reduce the difference between the SNNs' output and the desired output, while still maintaining naturalistic dynamics and structure. This is achieved through iteratively

modifying the weights of the recurrent SNN. Input and output weights may also be permitted to change.

Since tasks are temporal, the task error (aka loss) at each timepoint is the mean squared error between the desired output (aka target) at time t and the SNNs' actual output (aka prediction) at time t.

$$E^{t}_{task} = MSE(y^{t}_{pred}, y^{t}_{target})$$

To maintain naturalistic dynamics, additional components are added to the total loss. For example, a target spike rate can be specified. The MSE deviation of the SNN's actual spike rate from that target rate is added to the loss (Zhu et al. 2020).

$$E_{rate} = MSE(rate_{SNN}, rate_{target})$$

$$E_{total} = E_{task} + E_{rate}$$

For all our experiments, we train networks in three ways: to minimize task loss, to minimize rate loss, and to minimize both task and rate loss. We report results separately for all three cases for each version of our SNNs.

4.9 Gradient descent with Adam optimizer

The goal of training is to reduce the total loss. To do so, we iteratively modify the weights in the network using a version of stochastic gradient descent.

The amount by which each weight should change in each iteration is determined by its gradient, which can be written as:

$$\frac{dE}{dW_{ij}} = \sum_t \frac{dE}{dz_i^t} * \frac{dz_i^t}{ds_i^t} * \frac{ds_i^t}{dW_{ij}}$$

The only new variable here is s_i^t, which is the hidden state of neuron i at time t, and includes membrane potential and adaptation. $s_i^t = [a_i^t, v_i^t]$.

Since the task and training take place over time, the total loss is summed over all time points t. Also, since the network is recurrently connected, the output, the hidden state of all units, and the spiking activity of all units at time t is dependent on all other units' states and activities at time $t - 1$, and at time $t - 2$, and so on to $t = 0$. Those previous states and activities in turn depend on the weights. Therefore, terms are recursively expanded over past time steps (details in Bellec et al. 2020).

We use the Adam optimizer to determine precisely how weights should change based on the gradients. Typically in stochastic gradient descent, each gradient is multiplied by a static learning rate to yield the value by which each weight should change. The Adam optimizer instead adapts the learning rate for every variable over the course of training (details in Kingma & Ba 2015).

4.10 Spike pseudo-derivative

The above is the standard form of BPTT for recurrent neural networks. However, because the units in our network spike, and spikes are not differentiable (recall that z_i^t is determined by the

Heaviside step function), we use a pseudo-derivative to replace the term $\frac{dz_i^t}{ds_i^t}$ (Huh & Sejnowski 2018, Bellec et al. 2020). Outside of the refractory period, the pseudo-derivative is defined as:

$\psi_i^t = \frac{1}{v_{th}} \gamma_{pd} max(0, 1 - |\frac{v_i^t - A_i^t}{v_{th}}|)$, where $\gamma_{pd} = 0.3$.

During the refractory period, $\psi_i^t = 0$.

4.11 Sparsity and rewiring

To maintain the overall level of connectivity in the SNN, all weights that are initialized as 0's ($W_{ij} = 0$ indicates that units i and j are "disconnected") will maintain their 0 values during training. All other weights can be updated via gradient descent. An exception occurs when an existing edge weight flips sign (e.g. an excitatory edge weight becomes negative) or becomes 0. In that case, that edge is set to 0 (e.g. is "pruned" away) and a new edge is randomly drawn from the pool of 0-valued edges. The new edge's weight value is drawn from the initial weight distribution. This process is akin to synaptic rewiring in neocortex (Bellec et al. 2018).

Input and output layer weights are also permitted to rewire during training, except in versions of the model which are explicitly noted in the results.

4.12 Task

Models were trained on a visual coherence change detection task. The model is presented with videos of 460 white drifting dots on a black background, with dots moving in one of 4 global directions (0°, 90°, 180°, 270°), and at two coherence levels (100% or 15%, which is the

percentage of dots that are moving together in the same direction). All dots move with speeds of 10 pixels per ms. Dots are randomly placed on the screen at the start of each video trial and are initialized with random remaining durations to their lifetimes (total lifetime of 1000 ms). Each time a dot reaches its max lifetime, it is removed, and a new dot is randomly drawn at a new location.

Each video trial has a total duration of 4080 ms. Half of all trials have a change in coherence which occurs at a random time between 500 and 3500 ms within the trial. The task is for the model to report the coherence level at all timepoints of the trial.

4.13 CNN front-end

To make the videos interpretable to the main SNN, we created and trained a CNN model of retina and thalamus as a preprocessor that turns videos into spike sequences (McIntosh et al. 2016). This CNN model was trained to report the velocity (x,y) of global motion in a combined dataset of drifting dot videos and black-and-white natural motion videos.

The CNN model is composed of three successive blocks of Conv3D, MaxPooling3D, and dropout layers, after which the output is flattened into a prediction of x velocity and a prediction of y velocity. We take the output activations of the 16 units in the next-to-last layer to indicate Poisson firing rates. Poisson spikes are newly generated for each trial according to these numbers, and then given as input to a subset of main SNN units.

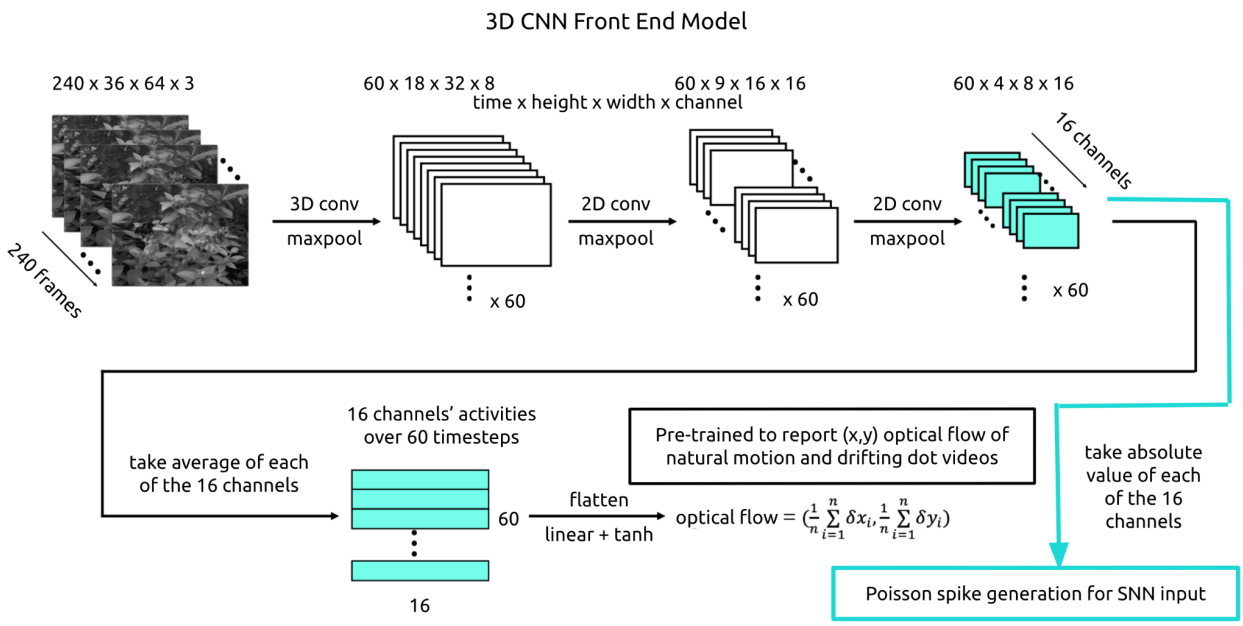

Figure S1: Model front-end architecture
To make videos interpretable to the main SNN, we built a 3D CNN front-end. The absolute value activations of the CNN's second-to-last layer were interpreted as firing rates from which Poisson spikes were generated to become main SNN input.

4.14 Training sequence

A single trial involves an input sequence and a target output sequence. The input sequence is composed of Poisson spiking activity from 16 input units. The target sequence is 0's and/or 1's - the label attached to each coherence level is swapped in different experiments.

SNNs are trained on a large set of these trials. There are a total of 600 unique trials, which are repeated and shuffled to create the desired total training set size. Experiments were run for a duration of 30 trials per batch x 10,000 total batch updates, which yields 300,000 total trials.

To reduce overfitting, in which the SNN becomes overly good at particular trials but not others, we accumulate E_{total} and $\frac{dE}{dW_{ij}}$ over a batch of 30 trials before updating weights. The process

then repeats for the next batch of 30 trials. Due to trial shuffling and repetition, each batch contains a unique set of trials. In this way, weight changes will improve the average performance over many different trials.

4.15 Software and hardware

All SNN training was completed using Python 3.8 or higher run on Nvidia GPUs with CUDA version 11.2 or higher.

4.16 Statistical analysis

To compare all distributions of SNN measures, such as naive and trained firing rates to the two coherence levels, naive and trained weight distributions of input, recurrent, and output layers, etc., we performed Kolmogorov-Smirnov two sample testing and reported both the distance between means and the p-value.

4.17 Weight changes during rate-only and task-only training

A similar pattern of weight changes occurred during training on only rate as during rate-and-task training. Recurrent weights changed the most (e→e: 0.009 ± 0.026 naive to 0.131 ± 0.358 trained; e→i: 0.012 ± 0.028 naive to 0.153 ± 0.413 trained; i→e: -0.150 ± 0.306 naive to 0.531 ± 1.655 trained; i→i: -0.171 ± 0.325 naive to -0.353 ± 1.457 trained; $p \approx 0.0$ for all naive and trained recurrent weight distributions). Inputs to excitatory units changed more than inputs to inhibitory units (in→e: 0.225 ± 0.502 naive to 0.313 ± 0.761 trained, $p = 1.827 \cdot 10^{-40}$; in→i: 0.184 ± 0.459 naive to 0.199 ± 0.555 trained, $p = 0.030$). Output layer weights did not change

(e→out: 0.009 ± 0.025 naive to 0.009 ± 0.025 trained, p = 1.0; i→out: -0.150 ± 0.307 naive to -0.150 ± 0.307 trained, p = 1.0).

After training on only task, recurrent weights also changed the most (e→e: 0.010 ± 0.025 naive to 0.126 ± 0.366 trained, p ≈ 0.0; e→i 0.012 ± 0.028 naive to 0.074 ± 0.335 trained, p =$2.200 \cdot 10^{-298}$; i→e -0.151 ± 0.309 naive to -0.656 ± 1.915 trained, p ≈ 0.0; i→i -0.170 ± 0.322 naive to -0.695 ± 1.923 trained, p = $1.319 \cdot 10^{-118}$), while input layer weights to inhibitory units (in→e: 0.313 ± 0.563 naive to 0.398 ± 0.744 trained, p = $3.137 \cdot 10^{-17}$; in→i: 0.272 ± 0.533 naive to 0.255 ± 0.576 trained, p = 0.077) and output layer weights (e→out: 0.010 ± 0.027 naive to 0.007 ± 0.022 trained, p = 0.202; i→out: -0.136 ± 0.286 naive to -0.129 ± 0.272 trained, p = 1.0) did not change.

ACKNOWLEDGEMENTS

We thank Wolfgang Maass, Franz Scherr, and Guillaume Bellec for providing us with invaluable initial guidance, unique insights, and code examples for us to begin training SNNs. We thank Elizabeth de Laittre, Harold Rockwell, Gabriella Wheeler Fox, and Tarek Jabri for their comments throughout these investigations which greatly improved our analyses. We thank Tarek Jabri, Josh Cruz, and Benton Girdler for their direct contributions to the codebase. We thank Stephanie Palmer, David Freedman, and John Maunsell for their guidance in the development and execution of this work.

DISCUSSION

We believe that trained SNNs demonstrate promise for advancing the interpretability of computation in neural network models and applying that understanding to neocortical networks. Building network models with a combination of realistic architectural features and optimizing them on ethologically meaningful tasks may enable new or improved theories of cortical computation. These models combine the strengths of task-trained non-spiking RNNs and non-task-trained SNNs and have the potential to become a frequent method of hypothesis generation and testing for future research into neocortical computation. To guide this work forward, I would like to discuss here the pipeline from greater interpretability (which SNNs make possible) to interpretation of those results in the context of neocortex as it relates to behavior, as well as various biofidelic architectural features that can be built into future models to study their contributions to neocortical computation.

Interpretability → interpretation

Models are constructed through a series of choices, and the results of the models are at least partially reflective of those choices. Sometimes this is positive: biofidelic choices, such as using spiking units and recurrent connectivity, can lead to greater understanding at a biofidelic, circuit-based level. However, we must admit to the other possibility of arbitrary choices leading to erroneous conclusions. Thus even if we can achieve thorough interpret*ability* of our models (see Introduction), how should we then *interpret* those results in the context of biological reality?

Adherence to experimental data, and the correct measurement of that data, can be crucial to the interpretation of results. Research has shown that numerical accuracy of models is critically

important, as small deviations on the level of individual neurons can lead to large differences in network-level behavior (Trensch et al. 2018). As more large-scale experimental datasets are carefully assembled (e.g. Billeh et al. 2020), the hope is that we can take the modeling results based on these data more seriously. But deciding when we have reached the correct level of detail to capture the mechanisms of computation will remain a challenge even when we have all the details in front of us (Borges 1946).

Luckily, finding out which details are important is an empirical matter. One metric that we can use is the ability of a model to capture experimental data that it was not built to reproduce. While prediction of neural activity and behavior should not be the end goal of modeling and does not itself constitute explanation (see Introduction), the breadth of a model's predictions can be a measure of its explanatory power. If a model can generalize and accurately predict neural data under a variety of conditions, that model is more likely to be doing so using the same underlying mechanisms that exist in biology. This is because good explanations tend to be parsimonious rather than requiring many different parts or differing implementations for various circumstances (Gauch 2002; Deutsch 2011).

Let us take, for example, a recent result in the modeling of grid cells in the entorhinal-hippocampal circuit (Schaeffer et al. 2022). The researchers demonstrated that deep-learning-based models of this circuit reveal less about principles and more about particular encoding and hyperparameter choices, in direct refutation of a series of studies in which training ANNs on path integration tasks led to the formation of grid-cell-like representations. Based on these results, the authors advised that instead of focusing on close predictions, modeling studies

should more fully explore the conditions in which model-brain correspondence does or does not emerge, and to consider whether these conditions align with biological constraints.

Based on these guidelines, a modeling study can be interpreted to hold some weight in the real world if the answers to the following three questions are "yes": 1) Does the model capture multiple facets of the biological system that it was not designed to capture? 2) Are the results robust to specific implementation choices in the model? 3) Are the conditions under which the results arise aligned with biological conditions?

For example, in chapter 1 we discovered that coordination of activity between groups of three units is necessary for stable biofidelic activity in sparse recurrent SNN models. Not only is this the case across connectivity schemes, demonstrating some robustness to implementation, it also explains–without intending to–why we observe clustering in both cortical synaptic connectivity and functional connectivity. In chapter 2 we discovered that to solve a binary report task, tuned input projections became separated by inhibitory and excitatory SNN targets, and inhibitory SNN units came to strongly inhibit units of the opposite tuning. This result captures how inhibitory intracortical connections are theorized to refine excitatory selectivity. These results did not arise in the case of training on rate alone, demonstrating that realistic task demands are indeed driving these changes. In addition, they did not arise when input and output layer weights were unconstrained in unrealistic ways. Although we do not perform those experiments here, an extension under good modeling principles is to explore if this pattern of tuning and connectivity generalizes to tasks that are not binary in nature but instead require choices across a spectrum of stimuli.

The accuracy and parsimony of models are related to the idea of stiff and sloppy dimensions in complex systems. Varying certain parameters or parameter combinations will lead to large differences in the behavior of a system–these are the stiff dimensions, while varying others only lead to negligible changes–these are the sloppy dimensions. For example, in SNN simulations of neocortex, the probabilities of connectivity between excitatory and inhibitory populations have been identified as stiff dimensions if realistic spiking dynamics are to be maintained (Jabri & MacLean 2022). The sloppiness of a complex system such as neocortex gives us room for forgiveness in our modeling work. Even if we do not have the precise experimental measurements to guide model parameters, we can still capture meaningful behaviors of the system because of its sloppiness (Gutenkunst et al. 2007).

Taking the flip side once more, building models with strict adherence to many details but improperly modeling stiff dimensions or other crucial features can lead to uncertain interpretations. For example, a large-scale, detailed columnar model of mouse V1 was constructed based on cortical data and trained to achieve a variety of visual tasks (Chen et al. 2022). Yet in the absence of data regarding how local network computations are communicated to downstream areas, the authors chose a very small population of units (30 excitatory neurons to signal each outcome–0.057% of the network size) to represent each possible outcome of a task decision. If a population produced the most spikes during a response window, that outcome was considered to be the network's decision. Moreover, picking just two neurons from the trained readout to produce outputs provided almost the same accuracy (Scherr & Maass 2021). This result begs the question of how much computational relevance we can truly ascribe to the rest of

the large network. The choice of output from a network can be crucial for its function, and one for which we currently lack sufficient experimental guidance.

Decoding

The manner in which one decodes output from a model is consequential for our understanding of network computation. One favored strategy has been linear decoding from a subpopulation of model units. For example, in (Yamins & DiCarlo 2014), the authors trained linear classifiers for different layers of their deep network and used them to evaluate model correspondence to neocortical data. However, trained linear readouts are computationally very powerful, and can therefore mask the computational contributions of the neural network model itself (Maass & Markram 2004). By using an untrained V1 model (Chen et al. 2022) and only training linear readouts, the average task accuracy was very high at 87%.

Through the experiments in chapter 2, we have seen firsthand that the choices made in model output have genuine consequences for the computational engagement of the recurrent model. In particular, we observed a tendency for solutions to change input and output layer weights more than recurrent weights when constraints were not implemented. This resulted in a network that behaved like a liquid state machine (LSM), a type of reservoir computer that makes use of linear readouts of the dynamics of recurrent units to achieve a wide variety of tasks. Research with LSMs has provided some insights into the information processing of recurrent networks (Buonomano & Maass 2009; Maass 2011), since they are well-designed to handle continuous time inputs and exhibit rich dynamics. However, they are also extremely challenging to understand. LSMs can replicate some brain functionality, but there is very little control or ability

to establish how or what computations are being performed by what dynamics. This difficulty, combined with our primary objective to understand how behaviorally-relevant learning takes place through changes in recurrent neocortical networks, meant that we were not satisfied with our model behaving like an LSM. We subsequently took steps to push the connectivity changes that support task computations into the recurrent layer. This allowed us to achieve structural changes in the recurrent network to support learning and a level of mechanistic understanding. As a whole, theorists working with trained recurrent networks should be deliberate in their decoding choices and assess whether the main recurrent network under study is truly responsible for improvements in task performance.

In addition to how output is decoded, how output targets are defined also influences the computational scheme which networks use to solve a task. Revisiting Schaeffer et al. 2022, the authors found that the emergence of grid-cell-like representations in models depended entirely on the encoding of the target. It remains a possibility that our choice to use ‘0’ and ‘1’ binary target labels may have driven the network to solve the task using the two dynamic regimes (low rate vs. high rate). When we experimented with an alternative winner-take-all scheme and a three-output (-1, 0, +1) scheme, the network still exhibited a separation of rates (data not shown). However, if labels were not numerical and instead categorical, similar to the implementation in Chen et al. 2011, the model may have reached a different solution strategy. Our implementations of a categorical output did not lead to good task performance, so we did not analyze those networks. Given the impact that output choices have on model computations, it is critical to turn to experimental data for guidance wherever possible. One approach is to study which neural signals are consequential for downstream activity and behavior in animals. A set of recent results

suggests that animals exhibit a preference to increments as opposed to decrements in cortical activity to guide behavior. Using a visual contrast change detection task and optogenetic stimulation in mouse V1, researchers found that perceptual reports were triggered exclusively by increments in V1 spike counts and impaired by decrements (Cone et al. 2020). The authors recapitulate this result in similarly-trained, low-rate RNNs. They hypothesize that increments are preferred for V1 decoding because the low baseline firing rates in neocortex create a limited coding range for spike decrements. Theoretical studies in SNNs have also found a related asymmetry in the robustness of networks to perturbations: SNNs were not affected by broad inhibitory perturbations, yet highly sensitive to excitatory perturbations, even to single model neurons (Calaim et al. 2022). The authors relate this result to the ability of animals to recognize very small excitatory perturbations.

Our recurrent SNNs were optimized to maintain low firing rates as well as solve a visual report task, making them a candidate for commentary on the above results. We did not directly study the impact of spike increments and decrements on task performance or robustness to perturbations. Removal of the low rate constraint (e.g. permitting the network to use more spikes to solve the task) also did not lead to strong improvements in task performance. However, we did observe that the report of the low-rate (‘0’ label) regime was much more stable than the report of the high-rate (‘1’ label) regime. In the high-rate regime in the trained network, excitatory input dominates. Thus increases in excitation in our models are more likely to cause changes in overall network dynamics and more strongly impact the output to downstream areas, consistent with prior modeling studies (Calaim et al. 2022). This greater sensitivity to excitation could have resulted in improved SNN performance to spike increments especially if we had chosen to use a

nonbinary task, as an output scheme with more possible choices would benefit from a larger dynamic range. Taken together, this set of results suggests that future biofidelic models should experiment with excitation-based decoding schemes.

Increasing realism in models

We incorporated biofidelity into our models based on evidence for some of the most crucial features underlying cortical computation. However, there are additional features that may play important roles, which we review below. These traits have not been left out of models for lack of recognition of their importance–rather, increasing biofidelity in task-trained networks is a substantial undertaking and it is unclear which level of realism is optimal. Facets of training often need to be modified in order to accommodate the new features. This can be more straightforward, such as modifying hyperparameters, which we did in our models (Zhu et al. 2020), or it can involve novel mathematical formulations to properly assign errors or prevent gradients from exploding or vanishing.

This second variety of modification is often outside the purview of neuroscientists. However, as neuromorphic computing with trained SNNs becomes more prevalent, we hope that computer scientists and statisticians can work to accommodate additional biofidelity. These features are likely to improve the function and efficiency of neuromorphic systems, as well as ultimately aid in our understanding of cortical computation. A few of these features include conductance-based synapses, the numerical scale of networks, and alternative plasticity rules. We would also like to advocate for increasing the ethological relevance and complexity of the tasks used for training.

Neuronal details

Changes in inhibitory connectivity were crucial to the ability of our SNNs to perform the coherence report task in chapter 2. We only used a generic inhibitory neuron in this work; however, it is known that different inhibitory interneuron classes have distinct signatures of activity and connectivity (Kepecs & Fishell 2014) and make unique computational contributions to behaviors such as visual contrast perception (Cone et al. 2019). Theoretical results also demonstrate that having a variety of neurons with different dynamics, such as distinct spike thresholds, is beneficial for information capacity especially under realistic, noisy conditions (Sharpee 2017). Given the computational importance of neuron subclasses, we hope that future work with trained SNNs will include multiple neuron types and thus further elucidate their unique computational roles. Such an undertaking is made possible due to the availability of large-scale datasets, and detailed modeling of distinct spiking neuron classes is already underway (Teeter et al. 2018; Billeh et al. 2020; Chen et al. 2022).

In line with increasing the level of detail in our models by building distinct neuron subclasses, we can also consider computations that happen within individual neurons. Recent studies have revealed that dendrites themselves perform computations, and (Payeur et al. 2019) delineated four classes of dendritic information processing: spatiotemporal filtering, information selection, information routing, and information multiplexing (Payeur et al. 2019, Yang et al. 2020, Beaulieu-Laroche et al. 2019, Stuart & Spruston 2015). Machine learning models have also begun to incorporate dendritic processing into their model units (Acharya et al. 2022, Chavlis & Poirazi 2021, Poirazi & Papoutsi 2020), and one study has used whole DNNs to model single

neurons (Beniaguev et al. 2021). The possible computational complexity of dendrites is worth exploring as one more step in mechanistic understanding of neocortical function.

Conductance-based synapses

The models employed in chapter 1 were built with conductance-based synapses, which are more biophysically grounded (Kuhn et al. 2004; Meffin et al. 2004) and more accurately capture the timescales of neuronal interactions and synaptic cooperativity in neocortex than current-based synapses. For example, only conductance-based models can reproduce the decrease in membrane input resistance when synaptic input is strong, as observed in intracellular recordings (Destexhe et al. 2003). When directly comparing matched conductance-based and current-based LIF networks, first-order statistics are comparable but second-order statistics diverge. Correlations between neurons and their modulation by the input was stronger in conductance-based models. As a result of these properties, spike train correlations carried more information about the input in conductance-based models (Cavallari et al. 2014). Along with features like recurrent sparse connectivity and adaptation, conductance-based synapses may be important for cortical computations that necessarily unfold over time.

However, a complication of conductance-based models is that they can only be approximated analytically (Rudolph-Lilith et al., 2012) and are especially challenging to work with in the context of task-optimization. Models with conductance-based synapses have been trained using methods such as STDP (Legenstein et al. 2008), but a solution has yet to be formulated for training them with global learning signals to support complex task performance. We are hopeful that a solution can be found in the near future.

Scale

It is generally accepted that animals with nervous systems of different relative sizes possess different capacities for behavior, even if the precise relationship is inconclusive (Roth & Dicke 2005). In the context of animals which have a neocortex, we are most interested in general principles of cortical computation rather than cross-organismal differences. We constructed our SNNs on the scale of 1000s (ch. 1) or 100s (ch. 2) of units to represent a limited area of neocortical circuitry, and to match the number of neurons we could record from layer 2/3 of mouse neocortex using 2-photon calcium imaging for direct comparisons. At such a scale, the model is readily tractable through the variety of analyses we have formerly performed on calcium imaging data.

However, this may be a more suitable approach for studying local circuit dynamics than for studying computation. It is possible that as neuronal networks increase in scale, their computational abilities may change drastically. The idea that the size of networks is consequential for computation is almost as old as neural network models themselves. Early work with Hopfield networks demonstrated that the number of possible stored patterns increases with the size of the network (Hopfield 1982; Amit et al. 1987, McEliece et al. 1987, Dotsenko et al. 1991). Adding hidden units to restricted Boltzmann machines leads strictly to improvement in model performance if inputs are discrete (Le Roux & Bengio 2008). In the case of deep networks, the addition of layers leads to more efficient representations (Hinton et al. 2006; Bengio et al. 2006; Bengio & LeCun 2007). As a further example, theoretical research has proven that shallow digital circuits are exponentially less efficient than deeper ones (Ajtai 1983; Håstad 1986; Allender 1996).

It is possible that similar computational principles of scale apply to neocortex. The number of units in our trained models falls far short of that of mouse visual cortex, as one example. Thus an area in which we can extend our modeling work–and the applicability of results to real neocortical computations–is through increasing the size of our SNNs. Building and training large-scale SNNs is an effort that others are already beginning. For example, a data-driven model of mouse V1 comprised of approximately 52,000 model neurons reproduced a number of mouse visual abilities, such as solving multiple temporal tasks and being robust to noise (Chen et al. 2022). However, the readout of this model was extremely sparse, which may have influenced training and results, as we discussed in a prior section. It will be useful to examine how scale and readout schemes interact, as well as the computational impacts of combinations of all biofidelic features we propose here.

Plasticity rules

The global plasticity rule we used in our SNNs (Bellec et al. 2020) is an extension of machine learning methods rather than a direct model of neocortical plasticity. We chose to use this rule because our main objective was to establish computations in our SNNs through synaptic changes, which we could then investigate; this method accomplished this goal. However, it is possible that training via backpropagation vs. biologically observed local plasticity will not produce the same variety of computations. There have been and continue to be efforts to contrast and discover correspondences between local, biologically plausible learning rules and global learning rules (Bengio et al. 2017, Gerstner et al. 2018, Shrestha et al. 2019, Zhang et al. 2020, Rosenbaum 2022, Shervani-Tabar & Rosenbaum 2022). These investigations can help guide the choice of plasticity rules in future studies with task-trained SNNs.

An additional consideration is that most animal behavior is likely encoded in the genome (Zador 2019). Therefore the changes that we observe in SNNs to support task performance capture both the "learning" that accumulates through evolutionary selection as well as learning through experience in an individual lifetime. This is not necessarily a concern for present or future work that focuses on the final computational solution, but researchers should note that, especially in future comparisons between trained SNNs and trained animal experiments, the naive state of the network model does not correspond to the naive state of the animal's brain.

We trained our models to minimize both task and rate loss at the same time. To better separate evolutionary/developmental learning and task-based learning, one could implement two phases of model optimization. For example, the model could be trained to reach desired dynamics before commencing training on the task. In addition, we could better relate rate-optimization to homeostatic cortical plasticity in future work. While homeostatic plasticity alone does not appear sufficient for learning realistic, time-varying tasks in recurrent SNNs (Zhu & Rosenbaum 2022), it is a widely observed variety of plasticity which pushes firing rates to baseline targets (Castillo et al., 2011; Vogels et al., 2011, 2013; Luz & Shamir, 2012; Hennequin et al., 2017; Schulz et al., 2021; Capogna et al., 2021). Using homeostatic inhibitory plasticity along with other plasticity rules may be an especially promising avenue to establish neocortical dynamics in conjunction with task learning in SNNs, especially given that changes in inhibitory connectivity were crucial to establishing computations in our models.

Task complexity

Discrepancies between model and brain may not be due to structural choices in the model itself, but rather due to improper choice of task. Just as we aim for biofidelity of model structure, we should aim for tasks that make realistic ethological demands on the model. It is possible that some of the auxiliary results we observed in our SNNs were due to the relative simplicity of the binary task.

In our models which lacked input and output layer constraints, the preferred solution to the binary coherence report task bypassed the necessity for synaptic changes in the recurrent layer. Machine learning research has demonstrated that it is possible to achieve robust task performance with a model architecture that dispenses with recurrence and instead relies entirely on an attention mechanism to draw direct dependencies between input and output (Vaswani et al. 2017). This result is related to our own discovery that, in the absence of input layer constraints, the SNN units which receive the strongest inputs also send the strongest outputs, thus effectively linking input and output directly and bypassing the need for computations dependent on recurrent circuitry.

Of course, barring monosynaptic reflexes, organisms in possession of a neocortex lack direct routes from sensory input to behavioral output. In the case of training on single tasks, it is sensible to dispense with a recurrent network and rely solely on attention mechanisms. However, the breadth of demands that the natural world places on an animal likely requires a degree of flexibility that such an architecture would fail to confer. As tasks increase in their complexity or likeness to real-world situations, it may no longer be easiest–or even at all feasible–to reach a

solution that does not rely on recurrent network dynamics. Instead, the power and computational benefits of recurrence (see Introduction) may need to be fully utilized in order to solve the tasks.

Motifs

Under greater task demands in which recurrence is necessary, additional neocortical features may display beneficial roles as well. For example, the motifs of functional connectivity which we characterize in chapter 1 could be crucial players in neocortical computation. While we did not observe increases in motif clustering over training in our models in chapter 2, we also did not observe decreases in motifs. Since the presence of higher order motifs are indicative of synaptic cooperativity (Chambers & MacLean 2016), and the stronger synaptic weights that arose during SNN training–and which should also arise in neocortex through Hebbian plasticity–require less cooperativity (Bojanek & Zhu et al. 2020), we might expect learning to correspond to a decrease in motifs. However, several lines of evidence suggest the contrary–that motif counts should in fact increase during computation. For example, motifs may directly correspond to logic gates (Curto et al. 2019); three-neuron motifs can improve coding (Cayco-Gajic et al. 2015; Shi et al. 2015) and enhance perceptual accuracy and the prediction of responses in visual cortex (Dechery et al. 2018; Shahidi et al. 2019; Kotekal & MacLean 2020). Furthermore, networks which process information, such as the ISCAS 89 benchmark circuit, gene transcription networks, and neuronal synaptic networks in *C. elegans*, are all characterized by robust presence of several higher order motifs (Milo et al. 2002). Together, these results suggest that computation in neocortex and robust performance on more challenging tasks in SNNs–especially future SNNs built with conductance-based synapses that require increased cooperativity–will rely on the formation of higher order functional motifs over training.

In conclusion

Let us return to our starting question: “why is it that, through learning, neocortical networks become capable of computations?” Through this work, we have begun exploring “why” in spiking neural network models of neocortex. We identified SNNs as a tool with great promise due to their potential for interpretability and more direct interpretation in the context of neocortex. As demonstrated through a rich literature of structure-function studies with SNNs, which we contribute to in chapter 1, SNNs can reveal mechanistic truths as to why they exhibit certain behaviors. Their existing biofidelic traits, such as sparse recurrent connectivity and spiking nature, make their computational strategies more readily applicable to strategies that may be employed by neocortex. In chapter 2, we studied a specific task and discovered that the model’s computational solution took the form of changes in inhibitory recurrent connectivity and tuned input connectivity. This result reflects the importance of inhibitory plasticity in refining inputs, defining selectivity, and shaping computations in neocortex. We are optimistic that future work with SNNs will continue to advance our understanding of computation in neocortex through broadening the scope of tasks and the biofidelity of the models. Some of these efforts are already rapidly underway, and we look forward to future findings.

REFERENCES

Abbott, L. F., & Nelson, S. B. (2000). Synaptic plasticity: taming the beast. Nature neuroscience, 3(11), 1178-1183. https://www.nature.com/articles/nn1100_1178.

Acharya, J., Basu, A., Legenstein, R., Limbacher, T., Poirazi, P., & Wu, X. (2022). Dendritic computing: branching deeper into machine learning. Neuroscience, 489, 275-289. https://doi.org/10.1016/j.neuroscience.2021.10.001.

Ahmad, S., & Scheinkman, L. (2019). How can we be so dense? The benefits of using highly sparse representations. arXiv preprint arXiv:1903.11257. https://arxiv.org/abs/1903.11257.

Ajtai, M. (1983). $\sum$ 11-formulae on finite structures. Annals of pure and applied logic, 24(1), 1-48. https://doi.org/10.1016/0168-0072(83)90038-6.

Alitto, H. J., & Dan, Y. (2010). Function of inhibition in visual cortical processing. Current opinion in neurobiology, 20(3), 340-346. https://doi.org/10.1016/j.conb.2010.02.012.

Allender, E. (1996). Circuit complexity before the dawn of the new millennium. In Foundations of Software Technology and Theoretical Computer Science: 16th Conference Hyderabad, India, December 18–20, 1996 Proceedings 16 (pp. 1-18). Springer Berlin Heidelberg. https://link.springer.com/chapter/10.1007/3-540-62034-6_33.

Amit, D. J., Gutfreund, H., & Sompolinsky, H. (1987). Statistical mechanics of neural networks near saturation. Annals of physics, 173(1), 30-67. https://doi.org/10.1016/0003-4916(87)90092-3.

Ayuso-Martinez, A., Casanueva-Morato, D., Dominguez-Morales, J. P., Jimenez-Fernandez, A., & Jimenez-Moreno, G. (2022, July). Spike-based building blocks for performing logic operations using Spiking Neural Networks on SpiNNaker. In 2022 International Joint Conference on Neural Networks (IJCNN) (pp. 1-9). IEEE.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K. R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one, 10(7), e0130140. https://doi.org/10.1371/journal.pone.0130140.

Baggio, G. (2018). Meaning in the brain. MIT Press.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473. https://arxiv.org/abs/1409.0473.

Baker, B., Lansdell, B., & Kording, K. (2021). A philosophical understanding of representation for neuroscience. arXiv preprint arXiv:2102.06592. https://arxiv.org/abs/2102.06592.

Banino, A., Barry, C., Uria, B., Blundell, C., Lillicrap, T., Mirowski, P., ... & Kumaran, D. (2018). Vector-based navigation using grid-like representations in artificial agents. Nature, 557(7705), 429-433. https://www.nature.com/articles/s41586-018-0102-6.

Barbour, B., Brunel, N., Hakim, V., & Nadal, J. P. (2007). What can we learn from synaptic weight distributions?. TRENDS in Neurosciences, 30(12), 622-629. https://www.sciencedirect.com/science/article/pii/S0166223607002615.

Barrett, D. G., Morcos, A. S., & Macke, J. H. (2019). Analyzing biological and artificial neural networks: challenges with opportunities for synergy?. Current opinion in neurobiology, 55, 55-64. https://doi.org/10.1016/j.conb.2019.01.007.

Barzel, B., & Barabási, A. L. (2013). Universality in network dynamics. Nature physics, 9(10), 673-681. https://www.nature.com/articles/nphys2741.

Bashivan, P., Kar, K., & DiCarlo, J. J. (2019). Neural population control via deep image synthesis. Science, 364(6439), eaav9436. https://www.science.org/doi/full/10.1126/science.aav9436.

Beaulieu-Laroche, L., Toloza, E. H., Brown, N. J., & Harnett, M. T. (2019). Widespread and highly correlated somato-dendritic activity in cortical layer 5 neurons. Neuron, 103(2), 235-241. https://doi.org/10.1016/j.neuron.2019.05.014.

Beggs, J. M., & Plenz, D. (2003). Neuronal avalanches in neocortical circuits. Journal of neuroscience, 23(35), 11167-11177. https://doi.org/10.1523/JNEUROSCI.23-35-11167.2003.

Bellec G, Kappel D, Maass W, Legenstein R. Deep rewiring: Training very sparse deep networks. arXiv:1711.05136v5 [cs.NE] [Preprint]. 2018. https://arxiv.org/abs/1711.05136v5.

Bellec, G., Scherr, F., Subramoney, A., Hajek, E., Salaj, D., Legenstein, R., & Maass, W. (2020). A solution to the learning dilemma for recurrent networks of spiking neurons. Nature communications, 11(1), 3625. https://www.nature.com/articles/s41467-020-17236-y.

Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2006). Greedy layer-wise training of deep networks. Advances in neural information processing systems, 19. https://proceedings.neurips.cc/paper/2006/hash/5da713a690c067105aeb2fae32403405-Abstract.html.

Bengio, Y., & LeCun, Y. (2007). Scaling learning algorithms towards AI. Large-scale kernel machines, 34(5), 1-41. https://pdfs.semanticscholar.org/f01e/080777b59d6978e412ded8995edabbaa62f0.pdf.

Bengio, Y., Mesnard, T., Fischer, A., Zhang, S., & Wu, Y. (2017). STDP-compatible approximation of backpropagation in an energy-based model. Neural computation, 29(3), 555-577. https://ieeexplore.ieee.org/abstract/document/7864516.

Beniaguev, D., Segev, I., & London, M. (2021). Single cortical neurons as deep artificial neural networks. Neuron, 109(17), 2727-2739. https://doi.org/10.1016/j.neuron.2021.07.002.

Bertschinger, N., & Natschläger, T. (2004). Real-time computation at the edge of chaos in recurrent neural networks. Neural computation, 16(7), 1413-1436. https://ieeexplore.ieee.org/abstract/document/6790172.

Billeh, Y. N., Cai, B., Gratiy, S. L., Dai, K., Iyer, R., Gouwens, N. W., ... & Arkhipov, A. (2020). Systematic integration of structural and functional data into multi-scale models of mouse primary visual cortex. Neuron, 106(3), 388-403. https://doi.org/10.1016/j.neuron.2020.01.040.

Bojanek, K.*, Zhu, Y.*, MacLean, J. N. (2020). Cyclic transitions between higher order motifs underlie sustained asynchronous spiking in sparse recurrent networks. PLOS Comput Biol, 16(9), e1007409. https://doi.org/10.1371/journal.pcbi.1007409. *co-first-authors.

Borges, J. L. (1946). Del rigor en la ciencia.

Bos, H., Oswald, A. M., & Doiron, B. (2020). Untangling stability and gain modulation in cortical circuits with multiple interneuron classes. bioRxiv, 2020-06. https://www.biorxiv.org/content/10.1101/2020.06.15.148114v2.abstract.

Brette, R. (2015). Philosophy of the spike: rate-based vs. spike-based theories of the brain. Frontiers in systems neuroscience, 151. https://doi.org/10.3389/fnsys.2015.00151.

Brette, R., & Gerstner, W. (2005). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. Journal of neurophysiology, 94(5), 3637-3642. https://journals.physiology.org/doi/full/10.1152/jn.00686.2005.

Brunel, N. (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. Journal of computational neuroscience, 8, 183-208. https://link.springer.com/article/10.1023/A:1008925309027.

Brunel, N. (2016). Is cortical connectivity optimized for storing information?. Nature neuroscience, 19(5), 749-755. https://www.nature.com/articles/nn.4286.

Brunel, N., Hakim, V., Isope, P., Nadal, J. P., & Barbour, B. (2004). Optimal information storage and the distribution of synaptic weights: perceptron versus Purkinje cell. Neuron, 43(5), 745-757. https://doi.org/10.1016/j.neuron.2004.08.023.

Bu, T., Fang, W., Ding, J., Dai, P., Yu, Z., & Huang, T. (2023). Optimal ANN-SNN conversion for high-accuracy and ultra-low-latency spiking neural networks. arXiv preprint arXiv:2303.04347. https://arxiv.org/abs/2303.04347.

Buesing, L., Bill, J., Nessler, B., & Maass, W. (2011). Neural dynamics as sampling: a model for stochastic computation in recurrent networks of spiking neurons. PLoS computational biology, 7(11), e1002211. https://doi.org/10.1371/journal.pcbi.1002211.

Buonomano, D. V., & Maass, W. (2009). State-dependent computations: spatiotemporal processing in cortical networks. Nature Reviews Neuroscience, 10(2), 113-125. https://elifesciences.org/articles/73276.

Calaim, N., Dehmelt, F. A., Gonçalves, P. J., & Machens, C. K. (2022). The geometry of robustness in spiking neural networks. Elife, 11, e73276. https://doi.org/10.7554/eLife.73276.

Campagnola, L., Seeman, S. C., Chartrand, T., Kim, L., Hoggarth, A., Gamlin, C., ... & Jarsky, T. (2022). Local connectivity and synaptic dynamics in mouse and human neocortex. Science, 375(6585), eabj5861. https://doi.org/10.1126/science.abj5861.

Capogna, M., Castillo, P. E., & Maffei, A. (2021). The ins and outs of inhibitory synaptic plasticity: Neuron types, molecular mechanisms and functional roles. European Journal of Neuroscience, 54(8), 6882-6901. https://doi.org/10.1111/ejn.14907.

Carandini, M., & Ringach, D. L. (1997). Predictions of a recurrent model of orientation selectivity. Vision research, 37(21), 3061-3071. https://www.sciencedirect.com/science/article/pii/S0042698997001004.

Castillo, P. E., Chiu, C. Q., & Carroll, R. C. (2011). Long-term plasticity at inhibitory synapses. Current opinion in neurobiology, 21(2), 328-338. https://www.sciencedirect.com/science/article/pii/S0959438811000213.

Cavallari, S., Panzeri, S., & Mazzoni, A. (2014). Comparison of the dynamics of neural interactions between current-based and conductance-based integrate-and-fire recurrent networks. Frontiers in neural circuits, 8, 12. https://doi.org/10.3389/fncir.2014.00012.

Cayco-Gajic, N. A., Zylberberg, J., & Shea-Brown, E. (2015). Triplet correlations among similarly tuned cells impact population coding. Frontiers in computational neuroscience, 9, 57. https://www.frontiersin.org/articles/10.3389/fncom.2015.00057/full.

Chalk, M., Marre, O., & Tkačik, G. (2018). Toward a unified theory of efficient, predictive, and sparse coding. Proceedings of the National Academy of Sciences, 115(1), 186-191. https://doi.org/10.1073/pnas.1711114115.

Chambers, B., & MacLean, J. N. (2016). Higher-order synaptic interactions coordinate dynamics in recurrent networks. PLoS computational biology, 12(8), e1005078. https://doi.org/10.1371/journal.pcbi.1005078.

Chambers, B., Levy, M., Dechery, J. B., & MacLean, J. N. (2018). Ensemble stacking mitigates biases in inference of synaptic connectivity. Network Neuroscience, 2(1), 60-85. https://doi.org/10.1162/NETN_a_00032.

Chapeton, J., Fares, T., LaSota, D., & Stepanyants, A. (2012). Efficient associative memory storage in cortical circuits of inhibitory and excitatory neurons. Proceedings of the National Academy of Sciences, 109(51), E3614-E3622. https://doi.org/10.1073/pnas.1211467109.

Chavlis, S., & Poirazi, P. (2021). Drawing inspiration from biological dendrites to empower artificial neural networks. Current opinion in neurobiology, 70, 1-10. https://doi.org/10.1016/j.conb.2021.04.007.

Chen, G., Scherr, F., & Maass, W. (2022). A data-based large-scale model for primary visual cortex enables brain-like robust and versatile visual processing. Science Advances, 8(44), eabq7592. https://www.science.org/doi/full/10.1126/sciadv.abq7592.

Chu, D. (2023). Information theoretical properties of a spiking neuron trained with Hebbian and STDP learning rules. Natural Computing, 1-19. https://link.springer.com/article/10.1007/s11047-022-09939-6.

Chung, S., & Abbott, L. F. (2021). Neural population geometry: An approach for understanding biological and artificial neural networks. Current opinion in neurobiology, 70, 137-144. https://doi.org/10.1016/j.conb.2021.10.010.

Churchland, P. S., & Grush, R. (1999). Computation and the brain. The MIT encyclopedia of cognitive sciences, 155-158.

Churchland, P. S., & Sejnowski, T. J. (1992). The computational brain. MIT press.

Clopath, C., Büsing, L., Vasilaki, E., & Gerstner, W. (2010). Connectivity reflects coding: a model of voltage-based STDP with homeostasis. Nature neuroscience, 13(3), 344-352. https://www.nature.com/articles/nn.2479.

Clopath, C., & Brunel, N. (2013). Optimal properties of analog perceptrons with excitatory weights. PLoS computational biology, 9(2), e1002919. https://doi.org/10.1371/journal.pcbi.1002919.

Clopath, C., Nadal, J.P. & Brunel, N. (2012). Storage of correlated patterns in standard and bistable Purkinje cell models. PLoS Comput. Biol. 8, e1002448. https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1002448.

Cohen, U., Chung, S., Lee, D. D., & Sompolinsky, H. (2020). Separability and geometry of object manifolds in deep neural networks. Nature communications, 11(1), 746. https://www.nature.com/articles/s41467-020-14578-5.

Cone, J. J., Scantlen, M. D., Histed, M. H., & Maunsell, J. H. (2019). Different inhibitory interneuron cell classes make distinct contributions to visual contrast perception. Eneuro, 6(1). https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6414440/.

Cone, J. J., Bade, M. L., Masse, N. Y., Page, E. A., Freedman, D. J., & Maunsell, J. H. (2020). Mice preferentially use increases in cerebral cortex spiking to detect changes in visual stimuli. Journal of Neuroscience, 40(41), 7902-7920. https://www.jneurosci.org/content/40/41/7902.abstract.

Cooper, S. B. (2013). What Makes A Computation Unconventional?. Computing Nature: Turing Centenary Perspective, 255-269. https://link.springer.com/chapter/10.1007/978-3-642-37225-4_17.

Curto, C., Langdon, C., & Morrison, K. (2019). Robust motifs of threshold-linear networks. arXiv preprint arXiv:1902.10270. https://arxiv.org/abs/1902.10270.

Dahmen, D., Recanatesi, S., Ocker, G. K., Jia, X., Helias, M., & Shea-Brown, E. (2020). Strong coupling and local control of dimensionality across brain areas. Biorxiv, 2020-11. https://www.biorxiv.org/content/10.1101/2020.11.02.365072v4.abstract.

Davies M, Wild A, Orchard G, Sandamirskaya Y, Fonseca Guerra GA, Joshi P, Plank P, Risbud S. Advancing neuromorphic computing with Loihi: a survey of results and outlook. Proc. IEEE. 2021;109(5): 911-934. https://doi.org/10.1109/JPROC.2021.3067593.

Day-Cooney, J., Cone, J. J., & Maunsell, J. H. (2022). Perceptual weighting of V1 spikes revealed by optogenetic white noise stimulation. Journal of Neuroscience, 42(15), 3122-3132. https://www.jneurosci.org/content/42/15/3122.abstract.

DeAngelis, G. C., Robson, J. G., Ohzawa, I., & Freeman, R. D. (1992). Organization of suppression in receptive fields of neurons in cat visual cortex. Journal of Neurophysiology, 68(1), 144-163. https://doi.org/10.1152/jn.1992.68.1.144.

deCharms, R. C., & Merzenich, M. M. (1996). Primary cortical representation of sounds by the coordination of action-potential timing. Nature, 381(6583), 610-613. https://www.nature.com/articles/381610a0.

Dechery, J. B., & MacLean, J. N. (2018). Functional triplet motifs underlie accurate predictions of single-trial responses in populations of tuned and untuned V1 neurons. PLoS computational biology, 14(5), e1006153. https://doi.org/10.1371/journal.pcbi.1006153.

Deco, G., & Schürmann, B. (1999). Spatiotemporal coding in the cortex: information flow-based learning in spiking neural networks. Neural computation, 11(4), 919-934. https://ieeexplore.ieee.org/abstract/document/6790547.

DePasquale, B., Sussillo, D., Abbott, L. F., & Churchland, M. M. (2023). The centrality of population-level factors to network computation is demonstrated by a versatile approach for training spiking networks. Neuron. https://doi.org/10.1016/j.neuron.2022.12.007.

Destexhe, A., Rudolph, M., & Paré, D. (2003). The high-conductance state of neocortical neurons in vivo. Nature reviews neuroscience, 4(9), 739-751. https://www.nature.com/articles/nrn1198.

Deutsch, D. (1998). The fabric of reality. Penguin UK.

Deutsch, D. (2011). The beginning of infinity: Explanations that transform the world. Penguin UK.

Dotsenko, V. S., Yarunin, N. D., & Dorotheyev, E. A. (1991). Statistical mechanics of Hopfield-like neural networks with modified interactions. Journal of Physics A: Mathematical and General, 24(10), 2419. https://iopscience.iop.org/article/10.1088/0305-4470/24/10/026/meta.

Douglas, R. M., Neve, A., Quittenbaum, J. P., Alam, N. M., & Prusky, G. T. (2006). Perception of visual motion coherence by rats and mice. Vision research, 46(18), 2842-2847. https://doi.org/10.1016/j.visres.2006.02.025.

Duncker, L., Driscoll, L., Shenoy, K. V., Sahani, M., & Sussillo, D. (2020). Organizing recurrent network dynamics by task-computation to enable continual learning. Advances in neural information processing systems, 33, 14387-14397. https://proceedings.neurips.cc/paper/2020/hash/a576eafbce762079f7d1f77fca1c5cc2-Abstract.html.

Ecker, A. S., Berens, P., Keliris, G. A., Bethge, M., Logothetis, N. K., & Tolias, A. S. (2010). Decorrelated neuronal firing in cortical microcircuits. science, 327(5965), 584-587. https://www.science.org/doi/full/10.1126/science.1179867.

Eliasmith, C., & Trujillo, O. (2014). The use and abuse of large-scale brain models. Current opinion in neurobiology, 25, 1-6. https://doi.org/10.1016/j.conb.2013.09.009.

Eysel, U. T., Crook, J. M., & Machemer, H. F. (1990). GABA-induced remote inactivation reveals cross-orientation inhibition in the cat striate cortex. Experimental Brain Research, 80, 626-630. https://link.springer.com/article/10.1007/BF00228003.

Fagiolo, G. (2007). Clustering in complex directed networks. Physical Review E, 76(2), 026107. https://journals.aps.org/pre/abstract/10.1103/PhysRevE.76.026107.

Ferster, D., & Miller, K. D. (2000). Neural mechanisms of orientation selectivity in the visual cortex. Annual review of neuroscience, 23(1), 441-471. https://doi.org/10.1146/annurev.neuro.23.1.441.

Flint, J., Greenspan, R. J., & Kendler, K. S. (2020). How Genes Influence Behavior 2e. Oxford University Press.

Ganmor, E., Segev, R., & Schneidman, E. (2015). A thesaurus for a neural population code. Elife, 4, e06134. https://doi.org/10.7554/eLife.06134.

Gauch Jr, H. (2002). PARSIMONY AND EFFICIENCY. In Scientific Method in Practice (pp. 269-326). Cambridge: Cambridge University Press. https://doi.org/10.1017/CBO9780511815034.010.

Gerstner, W., Lehmann, M., Liakoni, V., Corneil, D., & Brea, J. (2018). Eligibility traces and plasticity on behavioral time scales: experimental support of neohebbian three-factor learning

rules. Frontiers in neural circuits, 12, 53. https://www.frontiersin.org/articles/10.3389/fncir.2018.00053/full.

Goehring, T., Keshavarzi, M., Carlyon, R. P., & Moore, B. C. (2019). Using recurrent neural networks to improve the perception of speech in non-stationary noise by people with cochlear implants. The Journal of the Acoustical Society of America, 146(1), 705-718. https://doi.org/10.1121/1.5119226.

Graves, A. (2013). Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850. https://arxiv.org/abs/1308.0850.

Griffith, J. S., & Horn, G. (1966). An analysis of spontaneous impulse activity of units in the striate cortex of unrestrained cats. The Journal of Physiology, 186(3), 516. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1395914/.

Gutenkunst, R. N., Waterfall, J. J., Casey, F. P., Brown, K. S., Myers, C. R., & Sethna, J. P. (2007). Universally sloppy parameter sensitivities in systems biology models. PLOS Computational Biology, 3(10), e189. https://doi.org/10.1371/journal.pcbi.0030189.

Haider, B., Duque, A., Hasenstaub, A. R., & McCormick, D. A. (2006). Neocortical network activity in vivo is generated through a dynamic balance of excitation and inhibition. Journal of Neuroscience, 26(17), 4535-4545. https://doi.org/10.1523/JNEUROSCI.5297-05.2006.

Haldeman, C., & Beggs, J. M. (2005). Critical branching captures activity in living neural networks and maximizes the number of metastable states. Physical review letters, 94(5), 058101. https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.94.058101.

Hasenstaub, A., Otte, S., Callaway, E., & Sejnowski, T. J. (2010). Metabolic cost as a unifying principle governing neuronal biophysics. Proceedings of the National Academy of Sciences, 107(27), 12329-12334. https://www.pnas.org/doi/abs/10.1073/pnas.0914886107.

Håstad, J. (1986). Computational limitations for small depth circuits (Doctoral dissertation, Massachusetts Institute of Technology).

Hebb, D. O. (1949). *The Organization of Behavior: A Neuropsychological Theory*. Psychology Press.

Hennequin, G., Agnes, E. J., & Vogels, T. P. (2017). Inhibitory plasticity: balance, control, and codependence. Annual review of neuroscience, 40, 557-579. https://doi.org/10.1146/annurev-neuro-072116-031005.

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. Neural computation, 18(7), 1527-1554. https://doi.org/10.1162/neco.2006.18.7.1527.

Histed, M. H., Ni, A. M., & Maunsell, J. H. (2013). Insights into cortical mechanisms of behavior from microstimulation experiments. Progress in neurobiology, 103, 115-130. https://doi.org/10.1016/j.pneurobio.2012.01.006.

Hlinka, J., Hartman, D., & Paluš, M. (2012). Small-world topology of functional connectivity in randomly connected dynamical systems. Chaos: An Interdisciplinary Journal of Nonlinear Science, 22(3), 033107. https://doi.org/10.1063/1.4732541.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. Proceedings of the national academy of sciences, 79(8), 2554-2558. https://www.pnas.org/doi/abs/10.1073/pnas.79.8.2554.

Houghton, C., & Kreuz, T. (2012). On the efficient calculation of van Rossum distances. Network: Computation in Neural Systems, 23(1-2), 48-58. https://doi.org/10.3109/0954898X.2012.673048.

Hromádka, T., DeWeese, M. R., & Zador, A. M. (2008). Sparse representation of sounds in the unanesthetized auditory cortex. PLoS biology, 6(1), e16. https://doi.org/10.1371/journal.pbio.0060016.

Hu, Y., Brunton, S. L., Cain, N., Mihalas, S., Kutz, J. N., & Shea-Brown, E. (2018). Feedback through graph motifs relates structure and function in complex networks. Physical Review E, 98(6), 062312. https://journals.aps.org/pre/abstract/10.1103/PhysRevE.98.062312.

Hu, Y., Trousdale, J., Josić, K., & Shea-Brown, E. (2013). Motif statistics and spike correlations in neuronal networks. Journal of Statistical Mechanics: Theory and Experiment, 2013(03), P03012. https://doi.org/10.1088/1742-5468/2013/03/P03012.

Huh, D., & Sejnowski, T. J. (2018). Gradient descent for spiking neural networks. Advances in neural information processing systems, 31. http://papers.nips.cc/paper/7417-gradient-descent-for-spiking-neural-networks.

Jhuang, H., Serre, T., Wolf, L., & Poggio, T. (2007, October). A biologically inspired system for action recognition. In 2007 IEEE 11th international conference on computer vision (pp. 1-8). Ieee. https://ieeexplore.ieee.org/abstract/document/4408988.

Jovanović, S., & Rotter, S. (2016). Interplay between graph topology and correlations of third order in spiking neuronal networks. PLoS computational biology, 12(6), e1004963. https://doi.org/10.1371/journal.pcbi.1004963.

Kar, K., Kornblith, S., & Fedorenko, E. (2022). Interpretability of artificial neural network models in artificial intelligence versus neuroscience. Nature Machine Intelligence, 1-3. https://www.nature.com/articles/s42256-022-00592-3.

Karimipanah, Y., Ma, Z., & Wessel, R. (2016). New hallmarks of criticality in recurrent neural networks. arXiv preprint arXiv:1610.01217. https://arxiv.org/pdf/1610.01217.pdf.

Karpathy, A., Johnson, J., & Fei-Fei, L. (2015). Visualizing and understanding recurrent networks. arXiv preprint arXiv:1506.02078. https://arxiv.org/abs/1506.02078.

Katzner, S., Busse, L., & Carandini, M. (2011). GABAA inhibition controls response gain in visual cortex. Journal of Neuroscience, 31(16), 5931-5941. https://www.jneurosci.org/content/31/16/5931.short.

Kepecs, A., & Fishell, G. (2014). Interneuron cell types are fit to function. Nature, 505(7483), 318-326. https://www.nature.com/articles/nature12983.

Kietzmann, T. C., McClure, P., & Kriegeskorte, N. (2017). Deep neural networks in computational neuroscience. BioRxiv, 133504. https://www.biorxiv.org/content/10.1101/133504v2.abstract.

Kietzmann, T. C., Spoerer, C. J., Sörensen, L. K., Cichy, R. M., Hauk, O., & Kriegeskorte, N. (2019). Recurrence is required to capture the representational dynamics of the human visual system. Proceedings of the National Academy of Sciences, 116(43), 21854-21863. https://doi.org/10.1073/pnas.1905544116.

Kim, Y., & Panda, P. (2021). Visual explanations from spiking neural networks using inter-spike intervals. Scientific reports, 11(1), 19037. https://www.nature.com/articles/s41598-021-98448-0.

Kirkels, L. A. M. H., Zhang, W., Havenith, M. N., Tiesinga, P., Glennon, J., Van Wezel, R. J. A., & Duijnhouwer, J. (2018). The opto-locomotor reflex as a tool to measure sensitivity to moving random dot patterns in mice. Scientific reports, 8(1), 1-9. https://www.nature.com/articles/s41598-018-25844-4.

Koch, K. W., & Fuster, J. M. (1989). Unit activity in monkey parietal cortex related to haptic perception and temporary memory. Experimental Brain Research, 76, 292-306. https://link.springer.com/article/10.1007/BF00247889.

Kohn A, Coen-Cagli R, Kanitscheider I, Pouget A. Correlations and neuronal population information. Annu Rev Neurosci. 2016;39: 237-256. https://doi.org/10.1146/annurev-neuro-070815-013851.

Kornblith, S., Norouzi, M., Lee, H., & Hinton, G. (2019, May). Similarity of neural network representations revisited. In International Conference on Machine Learning (pp. 3519-3529). PMLR. http://proceedings.mlr.press/v97/kornblith19a.html.

Kotekal, S., & MacLean, J. N. (2020). Recurrent interactions can explain the variance in single trial responses. PLoS computational biology, 16(1), e1007591. https://doi.org/10.1371/journal.pcbi.1007591.

Koulakov, A. A., Hromádka, T., & Zador, A. M. (2009). Correlated connectivity and the distribution of firing rates in the neocortex. Journal of Neuroscience, 29(12), 3685-3694. https://doi.org/10.1523/JNEUROSCI.4500-08.2009.

Kriener, B., Enger, H., Tetzlaff, T., Plesser, H. E., Gewaltig, M. O., & Einevoll, G. T. (2014). Dynamics of self-sustained asynchronous-irregular activity in random networks of spiking neurons with strong synapses. Frontiers in computational neuroscience, 8, 136. https://doi.org/10.3389/fncom.2014.00136.

Kuhn, A., Aertsen, A., and Rotter, S. (2004). Neuronal integration of synaptic input in the fluctuation-driven regime. Journal of neuroscience, 24, 2345–2356. https://doi.org/10.1523/JNEUROSCI.3349-03.2004.

Kumar, A., Rotter, S., & Aertsen, A. (2008). Conditions for propagating synchronous spiking and asynchronous firing rates in a cortical network model. Journal of neuroscience, 28(20), 5268-5280. https://doi.org/10.1523/JNEUROSCI.2542-07.2008.

Kumar, A., Schrader, S., Aertsen, A., & Rotter, S. (2008). The high-conductance state of cortical networks. Neural computation, 20(1), 1-43. https://ieeexplore.ieee.org/abstract/document/6795938.

Lagzi, F., Bustos, M. C., Oswald, A. M., & Doiron, B. (2021). Assembly formation is stabilized by Parvalbumin neurons and accelerated by Somatostatin neurons. bioRxiv, 2021-09. https://www.biorxiv.org/content/10.1101/2021.09.06.459211v1.abstract.

Lankarany M, Prescott SA. Multiplexed coding through synchronous and asynchronous spiking. BMC Neurosci. 2015;16(1): 1-2. https://doi.org/10.1186/1471-2202-16-S1-P198.

Lefort, S., Tomm, C., Sarria, J. C. F., & Petersen, C. C. (2009). The excitatory neuronal network of the C2 barrel column in mouse primary somatosensory cortex. Neuron, 61(2), 301-316. https://www.sciencedirect.com/science/article/pii/S0896627308010921.

Legenstein, R., Pecevski, D., & Maass, W. (2008). A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. PLoS computational biology, 4(10), e1000180. https://doi.org/10.1371/journal.pcbi.1000180.

Le Roux, N., & Bengio, Y. (2008). Representational power of restricted Boltzmann machines and deep belief networks. Neural computation, 20(6), 1631-1649. https://ieeexplore.ieee.org/abstract/document/6796877.

Lee, J. H., Delbruck, T., & Pfeiffer, M. (2016). Training deep spiking neural networks using backpropagation. Frontiers in neuroscience, 10, 508. https://www.frontiersin.org/articles/10.3389/fnins.2016.00508/full.

Lewis, J. (2008, March 10). *The Perfect Novel You've Never Heard Of: Rediscovering Juan Rulfo's Pedro Paramo.* Slate. https://web.archive.org/web/20200105080259/https://slate.com/culture/2008/03/rediscovering-juan-rulfo-s-pedro-paramo.html.

Lindsay, G. W. (2021). Convolutional neural networks as a model of the visual system: Past, present, and future. Journal of cognitive neuroscience, 33(10), 2017-2031. https://doi.org/10.1162/jocn_a_01544.

Litwin-Kumar, A., & Doiron, B. (2012). Slow dynamics and high variability in balanced cortical networks with clustered connections. Nature neuroscience, 15(11), 1498-1505. https://www.nature.com/articles/nn.3220.

Lourenço, J., Koukouli, F., & Bacci, A. (2020). Synaptic inhibition in the neocortex: Orchestration and computation through canonical circuits and variations on the theme. Cortex, 132, 258-280. https://doi.org/10.1016/j.cortex.2020.08.015.

Luz, Y., & Shamir, M. (2012). Balancing feed-forward excitation and inhibition via Hebbian inhibitory synaptic plasticity. PLoS computational biology, 8(1), e1002334. https://doi.org/10.1371/journal.pcbi.1002334.

Maass, W. (2011). Liquid state machines: motivation, theory, and applications. Computability in context: computation and logic in the real world, 275-296. https://www.worldscientific.com/doi/abs/10.1142/9781848162778_0008.

Maass, W., & Markram, H. (2004). On the computational power of circuits of spiking neurons. Journal of computer and system sciences, 69(4), 593-616. https://www.sciencedirect.com/science/article/pii/S0022000004000406.

Macpherson, T., Churchland, A., Sejnowski, T., DiCarlo, J., Kamitani, Y., Takahashi, H., & Hikida, T. (2021). Natural and Artificial Intelligence: A brief introduction to the interplay between AI and neuroscience research. Neural Networks, 144, 603-613. https://doi.org/10.1016/j.neunet.2021.09.018.

Marblestone, A. H., Wayne, G., & Kording, K. P. (2016). Toward an integration of deep learning and neuroscience. Frontiers in computational neuroscience, 94. https://doi.org/10.3389/fncom.2016.00094.

Marques, T., Summers, M. T., Fioreze, G., Fridman, M., Dias, R. F., Feller, M. B., & Petreanu, L. (2018). A role for mouse primary visual cortex in motion perception. Current Biology, 28(11), 1703-1713. https://doi.org/10.1016/j.cub.2018.04.012.

Masse, N. Y., Yang, G. R., Song, H. F., Wang, X. J., & Freedman, D. J. (2019). Circuit mechanisms for the maintenance and manipulation of information in working memory. Nature neuroscience, 22(7), 1159-1167. https://www.nature.com/articles/s41593-019-0414-3.

Masse, N. Y., Rosen, M. C., Tsao, D. Y., & Freedman, D. J. (2022). Flexible cognition in context-modulated reservoir networks. bioRxiv, 2022-05. https://www.biorxiv.org/content/10.1101/2022.05.09.491102v3.abstract.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5, 115-133. https://link.springer.com/article/10.1007/BF02478259.

McEliece, R., Posner, E., Rodemich, E., & Venkatesh, S. (1987). The capacity of the Hopfield associative memory. IEEE transactions on Information Theory, 33(4), 461-482. https://ieeexplore.ieee.org/abstract/document/1057328.

McIntosh, L., Maheswaranathan, N., Nayebi, A., Ganguli, S., & Baccus, S. (2016). Deep learning models of the retinal response to natural scenes. Advances in neural information processing systems, 29. https://proceedings.neurips.cc/paper/2016/hash/a1d33d0dfec820b41b54430b50e96b5c-Abstract.html.

Meffin, H., Burkitt, A. N., & Grayden, D. B. (2004). An analytical model for the 'large, fluctuating synaptic conductance state'typical of neocortical neurons in vivo. Journal of computational neuroscience, 16, 159-175. https://link.springer.com/article/10.1023/B:JCNS.0000014108.03012.81.

Mejias, J. F., & Longtin, A. (2012). Optimal heterogeneity for coding in spiking neural networks. Physical Review Letters, 108(22), 228102. https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.108.228102.

Milgram, S. (1967). The small world problem. Psychology today, 2(1), 60-67. http://snap.stanford.edu/class/cs224w-readings/milgram67smallworld.pdf.

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., & Alon, U. (2002). Network motifs: simple building blocks of complex networks. Science, 298(5594), 824-827. https://www.science.org/doi/full/10.1126/science.298.5594.824.

Moons, W. G., Mackie, D. M., & Garcia-Marques, T. (2009). The impact of repetition-induced familiarity on agreement with weak and strong arguments. Journal of Personality and Social Psychology, 96(1), 32. https://doi.org/10.1037/a0013461.

Morrison, D. J., Rashid, A. J., Yiu, A. P., Yan, C., Frankland, P. W., & Josselyn, S. A. (2016). Parvalbumin interneurons constrain the size of the lateral amygdala engram. Neurobiology of learning and memory, 135, 91-99. https://doi.org/10.1016/j.nlm.2016.07.007.

Morrone, M. C., Burr, D. C., & Maffei, L. (1982). Functional implications of cross-orientation inhibition of cortical visual cells. I. Neurophysiological evidence. Proceedings of the Royal Society of London. Series B. Biological Sciences, 216(1204), 335-354. https://royalsocietypublishing.org/doi/abs/10.1098/rspb.1982.0078.

Muldoon, S. F., Bridgeford, E. W., & Bassett, D. S. (2016). Small-world propensity and weighted brain networks. Scientific reports, 6(1), 22057. https://www.nature.com/articles/srep22057.

Nielsen, B. (2006). Correlograms for non‐stationary autoregressions. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 68(4), 707-720. https://doi.org/10.1111/j.1467-9868.2006.00563.x.

Nigam, S., Shimono, M., Ito, S., Yeh, F. C., Timme, N., Myroshnychenko, M., ... & Beggs, J. M. (2016). Rich-club organization in effective connectivity among cortical neurons. Journal of Neuroscience, 36(3), 670-684. https://www.jneurosci.org/content/36/3/670.short.

Ocker, G. K., Hu, Y., Buice, M. A., Doiron, B., Josić, K., Rosenbaum, R., & Shea-Brown, E. (2017). From the statistics of connectivity to the statistics of spike times in neuronal networks. Current opinion in neurobiology, 46, 109-119. https://doi.org/10.1016/j.conb.2017.07.011.

Olshausen, B. A. (2002). 13 sparse codes and spikes. Probabilistic models of the brain, 257.

Olshausen, B. A., & Field, D. J. (2004). Sparse coding of sensory inputs. Current opinion in neurobiology, 14(4), 481-487. https://www.sciencedirect.com/science/article/pii/S0959438804001035.

Orlandi, J. G., Soriano, J., Alvarez-Lacalle, E., Teller, S., & Casademunt, J. (2013). Noise focusing and the emergence of coherent activity in neuronal cultures. Nature Physics, 9(9), 582-590. https://www.nature.com/articles/nphys2686.

Pajevic, S., & Plenz, D. (2009). Efficient network reconstruction from dynamical cascades identifies small-world topology of neuronal avalanches. PLoS computational biology, 5(1), e1000271. https://doi.org/10.1371/journal.pcbi.1000271.

Palm, G., & Sommer, F. T. (1992). Information capacity in recurrent McCulloch-Pitts networks with sparsely coded memory states. Network: Computation in Neural Systems, 3(2), 177. https://doi.org/10.1088/0954-898X/3/2/006.

Pan, W., Zhao, F., Zeng, Y., & Han, B. (2023). Adaptive structure evolution and biologically plausible synaptic plasticity for recurrent spiking neural networks. arXiv preprint arXiv:2304.01015. https://arxiv.org/abs/2304.01015.

Payeur, A., Béïque, J. C., & Naud, R. (2019). Classes of dendritic information processing. Current opinion in neurobiology, 58, 78-85. https://doi.org/10.1016/j.conb.2019.07.006.

Perin, R., Berger, T. K., & Markram, H. (2011). A synaptic organizing principle for cortical neuronal groups. Proceedings of the National Academy of Sciences, 108(13), 5419-5424. https://doi.org/10.1073/pnas.1016051108.

Pezzulo, G., Barsalou, L. W., Cangelosi, A., Fischer, M. H., McRae, K., & Spivey, M. J. (2011). The mechanics of embodiment: A dialog on embodiment and computational modeling. Frontiers in psychology, 2, 5. https://www.frontiersin.org/articles/10.3389/fpsyg.2011.00005/full.

Poirazi, P., & Papoutsi, A. (2020). Illuminating dendritic function with computational models. Nature Reviews Neuroscience, 21(6), 303-321. https://www.nature.com/articles/s41583-020-0301-7.

Ponce, C. R., Xiao, W., Schade, P. F., Hartmann, T. S., Kreiman, G., & Livingstone, M. S. (2019). Evolving images for visual neurons using a deep generative network reveals coding principles and neuronal preferences. Cell, 177(4), 999-1009. https://www.sciencedirect.com/science/article/pii/S0092867419303915.

Rajalingham, R., Issa, E. B., Bashivan, P., Kar, K., Schmidt, K., & DiCarlo, J. J. (2018). Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. Journal of Neuroscience, 38(33), 7255-7269. https://doi.org/10.1523/JNEUROSCI.0388-18.2018.

Reimann, M. W., Nolte, M., Scolamiero, M., Turner, K., Perin, R., Chindemi, G., ... & Markram, H. (2017). Cliques of neurons bound into cavities provide a missing link between structure and function. Frontiers in computational neuroscience, 48. https://doi.org/10.3389/fncom.2017.00048.

Renart, A., De La Rocha, J., Bartho, P., Hollender, L., Parga, N., Reyes, A., & Harris, K. D. (2010). The asynchronous state in cortical circuits. science, 327(5965), 587-590. https://www.science.org/doi/full/10.1126/science.1179850.

Resulaj, A., Ruediger, S., Olsen, S. R., & Scanziani, M. (2018). First spikes in visual cortex enable perceptual discrimination. Elife, 7, e34044. https://elifesciences.org/articles/34044.

Robinson, A. J., & Fallside, F. (1987). The utility driven dynamic error propagation network (Vol. 1). Cambridge: University of Cambridge Department of Engineering.

Rolls, E. T. (2021). Brain computations: what and how. Oxford University Press, USA.

Rosenbaum, R. (2022). On the relationship between predictive coding and backpropagation. Plos one, 17(3), e0266102. https://doi.org/10.1371/journal.pone.0266102.

Rosenblatt, F. (1957). The perceptron, a perceiving and recognizing automaton Project Para. Cornell Aeronautical Laboratory.

Rosenblatt, F. (1961). Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Cornell Aeronautical Lab Inc Buffalo NY. https://apps.dtic.mil/sti/citations/AD0256582.

Roth, G., & Dicke, U. (2005). Evolution of the brain and intelligence. Trends in cognitive sciences, 9(5), 250-257. https://www.sciencedirect.com/science/article/pii/S1364661305000823.

Rothschild, G., Nelken, I., & Mizrahi, A. (2010). Functional organization and population dynamics in the mouse primary auditory cortex. Nature neuroscience, 13(3), 353-360. https://www.nature.com/articles/nn.2484.

Roxin, A., Brunel, N., Hansel, D., Mongillo, G., & van Vreeswijk, C. (2011). On the distribution of firing rates in networks of cortical neurons. Journal of Neuroscience, 31(45), 16217-16226. https://www.jneurosci.org/content/31/45/16217.short.

Rudolph-Lilith, M., Dubois, M., & Destexhe, A. (2012). Analytical integrate-and-fire neuron models with conductance-based dynamics and realistic postsynaptic potential time course for event-driven simulation strategies. Neural computation, 24(6), 1426-1461. https://doi.org/10.1162/NECO_a_00278.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. nature, 323(6088), 533-536. https://www.nature.com/articles/323533a0.

Runyan, C. A., Piasini, E., Panzeri, S., & Harvey, C. D. (2017). Distinct timescales of population coding across cortex. Nature, 548(7665), 92-96. https://www.nature.com/articles/nature23020.

Saadi, S. (2009, August 7). Broughton, C. (ed.) *Book Of A Lifetime: Pedro Páramo, By Juan Rulfo.* The Independent. https://www.independent.co.uk/arts-entertainment/books/reviews/book-of-a-lifetime-pedro-p-225-ramo-by-juan-rulfo-1768135.html.

Sadovsky, A. J., & MacLean, J. N. (2014). Mouse visual neocortex supports multiple stereotyped patterns of microcircuit activity. Journal of Neuroscience, 34(23), 7769-7777. https://doi.org/10.1523/JNEUROSCI.0169-14.2014.

Salaj, D., Subramoney, A., Kraisnikovic, C., Bellec, G., Legenstein, R., & Maass, W. (2021). Spike frequency adaptation supports network computations on temporally dispersed information. Elife, 10, e65459. https://doi.org/10.7554/eLife.65459.

Salinas, E., & Sejnowski, T. J. (2001). Correlated neuronal activity and the flow of neural information. Nature reviews neuroscience, 2(8), 539-550. https://www.nature.com/articles/35086012.

Schaeffer, R., Khona, M., & Fiete, I. (2022). No free lunch from deep learning in neuroscience: A case study through models of the entorhinal-hippocampal circuit. bioRxiv, 2022-08. https://www.biorxiv.org/content/10.1101/2022.08.07.503109v1.abstract.

Scherr, F., & Maass, W. (2021). Analysis of the computational strategy of a detailed laminar cortical microcircuit model for solving the image-change-detection task. bioRxiv, 2021-11. https://doi.org/10.1101/2021.11.17.469025.

Schmidhuber, J. (2022). Annotated History of Modern AI and Deep Learning. arXiv preprint arXiv:2212.11279. https://arxiv.org/abs/2212.11279.

Schmutz, V., Brea, J., & Gerstner, W. (2023). Convergence of redundancy-free spiking neural networks to rate networks. arXiv preprint arXiv:2303.05174. https://arxiv.org/abs/2303.05174.

Schulz, A., Miehl, C., Berry II, M. J., & Gjorgjieva, J. (2021). The generation of cortical novelty responses through inhibitory plasticity. Elife, 10, e65309. https://elifesciences.org/articles/65309.

Seeman, S. C., Campagnola, L., Davoudian, P. A., Hoggarth, A., Hage, T. A., Bosma-Moody, A., ... & Jarsky, T. (2018). Sparse recurrent excitatory connectivity in the microcircuit of the adult mouse and human cortex. elife, 7, e37349. https://elifesciences.org/articles/37349.

Sengupta, B., Laughlin, S. B., & Niven, J. E. (2013). Balanced excitatory and inhibitory synaptic currents promote efficient coding and metabolic efficiency. PLoS computational biology, 9(10), e1003263. https://doi.org/10.1371/journal.pcbi.1003263.

Shahidi, N., Andrei, A. R., Hu, M., & Dragoi, V. (2019). High-order coordination of cortical spiking activity modulates perceptual accuracy. Nature neuroscience, 22(7), 1148-1158. https://www.nature.com/articles/s41593-019-0406-3.

Sharmin, S., Rathi, N., Panda, P., & Roy, K. (2020). Inherent adversarial robustness of deep spiking neural networks: Effects of discrete input encoding and non-linear activations. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16 (pp. 399-414). Springer International Publishing. https://link.springer.com/chapter/10.1007/978-3-030-58526-6_24.

Sharpee, T. O. (2017). Optimizing neural information capacity through discretization. Neuron, 94(5), 954-960. https://doi.org/10.1016/j.neuron.2017.04.044.

Shervani-Tabar, N., & Rosenbaum, R. (2023). Meta-learning biologically plausible plasticity rules with random feedback pathways. Nature Communications, 14(1), 1805. https://www.nature.com/articles/s41467-023-37562-1.

Shew, W. L., Yang, H., Petermann, T., Roy, R., & Plenz, D. (2009). Neuronal avalanches imply maximum dynamic range in cortical networks at criticality. Journal of neuroscience, 29(49), 15595-15600. https://doi.org/10.1523/JNEUROSCI.3864-09.2009.

Shew, W. L., Yang, H., Yu, S., Roy, R., & Plenz, D. (2011). Information capacity and transmission are maximized in balanced cortical networks with neuronal avalanches. Journal of neuroscience, 31(1), 55-63. https://www.jneurosci.org/content/31/1/55.short.

Shi, L., Niu, X., & Wan, H. (2015). Effect of the small-world structure on encoding performance in the primary visual cortex: an electrophysiological and modeling analysis. Journal of Comparative Physiology A, 201, 471-483. https://link.springer.com/article/10.1007/s00359-015-0996-5.

Shimono, M., & Beggs, J. M. (2015). Functional clusters, hubs, and communities in the cortical microconnectome. Cerebral Cortex, 25(10), 3743-3757. https://academic.oup.com/cercor/article/25/10/3743/390815.

Shrestha, A., Fang, H., Wu, Q., & Qiu, Q. (2019, July). Approximating back-propagation for a biologically plausible local learning rule in spiking neural networks. In Proceedings of the International Conference on Neuromorphic Systems (pp. 1-8). https://doi.org/10.1145/3354265.3354275.

Softky, W. R., & Koch, C. (1993). The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs. Journal of neuroscience, 13(1), 334-350. https://www.jneurosci.org/content/13/1/334.short.

Sohal, V. S., & Rubenstein, J. L. (2019). Excitation-inhibition balance as a framework for investigating mechanisms in neuropsychiatric disorders. Molecular psychiatry, 24(9), 1248-1257. https://www.nature.com/articles/s41380-019-0426-0.

Sommer, H., & Schreiber, L. (2012). Is logic in the mind or in the world? why a philosophical question can affect the understanding of intelligence. Journal of Artificial General Intelligence, 3(1), 25-47.

Song, S., Sjöström, P. J., Reigl, M., Nelson, S., & Chklovskii, D. B. (2005). Highly nonrandom features of synaptic connectivity in local cortical circuits. PLoS biology, 3(3), e68. https://doi.org/10.1371/journal.pbio.0030068.

Spoerer, C. J., Kietzmann, T. C., Mehrer, J., Charest, I., & Kriegeskorte, N. (2020). Recurrent neural networks can explain flexible trading of speed and accuracy in biological vision. PLoS computational biology, 16(10), e1008215. https://doi.org/10.1371/journal.pcbi.1008215.

Stuart, G. J., & Spruston, N. (2015). Dendritic integration: 60 years of progress. Nature neuroscience, 18(12), 1713-1721. https://www.nature.com/articles/nn.4157.

Sundararajan, M., Taly, A., & Yan, Q. (2017, July). Axiomatic attribution for deep networks. In International conference on machine learning (pp. 3319-3328). PMLR. http://proceedings.mlr.press/v70/sundararajan17a.html.

Symons, J. (2008). Computational models of emergent properties. Minds and Machines, 18, 475-491. https://link.springer.com/article/10.1007/s11023-008-9120-8.

Tavanaei, A., Ghodrati, M., Kheradpisheh, S. R., Masquelier, T., & Maida, A. (2019). Deep learning in spiking neural networks. Neural networks, 111, 47-63. https://www.sciencedirect.com/science/article/pii/S0893608018303332.

Teeter, C., Iyer, R., Menon, V., Gouwens, N., Feng, D., Berg, J., ... & Mihalas, S. (2018). Generalized leaky integrate-and-fire models classify multiple neuron types. Nature communications, 9(1), 709. https://www.nature.com/articles/s41467-017-02717-4.

Teramae, J. N., Tsubo, Y., & Fukai, T. (2012). Optimal spike-based communication in excitable networks with strong-sparse and weak-dense links. Scientific reports, 2(1), 485. https://www.nature.com/articles/srep00485.

Timme, N. M., Ito, S., Myroshnychenko, M., Nigam, S., Shimono, M., Yeh, F.-C., et al. (2016). High-degree neurons feed cortical computations. PLoS Comput Biol, 12(5), e1004858. https://doi.org/10.1371/journal.pcbi.1004858.

Trensch, G., Gutzen, R., Blundell, I., Denker, M., & Morrison, A. (2018). Rigorous neural network simulations: model cross-validation for boosting the correctness of simulation results. Front. Neuroinformatics. https://www.frontiersin.org/articles/10.3389/fninf.2018.00081/full.

Turing AM. Computing Machinery and Intelligence. In: Epstein R, Roberts G, Beber G, editors. Parsing the Turing Test. Springer, Dordrecht; 2009.

van Albada, S. J., Morales-Gregorio, A., Dickscheid, T., Goulas, A., Bakker, R., Bludau, S., ... & Diesmann, M. (2021). Bringing anatomical information into neuronal network models. In Computational Modelling of the Brain: Modelling Approaches to Cells, Circuits and Networks

(pp. 201-234). Cham: Springer International Publishing. https://link.springer.com/chapter/10.1007/978-3-030-89439-9_9.

van Rossum, M. C. (2001). A novel spike distance. Neural computation, 13(4), 751-763. https://ieeexplore.ieee.org/abstract/document/6790198/.

Van Vreeswijk, C., & Sompolinsky, H. (1996). Chaos in neuronal networks with balanced excitatory and inhibitory activity. Science, 274(5293), 1724-1726. https://www.science.org/doi/abs/10.1126/science.274.5293.1724.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Vegué, M., Perin, R., & Roxin, A. (2017). On the structure of cortical microcircuits inferred from small sample sizes. Journal of Neuroscience, 37(35), 8498-8510. https://doi.org/10.1523/JNEUROSCI.0984-17.2017.

Verzi, S. J., Rothganger, F., Parekh, O. D., Quach, T., Miner, N. E,, Vineyard, C. M., James, C. D., Aimone, J. B. (2018). Computing with spikes: The advantage of fine-grained timing. *Neural Computation, 30*, 2660–2690. https://doi.org/10.1162/neco_a_01113.

Vogels, T. P., & Abbott, L. F. (2005). Signal propagation and logic gating in networks of integrate-and-fire neurons. Journal of neuroscience, 25(46), 10786-10795. https://doi.org/10.1523/JNEUROSCI.3508-05.2005.

Vogels, T. P., Sprekeler, H., Zenke, F., Clopath, C., & Gerstner, W. (2011). Inhibitory plasticity balances excitation and inhibition in sensory pathways and memory networks. Science, 334(6062), 1569-1573. https://www.science.org/doi/full/10.1126/science.1211095.

Vogels, T. P., Froemke, R. C., Doyon, N., Gilson, M., Haas, J. S., Liu, R., ... & Sprekeler, H. (2013). Inhibitory synaptic plasticity: spike timing-dependence and putative network function. Frontiers in neural circuits, 7, 119. https://www.frontiersin.org/articles/10.3389/fncir.2013.00119/full.

Wang, Y., Markram, H., Goodman, P. H., Berger, T. K., Ma, J., & Goldman-Rakic, P. S. (2006). Heterogeneity in the pyramidal network of the medial prefrontal cortex. Nature neuroscience, 9(4), 534-542. https://www.nature.com/articles/nn1670.

Werbos, P. J. (1988). Generalization of backpropagation with application to a recurrent gas market model. Neural networks, 1(4), 339-356. https://www.sciencedirect.com/science/article/pii/089360808890007X.

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. Proceedings of the IEEE, 78(10), 1550-1560. https://ieeexplore.ieee.org/abstract/document/58337.

Whiteway, M. R., & Butts, D. A. (2019). The quest for interpretable models of neural population activity. Current opinion in neurobiology, 58, 86-93. https://doi.org/10.1016/j.conb.2019.07.004.

Winsberg, E. (2010). Science in the Age of Computer Simulation. University of Chicago Press.

Yamins, D. L., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., & DiCarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. Proceedings of the national academy of sciences, 111(23), 8619-8624. https://www.pnas.org/doi/abs/10.1073/pnas.1403112111.

Yang, J. Q., Wang, R., Ren, Y., Mao, J. Y., Wang, Z. P., Zhou, Y., & Han, S. T. (2020). Neuromorphic Engineering: From Biological to Spike‐Based Hardware Nervous Systems. Advanced Materials, 32(52), 2003610. https://doi.org/10.1002/adma.202003610.

Yu, S., Yang, H., Nakahara, H., Santos, G. S., Nikolić, D., & Plenz, D. (2011). Higher-order interactions characterized in cortical activity. Journal of neuroscience, 31(48), 17514-17526. https://www.jneurosci.org/content/31/48/17514.short.

Zador, A. M. (2019). A critique of pure learning and what artificial neural networks can learn from animal brains. Nature communications, 10(1), 3770. https://www.nature.com/articles/s41467-019-11786-6.

Zenke, F., & Vogels, T. P. (2021). The remarkable robustness of surrogate gradient learning for instilling complex function in spiking neural networks. Neural computation, 33(4), 899-925. https://doi.org/10.1162/neco_a_01367.

Zerlaut, Y., Zucca, S., Panzeri, S., & Fellin, T. (2019). The spectrum of asynchronous dynamics in spiking networks as a model for the diversity of non-rhythmic waking states in the neocortex. Cell reports, 27(4), 1119-1132. https://doi.org/10.1016/j.celrep.2019.03.102.

Zhang, M., Wang, J., Zhang, Z., Belatreche, A., Wu, J., Chua, Y., ... & Li, H. (2020). Spike-timing-dependent back propagation in deep spiking neural networks. arXiv preprint arXiv:2003.11837. https://arxiv.org/abs/2003.11837v1.

Zhu, V., & Rosenbaum, R. (2022). Evaluating the extent to which homeostatic plasticity learns to compute prediction errors in unstructured neuronal networks. Journal of Computational Neuroscience, 50(3), 357-373. https://link.springer.com/article/10.1007/s10827-022-00820-0.

Zhu, Y., Scherr, F., Maass, W., MacLean, J. (2020, November 9-12). Addition of neocortical features permits successful training of spiking neuronal network models [Conference presentation]. From Neuroscience to Artificially Intelligent Systems, Cold Spring Harbor Laboratory, NY, United States. https://meetings.cshl.edu/meetings.aspx?meet=naisys&year=20.

Zylberberg, J., Pouget, A., Latham, P. E., & Shea-Brown, E. (2017). Robust information propagation through noisy neural circuits. PLoS computational biology, 13(4), e1005497. https://doi.org/10.1371/journal.pcbi.1005497.