Channels and circuits: biophysical and network models of neuronal function

William F Podlaski

New College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Trinity 2018

Abstract

Neuroscience research studies the brain at various different levels of detail. Experimental work explores everything from the molecular machinery within each neuron, to the behavioral output of an organism. Similarly, computational models are designed to operate at many scales, and can address different questions depending upon the complexity of the description.

Detailed biophysical modelling incorporates findings about neuronal physiology and structure in order to test hypotheses and generate predictions. However, due to the complexity and number of modelling studies, it is difficult to compare models and to assess biological fidelity. In this work we present a curated database of nearly 3000 voltage-gated ion channel models used in published neuronal simulations, called *ICGenealogy*. Furthermore, we present a standardized formulation for ion channel dynamics which we fit to all models on this database. Through these two endeavors, we facilitate better experimentally-constrained modelling, while also providing insight into the diversity and complexity of ion channel dynamics seen in single neurons.

At a higher level, neuronal network models integrate findings about neuronal activity and cell types in order to explain representations and computations. Here, we present a model of context-dependent associative memory, which incorporates known principles of memory function from psychology research and proposes concrete functional roles for different components of neural circuits. Through analytics and simulations, we show that a contextual memory system not only provides benefits for memory capacity and robustness, but also enables control of memory expression. This work provides new conceptual ideas for memory research, suggests functional roles for different inhibitory cell types, and may help us to understand the interaction of different memory systems in the brain.

Together, the work presented here spans two distinct levels of detail and addresses current challenges both in biophysically detailed models, as well as computation in abstract network models.

Channels and circuits: biophysical and network models of neuronal function



William F Podlaski

New College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2018

Acknowledgements

This thesis is the culmination of four long years at the Vogels lab in Oxford. During that time, people and projects have come and gone, and I have grown in many ways. I would first of all like to thank my supervisor, Tim Vogels, for all of his support and guidance, and especially his enthusiasm. He taught me many things, about science and life (though mostly about myself), and I will forever be grateful for that. Importantly, he gave me tremendous freedom to explore different research areas, several of which have not been included in this thesis, but were instructive nevertheless. Every PhD is different, and each student has different needs. Tim always had a way of reassuring me that whatever was going on, I could handle it, and he had my back. Not to be overlooked, he also created and fostered a stimulating scientific environment in which rigor and excellence were always expected. I couldn't imagine doing my PhD anywhere else, and it will be hard to leave.

I would also like to thank all of the members of the Vogels lab who made the past four years an amazing experience. Special thanks goes to Everton Agnes, who has been a great friend and mentor. My PhD would not have been the same without him. I'd like to thank him, Friedemann Zenke, and Chaitanya Chintaluri for considering me a grumpy postdoc even before I become one. I'd also like to acknowledge Bayar Menzat, and thank him for the past four years of crazy times. We started our PhDs together – I think we've both changed a lot since we started, and it's nice to look back on the early days. Thanks to Jake Stroud for always setting the bar high in lab – sitting next to him for three years made me a better scientist. Thanks to Yayoi Teramoto for always looking out for me, and especially for taking me to the hospital when I broke my foot. Thanks to all of these people and the others in the lab for the great conversations about science and otherwise: Alon Baram, Homero Barrocas Soares Esmeraldo, Panos Bozèlos, Georgia Christodoulou, Rui P Costa, Victor Hernandez-Urbina, David Jia, and Richard Lonsdale. Thanks to everyone else at the CNCB for the beer, football, and good times. Thanks to Sinead Wright, and especially Fiona Woods, for taking care of everything behind the scenes at the CNCB. Thanks to Sarah Noujaim for doing the same at DPAG.

Thanks to everyone involved in the ICGenealogy project, including Rajnish Ranjan, Lukas Groschner and Gero Miesenboeck. A special thank you goes to Alex Seeholzer, whose drive and skills really pushed the project forward. Even in

the “darkest” times of the project, I have fond memories of working alongside him (mostly remotely). I would also like to acknowledge Chaitanya Chintaluri again, whose work in the past few months on the standardized ion channel formulation was crucial to the results presented here.

Finally, I’d like to thank my friends and family for providing support and perspective throughout the past four years. Thanks to my parents, Frank and Carol Podlaski, and my sister, Trisha, for their love and support. It means a lot to me and I don’t tell them enough. Thanks to Eszter for helping me to get through the first few years, for teaching me quite a lot and taking care of me. Thanks to all of the other family, friends and colleagues too numerous to mention.

Abstract

Neuroscience research studies the brain at various different levels of detail. Experimental work explores everything from the molecular machinery within each neuron, to the behavioral output of an organism. Similarly, computational models are designed to operate at many scales, and can address different questions depending upon the complexity of the description.

Detailed biophysical modelling incorporates findings about neuronal physiology and structure in order to test hypotheses and generate predictions. However, due to the complexity and number of modelling studies, it is difficult to compare models and to assess biological fidelity. In this work we present a curated database of nearly 3000 voltage-gated ion channel models used in published neuronal simulations, called *ICGenealogy*. Furthermore, we present a standardized formulation for ion channel dynamics which we fit to all models on this database. Through these two endeavors, we facilitate better experimentally-constrained modelling, while also providing insight into the diversity and complexity of ion channel dynamics seen in single neurons.

At a higher level, neuronal network models integrate findings about neuronal activity and cell types in order to explain representations and computations. Here, we present a model of context-dependent associative memory, which incorporates known principles of memory function from psychology research and proposes concrete functional roles for different components of neural circuits. Through analytics and simulations, we show that a contextual memory system not only provides benefits for memory capacity and robustness, but also enables control of memory expression. This work provides new conceptual ideas for memory research, suggests functional roles for different inhibitory cell types, and may help us to understand the interaction of different memory systems in the brain.

Together, the work presented here spans two distinct levels of detail and addresses current challenges both in biophysically detailed models, as well as computation in abstract network models.

Contents

List of Figures **xi**

List of Tables **xiii**

Preface **xv**

I Single neuron modelling and ion channels **1**

1 Introduction: biophysical neuron modelling **3**

1.1 Electrical properties of neurons 4

1.1.1 Passive cell properties 4

1.1.2 Active properties: biophysical models of ion channels 7

1.1.3 Capturing the diverse response properties of neurons 13

1.2 Tools and resources in neuroinformatics 16

2 Mapping the function of neuronal ion channels in model and experiment **19**

2.1 Introduction 20

2.2 Results 23

2.2.1 Ion channel model metadata and ancestor-descendant relationships 23

2.2.2 Assessing model similarity through voltage clamp protocols and clustering 27

2.2.3 Validity and variability of available models 33

2.2.4 Automated comparison of new ion channel models and experimental data 35

2.3 Discussion 38

2.4 Methods 40

2.4.1 Pyramidal cell model (Figure 1) 40

2.4.2 The *ModelDB* database, NEURON language and ion channel subtype nomenclature 41

2.4.3 Collection of metadata and ancestor-descendant relationships 42

2.4.4 Voltage-clamp protocols and data processing 45
2.4.5 Similarity measure and clustering 49
2.4.6 Generation of figures 52
2.5 Supplementary materials and figures 53
2.5.1 Abbreviations 53

3 A standardized model for voltage-gated ion channel dynamics 61
3.1 Introduction 62
3.2 Results 63
3.2.1 standardized model formulation 64
3.2.2 Direct model fitting 65
3.2.3 Comparison using voltage-clamp protocols 69
3.2.4 Current status 72
3.2.5 Realistic neuron and network simulations 72
3.2.6 Future work: clustering in parameter space and comparison with metadata 74
3.3 Discussion 76
3.4 Methods 78
3.4.1 Details of standardized model formulation 78
3.4.2 Direct model fitting 80
3.4.3 Voltage-clamp protocols 81
3.4.4 Single-neuron simulation 82
3.5 Supplementary figures 83

II Models of context-dependent processing in neuronal networks 85

4 Introduction: neural circuits and memory 87
4.1 Neuronal network dynamics, organization and function 88
4.1.1 Neuronal circuits of the brain 88
4.1.2 Abstract descriptions of neuronal dynamics 93
4.2 Memory 96
4.2.1 The psychology of learning and memory 97
4.2.2 The neuroscience of learning and memory 102
4.2.3 Computational models of associative memory 104
4.2.4 The Hopfield model 108
4.2.5 Extensions and biological realism 109

5 A model of context-dependent associative memory 113
5.1 Introduction . . . 114
5.2 Results . . . 116
5.2.1 Context-modular Hopfield networks with subnetwork selection exhibit enhanced memory capacity . . . 118
5.2.2 Modulation through dendritic input selection does not enhance capacity . . . 122
5.2.3 Targeted dendritic input gating on top of modular Hopfield further enhances capacity . . . 124
5.2.4 Robustness to noise, overloading and spurious memories . . . 128
5.2.5 Dynamic gating of memory expression . . . 130
5.3 Discussion . . . 132
5.4 Methods . . . 135
5.4.1 Model formulation . . . 135
5.4.2 Comparison with standard Hopfield network and low-activity network . . . 137
5.4.3 Analytical capacity estimation . . . 138
5.4.4 Numerical simulations . . . 143
5.4.5 Robustness to noise . . . 145
5.4.6 Dynamic gating model . . . 146
5.A Supplementary Methods . . . 147
5.A.1 Equivalence of {-1,+1} and {0,1} formulations . . . 147
5.A.2 Heuristic mean field analysis . . . 149
5.B Supplementary Figures . . . 160

6 Conclusion 165

Bibliography 173

List of Figures

1.1 Passive properties of the cell membrane. 5
1.2 The Hodgkin-Huxley model of action potential generation. 9

2.1 The choice of ion channel model influences the behavior of a simulated neuron. 22
2.2 Ion channel models can be categorized by metadata and ancestor-descendant relationships. 25
2.3 Graph of ancestor-descendant relationships for Kv models. 26
2.4 Voltage-clamp protocols for the quantitative analysis of ion channel models. 28
2.5 Quantitative analysis of Kv ion channel models: functional map and clusters of common behavior. 30
2.6 Quantitative analysis of Kv ion channel models: Circos plot. 31
2.7 Ion channel model groups defined by common subtype, neuron type and brain area show variability in behavior. 36
2.8 Automated analysis of new models and experimental data. 37
2.9 The ICGenealogy website allows for the interactive visualization of all data and analysis on the resource [ICGenealogy, 2017]. 39
2.10 Graphical description of the five voltage-clamp protocols used for ion channel model analysis 48
2.11 Nav class genealogy and clustering. 54
2.12 Cav class genealogy and clustering. 55
2.13 KCa class genealogy and clustering. 56
2.14 IH class genealogy and clustering. 57
2.15 Variability of groups defined by common subtype, neuron type, brain area and animal species. 58
2.16 Additional browser views on the ICGenealogy website. 59

3.1 Steady-state and time constant curves of gating variables for a random selection of voltage gated K, Na, Ca and Ih channels. 64
3.2 Direct standardized model fitting results. 66
3.3 Comparison of direct fitting across ion type for different model formulations. 68

3.4 Validation of standardized model formulation through ICGenealogy voltage-clamp protocols. 70
3.5 Current status of standardized model fitting procedure. 72
3.6 Test of standardized models in full single-neuron simulation. 75
3.7 Direct standardized model fits for full single-neuron simulation. . . 84

4.1 Neuronal input-output response functions. 96
4.2 Memory systems of the brain. 98
4.3 Types of transient (A,C) and attractor (B,D) memories. 106

5.1 Schematic of the context-modular Hopfield network. 118
5.2 Memory capacity of the context-modular Hopfield network with subnetwork selection. 121
5.3 Relative information content of context-modular Hopfield network with subnetwork modulation 123
5.4 Memory capacity of the context-modular Hopfield network with random dendritic input selection. 125
5.5 Schematic of targeted dendritic input gating. 126
5.6 Targeted dendritic input gating further enhances memory capacity . 127
5.7 Stability of relevant and irrelevant memories 129
5.8 Robustness to memory overloading 131
5.9 Dynamic gating of memory expression. 132
5.10 Memory capacity of low-activity Hopfield network. 161
5.11 Comparison of overlap of the context-modular Hopfield network with subnetwork selection vs. dendritic input selection. 162
5.12 Noise distribution and variance estimation for mean-field method . 163
5.13 Relative information content of context-modular Hopfield network with targeted dendritic input gating (T-DIG) 163
5.14 Evaluation of simulation approximation procedure 164

List of Tables

1.1 Ionic concentrations in neurons. 6

2.1 Subtypes of each of the five classes currently available in the resource (Kv, Nav, Cav, KCa, Ih) 44
2.2 Voltage-clamp protocol parameters for the five ion type classes. . . 46
2.3 Parameters for reversal potential and inside and outside concentrations used in simulation protocols for five ion type classes. 46
2.4 List of omitted files 49

Preface

The study of the mind and brain has taken many forms over the history of humankind. Modern neuroscience arguably began around the turn of the 20th century, when Camillo Golgi and Santiago Ramon y Cajal, among others, helped to identify the neuron as the fundamental unit of the nervous system [De Carlos and Borrell, 2007]. During the past century, the field has made many important discoveries – the mechanism behind neuronal communication [Hodgkin and Huxley, 1952a], response properties of sensory neurons [Hubel and Wiesel, 1962], and potential mechanisms of learning and memory [Bliss and Lømo, 1973], to name a few. Furthermore, computational and theoretical work has started to take a more prominent role in brain research [Abbott, 2008].

This thesis is divided into two standalone parts on different aspects of neuronal systems, with different levels of analysis and questions. In my opinion, they correspond to two important directions that neuroscience research has taken recently. The first of these relates to building tools and resources in order to curate and analyze neuroscientific knowledge and data. Due to recent technological advances, experimental neuroscience research is generating extremely large datasets at a consistent pace, which will only grow larger with time [Akil et al., 2011, Ferguson et al., 2014, Grillner, 2014]. Computational studies also produce an overwhelming amount of models and simulation data which need to be integrated and standardized [Gleeson et al., 2018]. To address these challenges, *neuroinformatics* is becoming increasingly important.

In **Part I** of this thesis, I present a new resource for computational neuroscience, specifically focused on ion channel models used in neuronal simulations [Podlaski et al., 2017]. This database, called *ICGenealogy*, aims to curate existing models, while also facilitating better experimentally-constrained modelling. Following a brief literature review of biophysical modelling in *Chapter 1*, I discuss this resource in detail in *Chapter 2*. Then, I address the issue of standardization by revisiting the idea of a standardized model formulation for ion channels, and fit all models on the ICGenealogy database to a common formalism. In *Chapter 3*, I discuss the results of this effort.

Another current direction of research, perhaps due to many of the same reasons mentioned above, is a shift away from the single neuron, replaced with the neuronal

network as the fundamental unit of computation in the brain [Yuste, 2015]. It is now state-of-the-art in experimental neuroscience to record from large populations of neurons [Jun et al., 2017], providing a much more global view of circuit dynamics and function. Recent studies focus on network-level phenomena such as mixed selectivity of individual neurons [Rigotti et al., 2013], strong recurrent connectivity leading to complex dynamics [Ozeki et al., 2009], and decoding of neuronal population dynamics through dimensionality reduction techniques [Kobak et al., 2016]. **Part II** of this thesis uses neural network models to study computational aspects of memory.

In *Chapter 4*, I provide an introduction to neural circuits of the brain, neuronal network models, and memory. Then, in *Chapter 5*, I present a network model of context-dependent memory based on the Hopfield network [Hopfield, 1982]. In this model, I attempt to integrate recent findings about local inhibitory microcircuits into a classic model of associative memory. Crucially, context dependence arises as a network-level phenomenon – changing the underlying network state – something which is not apparent at the single-neuron level. Finally, in *Chapter 6*, I end with a short summary of the work presented in this thesis and a brief mention of future directions.

While the text and figures presented in this thesis represent my own work, these projects have been pushed forward through collaborations with several other people who played important roles in conception and results. Tim Vogels advised and provided critical support for each of the studies presented here. In addition, the ICGenealogy database presented in Chapter 2 was done alongside Alex Seeholzer, Rajnish Ranjan, Lukas Groschner and Gero Miesenboeck. The work presented in Chapter 3 was done in collaboration with Chaitanya Chintaluri. Finally, the model presented in Chapter 5 was done with the guidance and support of Everton Agnes.

Part I

Single neuron modelling and ion channels

1

Introduction: biophysical neuron modelling

The fundamental unit of the nervous system has traditionally been considered to be the neuron [Shepherd, 2015]. Neurons are remarkably complex cells, both in structure and in function. Most neurons can be divided into a *soma*, which hosts the nucleus and other organelles, *dendrites*, which receive inputs from other neurons, and an *axon*, which sends outputs to downstream cells [Kandel et al., 2000]. The soma of a typical mammalian cortical neuron ranges from $5 - 20\mu$m, but the total length of the dendrites and axon may be on the order of centimeters (or more), allowing each individual neuron to communicate with thousands of other cells [Kandel et al., 2000]. Neurons are elaborate biological input-output devices, which process inputs from their synaptic partners, and generate large (100 mV amplitude) and short-lasting ($1 - 2$ msec) electrical output pulses called action potentials, or spikes [Gerstner et al., 2014b]. This is accomplished through the intricate arrangement of ion channels in the cell membrane. In this chapter, I briefly introduce the electrical properties of neurons and describe the ion channels that lead to their behavior. I conclude with a discussion on the current status of biophysical modelling, as well as a short word on tools and resources that have been developed to facilitate better practices in computational modelling.

1.1 Electrical properties of neurons

1.1.1 Passive cell properties

All cells have inherent electrical properties endowed to them by the cell membrane, which separates the internal environment of the cell from the outside world. The concentration and flow of different ions and other charged particles across this membrane determines the membrane potential. In the language of electrical circuits, cells can be seen as simple resistor-capacitor (RC) circuits (Fig. 1.1A), in which the membrane acts as a capacitor, connected in parallel with (leak) ion channels that function as a resistor [Gerstner et al., 2014b]. When a cell is driven by an external electrical current $I_{\text{ex}}(t)$, the dynamics of the membrane potential follow

$$C_m \frac{dV(t)}{dt} = \frac{1}{R_m}[V_r - V(t)] + I_{\text{ex}}(t), \tag{1.1}$$

where C_m is the membrane capacitance, R_m is the membrane resistance, and V_r is the *resting potential* of the cell. This can be rewritten as

$$\tau \frac{dV(t)}{dt} = V_r - V(t) + R_m I_{\text{ex}}(t,) \tag{1.2}$$

where $\tau = R_m C_m$ is the membrane time constant. We see in Fig. 1.1B that when external current is given, the cell acts as a "leaky integrator" – current simultaneously charges the capacitor and is released through the resistor, returning to baseline when the stimulus is removed.

The resting membrane potential of most neurons lies around -70mV [Kandel et al., 2000], which is determined by the concentration of ions inside and outside of the cell. The ions that have the strongest effect on neuronal dynamics are potassium (K^+), sodium (Na^+), calcium (Ca^{2+}), and chloride (Cl^-) [Hille et al., 2001]. Typical intracellular and extracellular concentrations of these ionic species are shown in Table 1.1 for the squid giant axon and a typical mammalian neuron. These concentration differences do not come for free – each neuron must constantly expend energy in order to maintain them through the use of ion pumps. One common ion pump is the Na^+-K^+ pump, which requires energy to function, but maintains

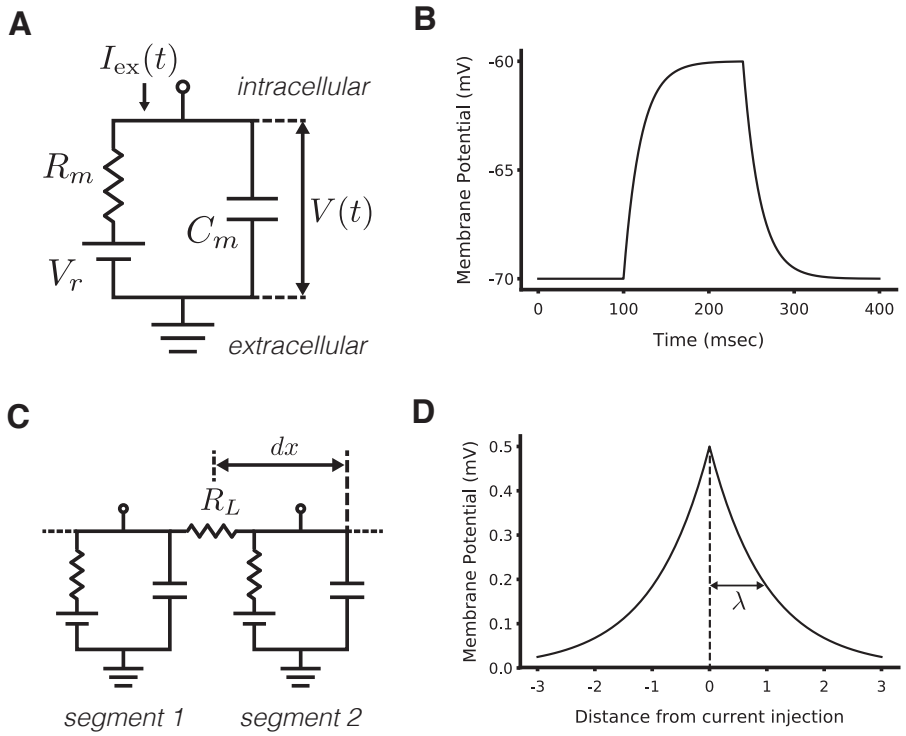


Figure 1.1: Passive properties of the cell membrane. **A**: A piece of membrane can be modelled as an RC-circuit, with capacitance C_m and resistance R_m (connected to a "battery" which determines the resting potential V_r). **B**: The membrane potential response to an injected step current (beginning at 100 msec and ending at 250 msec). **C**: Extended morphology can be modelled by considering small segments of the membrane as RC circuits, connected by a longitudinal resistance R_L. **D**: An injected current in a passive membrane decays exponentially with distance, following a length scale $\lambda = R_m/R_L$. Distance is plotted in normalized units relative to the length scale, and membrane potential is plotted relative to V_r. Adapted from Gerstner et al. [2014b] Figs 1.6, 3.5 and 3.6.

---

the large differences in concentration of sodium and potassium ions across the membrane needed for action potential generation (Table 1.1; Kandel et al. [2000]).

Considering a particular ion, we can model the effect that this concentration (and charge) difference has on the cell using the so-called *Nernst equation* [Hille et al., 2001], which describes the potential (E) across the membrane due to this ion when the system is at thermodynamic equilibrium

$$E = \frac{RT}{zF} \ln \frac{[C]_o}{[C]_i}, \tag{1.3}$$

where R is the universal gas constant, T is the temperature in kelvin, F is the

Faraday constant, z is the charge of the particular ion species, and $[C]_o$ and $[C]_i$ are the concentrations of the ion outside and inside of the cell, respectively. The variable E is called the reversal potential of the ionic species – ions flow so as to move the membrane potential towards this value, meaning that this flow will reverse direction at E. Reversal potentials for different ionic species are also shown in Table 1.1.

As already mentioned, several ionic species play important roles in neurons, each with different concentrations and different permeabilities across the membrane. We can determine the concurrent effects of each species together, which determines the *resting potential* of the cell, using the Goldman-Hodgkin-Katz (GHK) equation [Hille et al., 2001]

$$E = \frac{RT}{F} \ln \frac{P_{Na}[Na^+]_o + P_K[K^+]_o + P_{Ca}[Ca^{2+}]_o + P_{Cl}[Cl^-]_i}{P_{Na}[Na^+]_i + P_K[K^+]_i + P_{Ca}[Ca^{2+}]_i + P_{Cl}[Cl^-]_o}, \tag{1.4}$$

where P_{ion} is the relative permeability of each ionic species. The permeability of an ionic species reflects the ease at which that ion crosses the membrane, often measured as a conductance (in siemens). Here, since each term in the logarithm contains a permeability, they can also be expressed as relative permeabilities (unitless). At rest, most of the leak channels in a typical neuron are permeable to K^+, and so the permeability of potassium will generally be much larger than that of sodium, calcium, or chloride.

Given the complex, extended morphology of many neurons, electrical current can also flow within the neuron from the source of a current input to other areas. This simply modularizes the electrical circuit diagram from before, with each area

	Squid Giant Axon			Mammalian Neuron		
	Conc. in (mM)	Conc. out (mM)	E_{rev} (mV)	Conc. in (mM)	Conc. out (mM)	E_{rev} (mV)
K^+	400	20	-75	140	5	-87
Na^+	50	440	55	5 - 15	145	60 - 90
Cl^-	52	560	-60	4 - 30	110	-90 - -60
Ca^{2+}	0.0001	10	145	0.0001	1 - 2	120 - 130

Table 1.1: Concentrations of the major ionic species across the cell membrane in the giant axon of the squid (left) and a mammalian neuron (right). Reversal potentials for each ion are given at room temperature (25°C) for the squid and physiological temperature (37°C) for the mammalian neuron. Values from Purves et al. [2014] and Kandel et al. [2000].

of the neuron as an RC circuit connected with other modules through a longitudinal resistance R_L (Fig. 1.1C). The passive spread of current through a piece of dendrite can generally be described by the so-called *cable equation* [Gerstner et al., 2014b]

$$\frac{\partial}{\partial t}v(t,x) = \frac{\partial^2}{\partial x^2}v(t,x) - v(t,x) + i_{\text{ext}}(t,x), \tag{1.5}$$

where t and x are normalized time and distance (unitless), $v(t,x)$ is the membrane potential (relative to the resting potential), and $i_{\text{ext}}(t,x)$ is external input current. As shown in Fig. 1.1D, input current decays exponentially along the passive dendrite according to a particular length scale $\lambda = R_m/R_L$.

With these mathematical descriptions, we are able to fully describe the passive properties of a morphologically-complex neuron. But this does not explain how neurons are able to produce nonlinear responses such as action potentials. For that we need to describe their active properties.

1.1.2 Active properties: biophysical models of ion channels

Neuronal action potentials (APs) are generated through the selective permeability of the membrane to sodium and potassium ions, the mechanisms of which were described by Hodgkin and Huxley in the early 1950s [Hodgkin and Huxley, 1952a]. A few early studies provided clues towards this end, such as the discovery that ion conductance increases during action potential generation [Cole and Curtis, 1939], and that changing the external sodium concentration affects AP height [Hodgkin and Katz, 1949]. Around this time, due to the relative positions of the reversal potentials of sodium and potassium (Fig. 1.2A), it was hypothesized that the rise in the action potential was due to a temporary large increase in the permeability to sodium, followed by a reset due to permeability to potassium (Fig. 1.2B).

Action potential generation and the Hodgkin-Huxley model

In the early 1950s, Alan Hodgkin and Andrew Huxley performed a series of experiments using the squid giant axon to fully characterize the dynamics of these conductances [Hodgkin and Huxley, 1952a]. By varying ionic concentrations, in

concert with voltage-clamp protocols, they were able to measure the time course and steady state conductances of potassium and sodium currents separately.

Hodgkin and Huxley modelled each ion channel as a set of gating variables, which determine the conductance for each ion channel type. Generally, an ion channel can be in an *open* or *closed* state – this can be described by an *activation* gating variable, representing the probability of this channel type being in the open state. The sodium and potassium activation variables, normally referred to as m and n, respectively, are shown in Fig. 1.2B during the time-course of an action potential. Some ion channels also have an inactive state, in which even an open channel will not allow ions to flow through. This is modelled using an *inactivation* gating variable, representing the probability of a channel type to be in the inactivated state. The sodium inactivation variable, h, is also shown in Fig. 1.2B (the potassium current does not inactivate in this case).

The resulting model, now known as the Hodgkin-Huxley (HH) model, describes the membrane potential, $V(t)$, using the following four equations

$$C\frac{dV(t)}{dt} = -g_L(V(t)-V_r) - g_{Na}m^3h(V(t)-E_{Na}) - g_Kn^4(V(t)-E_K) + I_{\text{ex}}(t), \tag{1.6}$$

$$\frac{dm(t)}{dt} = \alpha_m(V)(1-m) + \beta_m(V)m, \tag{1.7}$$

$$\frac{dh(t)}{dt} = \alpha_h(V)(1-h) + \beta_h(V)h, \tag{1.8}$$

$$\frac{dn(t)}{dt} = \alpha_n(V)(1-n) + \beta_n(V)n, \tag{1.9}$$

where $\alpha_x(V)$ and $\beta_x(V)$ correspond to the transition rates between open and closed (or non-inactivated and inactivated) states, respectively, for each gating variable x. The equations for the gating variables can also be rewritten using a steady-state and time constant formulation as

$$\tau_x(V)\frac{dx}{dt} = x_\infty(V) - x(t), \tag{1.10}$$

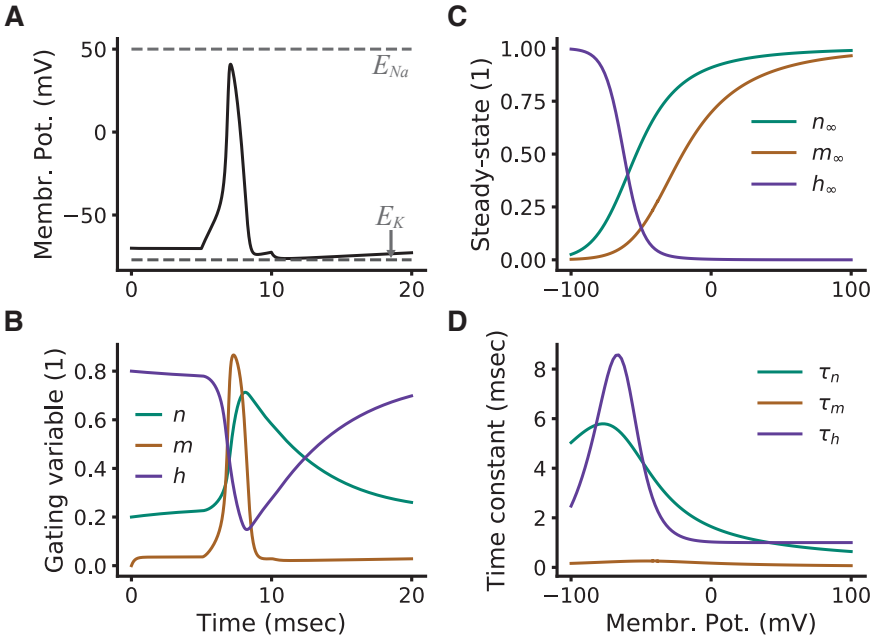


Figure 1.2: The Hodgkin-Huxley model of action potential generation. **A,B**: Membrane potential (A) and ion channel gating variable responses (B) to an injected step current (from 5−10 msec). First, the sodium gate activates (m, brown) and the membrane potential moves rapidly towards the sodium reversal potential E_{Na}. Then sodium inactivates (h, purple) followed by activation of potassium (n, green), and the membrane potential returns towards the potassium reversal potential E_K. **C,D**: Steady-state activation (and inactivation) curves (C), and time constant curves (D) as a function of voltage for the Hodgkin-Huxley sodium and potassium currents.

---

where $x_\infty(V) = \alpha_x(V)/[\alpha_x(V) + \beta_x(V)]$ and $\tau_x(V) = 1/[\alpha_x(V) + \beta_x(V)]$. These gating variables exhibit different steady-state responses and time constants depending on the membrane potential – these are plotted in Fig. 1.2C,D for the three gating variables of the HH model.

In this work, Hodgkin and Huxley provided a powerful framework to model arbitrary ion channel dynamics, which has been extensively used ever since. In *Chapters 2 and 3*, I study the diversity of HH models that have been created for neuronal simulations over the past several decades.

Thermodynamic models

The form of the transition rates in the Hodgkin-Huxley model can be determined using thermodynamic considerations from the theory of reaction rates [Tsien and

Noble, 1969, Stevens, 1978, Hill and Chen, 1972, Destexhe and Huguenard, 2000b]. I briefly summarize this formalism here, following Destexhe and Huguenard [2000b] and Destexhe and Huguenard [2000c].

We consider an ion channel which transitions between open and closed states

$$\mathrm{C} \underset{\beta(V)}{\overset{\alpha(V)}{\rightleftharpoons}} \mathrm{O}. \tag{1.11}$$

The rate of transition, $r(V)$ (corresponding to $\alpha(V)$ and $\beta(V)$ in Eqn. (1.11)), depends exponentially on the free-energy barrier between the two states

$$r(V) = r_0 \exp(-\Delta G(V)/RT), \tag{1.12}$$

where r_0 is a constant and $\Delta G(V) = G^*(V) - G_0(V)$ is the free-energy barrier, which can be written as the difference between the free energy at an intermediate state $G^*(V)$ and an initial state $G_0(V)$.

The form of the free-energy dependence on voltage can be approximated using a Taylor series expansion of the following form $G(V) = A_i + B_i V + C_i V^2 + ...$, leading to rate equations written as

$$r(V) = r_0 \exp\Big(- [(A^* + B^* V + C^* V^2 + ...) - (A_0 + B_0 V + C_0 V^2 + ...)]/RT\Big), \tag{1.13}$$

$$= r_0 \exp[-(a + bV + cV^2 + ...)/RT]. \tag{1.14}$$

Applying this to the open-closed rate equations above, we arrive at a general *nonlinear thermodynamic model*

$$\alpha(V) = \alpha_0 \exp[-(a_1 + b_1 V + c_1 V^2 + ...)/RT], \tag{1.15}$$

$$\beta(V) = \beta_0 \exp[-(a_2 + b_2 V + c_2 V^2 + ...)/RT]. \tag{1.16}$$

The number of terms included in this expansion depends upon the range of relevant membrane potential values and desired accuracy. Often, a linear dependence is assumed [Tsien and Noble, 1969, Borg-Graham, 1991], which accounts for free energy that is independent of the electric field, and effects of isolated charges and rigid dipoles [Destexhe and Huguenard, 2000b]. In that case, only parameters a_1,

a_2, b_1 and b_2 are nonzero. The nonlinear effects of higher-order terms may account for electronic polarization and pressure and mechanical constraints.

Putting this nonlinear thermodynamic model into the form of steady-state and tau curves (as in Eqn. (1.10)), we arrive at the following form

$$x_\infty(V) = \frac{1}{1 + e^{-[a+bV+cV^2+...]}}, \tag{1.17}$$

$$\tau_x(V) = \frac{A}{e^{-[a_1+b_1V+c_1V^2+...]} + e^{[a_2+b_2V+c_2V^2+...]}}, \tag{1.18}$$

where the steady-state curve only includes the first two terms of the Taylor expansion, and the constants α_0, β_0, R and T have been absorbed into the other parameters.

This general form can in principle be used to capture the dynamics of arbitrary voltage-dependent conductances. In *Chapter 3*, I explore the ability of a model of this form to fit and capture the dynamics of a dataset of ion channel models used in neuronal simulations.

Markov models

The Hodgkin-Huxley model, and the thermodynamic formalism in general, capture the physiology of neurons in many experimental paradigms. However, they ignore the inherent stochasticity of the opening and closing transitions of real channels [Neher and Sakmann, 1976, Colquhoun and Hawkes, 1981]. They also ignore dependencies between activation and inactivation dynamics [Armstrong, 1981, Aldrich et al., 1983], such as the fact that sodium channel inactivation is more likely to occur when the channel is open. This motivates the introduction of *Markov models*, which formulate ion channel gates as a series of state transitions between n different conformations

$$\mathrm{S}_1 \underset{P_{21}}{\overset{P_{12}}{\rightleftharpoons}} \mathrm{S}_2 \underset{P_{32}}{\overset{P_{23}}{\rightleftharpoons}} \cdots \underset{P_{n(n-1)}}{\overset{P_{(n-1)n}}{\rightleftharpoons}} \mathrm{S}_\mathrm{n}, \tag{1.19}$$

where S_i denotes state i, and P_{ij} is the transition probablility from state i to state j. In the limit of a large number of identical channels, as found in the cell membrane of neurons, this can be interpreted as describing the dynamics of the fraction of

channels in each state, and transition probabilities P_{ij} become rate constants r_{ij}, as described in Eqn. (1.11). The dynamics of the system can then be written as

$$\frac{ds_i}{dt} = \sum_{j=1}^{n} s_j r_{ji} + \sum_{j=1}^{n} s_i r_{ij}, \tag{1.20}$$

where s_i represents the fraction of channels in state i.

Markov models are a more general formalism for ion channel dynamics, with the Hodgkin-Huxley model as a special case. For example, consider a model of a Hodgkin-Huxley-style sodium channel [Vandenberg and Bezanilla, 1991])

$$\begin{array}{ccccccccc} C & \underset{r_6}{\overset{r_5}{\rightleftharpoons}} & C_1 & \underset{r_6}{\overset{r_5}{\rightleftharpoons}} & C_2 & \underset{r_6}{\overset{r_5}{\rightleftharpoons}} & C_3 & \underset{r_2}{\overset{r_1}{\rightleftharpoons}} & C_4 \\ & & & & & & r_8 \uparrow \downarrow r_{10} & & r_7 \uparrow \downarrow r_9 \\ & & & & & & I_1 & \underset{r_3}{\overset{r_4}{\rightleftharpoons}} & I_2 \end{array} \quad \begin{array}{c} \underset{r_4}{\overset{r_3}{\rightleftharpoons}} \; O \\ \\ \underset{r_1}{\overset{r_2}{\rightleftharpoons}} \; I_3. \end{array} \tag{1.21}$$

While this model, with nine states, is considerably more complicated than the Hodgkin-Huxley formulation (which would only require two state variables), it can capture both macroscopic current responses as Hodgkin and Huxley recorded, as well as single-channel properties.

Which formalism to use?

Markov models are the most flexible and biophysically accurate model formalism for ion channel dynamics. However, as shown in the previous section, they can become quite computationally expensive depending upon the complexity of the model [Fink and Noble, 2009]. The choice of which formalism to use depends largely on what behavior the model is intended to reproduce. The three formalisms discussed here all perform qualitatively similarly when attempting to reproduce action potential generation [Carbonell-Pascual et al., 2016], but a more complex Markov model [Vandenberg and Bezanilla, 1991] outperforms the simpler models when reproducing behavior in voltage-clamp or at the single channel level [Destexhe and Huguenard, 2000c]. In this thesis, I focus on simplified models of the Hodgkin-Huxley and thermodynamic form, primarily for the reason that these models are more typically used in neuronal simulations (as empirically seen in model entries on ModelDB; Hines et al. [2004]).

1.1.3 Capturing the diverse response properties of neurons

The motivation behind the results presented in the following two chapters is to facilitate better experimentally-constrained modelling of single neurons and networks. To provide a context for this work, I briefly describe here some general characteristics of biophysically detailed models as well as phenomenological spiking models. I refer the reader to Koch and Segev [1998] and Gerstner et al. [2014b] for more information.

Diversity of ion channels

It is now clear that in addition to the sodium and potassium channels responsible for action potential generation, many neurons contain additional ion channel conductances which modulate input integration and output generation in diverse ways [Catterall, 1995, Yu and Catterall, 2004]. Some ionic currents act to *depolarize* the cell, bringing it closer to action potential generation, whereas others *hyperpolarize*, bringing the membrane potential to more negative values. Importantly, these ion channels may also be spread non-uniformly around the morphology of a neuron, having interesting effects on its physiology [Hille et al., 2001].

The most common sodium ion channels, often called the transient (NaT) or fast currents (NaF), are responsible for the fast upswing of the action potential. These channels are immensely important for neuronal signaling, and it is perhaps for this reason that there is not much diversity in voltage-gated sodium ion channel behavior, as compared with other ionic species [Hille et al., 2001]. Nevertheless, some qualitatively different sodium currents are found in the nervous system, such as a non-inactivating (persistent) sodium current (NaP), which activates at more hyperpolarized membrane potential values compared with the transient current. This current may increase the efficacy of synaptic inputs at distal dendrites [Crill, 1996].

In contrast, a diversity of potassium ion channels have been discovered and characterized. Many of these have fast kinetics and contribute to the downswing of the action potential. Others with slower dynamics and inactivation have other functions. One prominent subtype with fast inactivation, often called the A-type current, can serve to increase the input-output bandwidth of some neurons [Koch,

2004]. Additionally, this subtype is often found in distal dendrites of mammalian neurons and contributes to dendritic integration and control of action potential backpropagation [Hoffman et al., 1997, Migliore et al., 1999]. Another potassium subtype with slow activation, called the M-type current, can control the excitability of neurons [Aiken et al., 1995]. Inward rectifier potassium channels have been linked to regulatory roles relating to maintenance of the resting membrane potential [Hibino et al., 2010]. Finally, calcium-dependent potassium (KCa) currents may contribute to spike-frequency adaptation [Lancaster and Adams, 1986].

Calcium ion channels are also diverse, and play a special role in the nervous system, as calcium has been linked to many intracellular processes such as neurotransmitter release and plasticity [Sjöström and Nelson, 2002]. Though calcium channels may have many regulatory and second-messenger roles, they may also contribute directly to the electrochemical membrane physiology of neurons, and I briefly mention two subtypes here (following Koch [2004]). The high-threshold, L-type channel is often found around the soma and proximal dendrites and plays important roles in regulating gene expression and plasticity [Lipscombe et al., 2004]. The low-threshold T-type channel was found to have a critical contribution to the two modes of operation of thalamic neurons – relay and burst [Jahnsen and Llinas, 1984].

Finally, there are also voltage-gated channels which are permeable to both potassium and sodium ions, and so are called cation currents [Hille et al., 2001]. Many of these are called hypoerpolarization-activated (IH) currents, because they open at hyperpolarized potentials (compared to the resting potential). These channels can cause slow depolarization of the membrane potential, and contribute to rhythmic and burst firing [McCormick and Pape, 1990]. Interestingly, some of these ion channels are directly modulated by cyclic AMP, which shifts the voltage-dependence [DiFrancesco and Tortora, 1991].

In addition to the physiological characterization of single ion channel currents, it is often the case that groups of currents will have nontrivial effects on neuronal responses. One example of this is single-neuron resonance, in which a neuron will display an intrinsic frequency preference for its input [Hutcheon and Yarom, 2000].

Resonance often arises through the interaction of passive cell properties with certain currents that act as "high-pass filters" (such as IH and certain K currents), and may be further amplified by persistent sodium or L-type calcium currents.

Biophysically detailed single neuron models

Given the diversity of ion channels found in neurons, it is no surprise that many models incorporate several ionic conductances in various distributions across a detailed morphology in order to achieve responses similar to experimental studies [Mainen and Sejnowski, 1995, Stuart and Spruston, 1998, Migliore et al., 1999, Destexhe and Paré, 1999]. Large-scale efforts from the Blue Brain Project and the Allen Institute have created experimentally-constrained mammalian neurons which exhibit a large range of desired properties, and have realistic morphologies and distributions of synapses [Markram et al., 2015, Allen-Institute, 2015].

Following Gerstner et al. [2014b] (section 3.4), I mention one example here from Hay et al. [2011] of a layer 5b pyramidal cell. This model has 200 compartments, capturing the intricate morphology of the neuron, along with ten active conductances: two sodium currents (NaT and NaP), four voltage-gated potassium currents (including DR and M-type), one calcium-gated potassium current (KCa), two calcium currents (L-type and T-type), and a nonspecific cation current (Ih). Some of these currents are distributed non-uniformly along the dendrites. This model can reproduce several interesting features of experimental data, such as bursting and dendritic spiking.

Phenomenological spiking neurons

Detailed neuron models with active conductances and morphology can reproduce many interesting effects seen in experiments, but their complexity limits analytical tractability and use in large-scale network simulations. Phenomenological spiking neuron models have become popular due to their simplicity [Dayan and Abbott, 2001, Gerstner et al., 2014b]. The simplest of these, the leaky integrate-and-fire model [Lapique, 1907], relies on two simplifications about neuronal spike generation. First of all, the short duration and stereotypical shape of the action potential

suggests that it can be abstracted away and replaced with a single spike time and a reset to some baseline value. Second of all, the membrane potential of most neurons behaves relatively linearly below a depolarized "threshold" value where the sodium channels begin to open. Thus, the membrane potential is well approximated by a passive RC circuit. By combining these two simplifications, we arrive at the leaky integrate-and-fire model

$$\tau_m \frac{dV(t)}{dt} = -(V(t) - V_r) + I(t), \tag{1.22}$$

$$\text{if } V(t) > \theta : V(t) \leftarrow V_{\text{reset}} \quad \text{(spiking condition)}, \tag{1.23}$$

where the threshold θ and reset potential V_{reset} are free parameters which can be fit to data.

Since its original formulation, several additions have been added to the integrate-and-fire neuron in order to better fit experimental data, such as an exponential threshold onset [Fourcaud-Trocmé et al., 2003] and an auxillary adaptation variable [Brette and Gerstner, 2005].

Which single-neuron model to use?

Similarly to choosing between the different ion channel model formulations, the choice of which single neuron model to use depends upon the complexity of the simulation and the desired accuracy with experimental results [Herz et al., 2006]. Counterintuitively, sometimes simpler models may be better able to reproduce experimental results, such as fast spike initiation [Gerstner and Naud, 2009, Brette, 2015b]. However, recent computational work on the function of dendrites is making the case that complex currents and compartmentalization may be essential for the understanding of single-neuron computation (e.g., Sacramento et al. [2017]).

1.2 Tools and resources in neuroinformatics

Computational models of all scales and complexity have been published for several decades, with the help of different model formulations, simulators, and

data structures [Brette et al., 2007]. However, a lack of standardization in the field makes it difficult to makes comparisons between different models and to reproduce results [McDougal et al., 2016].

Several resources have been introduced in order to aid model curation, maintenance and analysis. The online repository *ModelDB* hosts thousands of models, complete with annotated code and searchable metadata [Hines et al., 2004]. It is becoming more and more common for code to be deposited here, as well as GitHub, following the publication of a computational study. *Open Source Brain* is another initiative which aims to standardize and make available models across computational neuroscience, as well as providing visualization tools [Gleeson et al., 2018]. This initiative has specifically pushed for a simulator-independent model description language called NeuroML, which aims to provide a universal format for all models [Gleeson et al., 2010].

Other more specialized repositories host data for specific components of computational models or experimental data. *NeuroMorpho* is a repository which hosts morphologies of detailed single-neuron models [Ascoli et al., 2007]. *NeuroElectro* is a database which provides physiological data about different neuron types [Tripathy et al., 2014]. Finally, larger initiatives such as the Blue Brain Project and Allen Institute provide detailed information about the morphology and physiology of a wide range of cell types based upon experimental data, as well as corresponding neuron models [Markram et al., 2015, Allen-Institute, 2015].

In the following chapter of this thesis, I introduce a new database and interactive web browser specifically designed for neuronal ion channel models, called *ICGenealogy* [Podlaski et al., 2017].

2

Mapping the function of neuronal ion channels in model and experiment

Note

A version of this chapter has been published as [Podlaski et al., 2017]. The text and figures in this chapter have been modified from the published paper and represent my own work.

Abstract

Ion channels are fundamental for the function of single neurons and neuronal circuits, and computational modeling has become essential in understanding their complex interactions. However, the lack of standardization makes it difficult to interpret and compare the thousands of ion channel models that have been used as part of neuronal simulations. Here, we present a framework within which we analyzed a pilot set of 2378 voltage- or calcium-dependent ion channel models available for NEURON. Using annotated metadata and voltage-clamp protocols with cluster analysis, we built topologies of ion channel similarity and genealogy. This resource allows for quantitative comparison of ion channel kinetics in experiments and models alike, and is aimed to facilitate field-wide standardization of experimentally-constrained modelling.

2.1 Introduction

Neuronal activity is mediated by ion channels – transmembrane proteins that selectively allow the flow of ions into and out of the cell according to electrochemical gradients [Hille et al., 2001]. A great diversity of ion channel subtypes exists, including a variety of more than 140 subgroups in the voltage-gated ion channel superfamily, many of which are crucial for nervous system function [Yu and Catterall, 2004]. Ion channel subtypes and distributions vary across brain areas and neuron types, as well as across the different functional regions of a neuron [Catterall, 1984, Trimmer and Rhodes, 2004, Lai and Jan, 2006, Vacher et al., 2008], to enable specialization according to different signal processing tasks [Koch and Segev, 2000, Cai et al., 2004, Goldberg et al., 2008] and plasticity mechanisms [Sjöström and Nelson, 2002, Shah et al., 2010, Debanne et al., 2003]. Not surprisingly, ion channel mutations are also implicated in many disorders of the nervous system, such as epilepsy and migraine [Catterall et al., 2008, Lehmann-Horn and Jurkat-Rott, 1999].

The interaction of different ion channels can lead to complex dynamics, and computational models are often used to disentangle the effects of combined ionic currents on the underlying membrane potential of a given neuron. First utilized by Hodgkin and Huxley [1952a] to demonstrate action potential generation as the joint effort of sodium and potassium currents, biophysical models have become widely used [Brette et al., 2007, Gerstner et al., 2014a, Koch, 2004]. The *NEURON* simulation environment [Carnevale and Hines, 2006, Hines and Carnevale, 2001] has established itself as one of the preferred tools for biophysical modeling. It provides a platform to simulate customizable biophysical neuron models with complex, realistic morphologies together with plausible ion channel dynamics. This approach to detailed modeling facilitates the integration of experimental results to further explore the dynamics and computational properties of biologically plausible neurons and neural networks in silico [Mainen and Sejnowski, 1995, Stuart and Spruston, 1998, Migliore et al., 1999, Poirazi et al., 2003, Destexhe and Paré, 1999, Traub et al., 2003]. A widely-adopted archive for neuronal models is *ModelDB* [Hines et al., 2004], which currently contains over 2000 ion channel models as part of

its single-neuron and network model entries. It thus represents a highly diverse and powerful resource for researchers looking to integrate published models of ion channel kinetics into their own models.

Generally, matching model and experiment is essential in order draw any valid conclusions or predictions [Brette et al., 2007] – this is especially true for detailed biophysical models, in which many components have a direct biological counterpart. As an example, consider a pyramidal neuron model which is able to reproduce spiking behavior seen in experiments (Traub et al. [2003]; Fig. 2.1A, gray traces; see methods). However, these dynamics can change when one of the modeled ion channel currents is exchanged for an identically-labeled model from a different publication on ModelDB (Fig. 2.1A, colored traces). This is meant to illustrate the fact that the choice of model can sometimes be crucial for biological fidelity – yet there is no standardization by which to compare them with one another and with experiments.

ModelDB, as a pure repository of models, suffers from similar issues. The increasingly large number of models in the database, as well as non-standardized labeling and a high redundancy of ion channel models that are entirely or nearly identical, make it difficult to understand how models relate to biology and to determine which components are suitable for reuse. For example, a researcher looking to use an existing *A-type* potassium channel model will find that there are some 238 *A-type* models that exist on *ModelDB*, coming from both old and more recent publications, and spanning a range of behaviors (Fig. 2.1B, blue). Instead of a thorough and time-consuming fitting of appropriate ion channel dynamics, it is common for modelers to adapt previously published ion channel models for their own purposes. However, this may introduce experimentally unverified systematic changes or even errors into later generations of models and may have dramatic effects on the biological interpretation of the results.

A broad comparison between experimental datasets and models to determine their validity seems hardly possible given the lack of standardization at this point in time. We chose here, instead, to assay and categorize the kinetic variability and metadata of published ion channel models. This approach has two benefits.

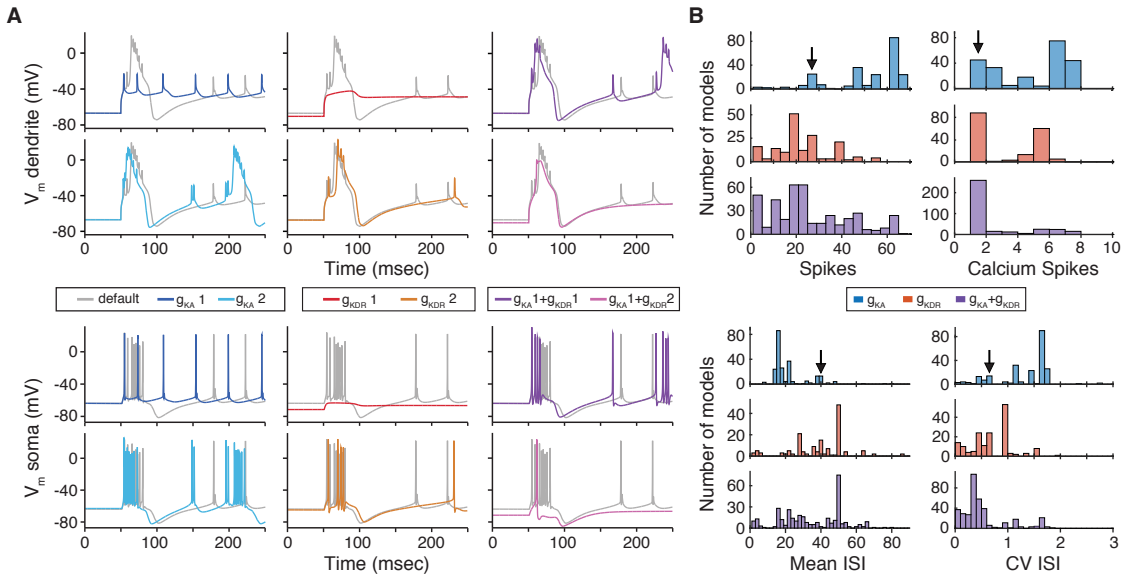


Figure 2.1: The choice of ion channel model influences the behavior of a simulated neuron. **A**: Simulation of a detailed layer 2/3 pyramidal neuron model, adapted from Traub et al. [2003] (see Methods for details). The model was stimulated with a 1.5nA current step beginning at 50msec, while recording the membrane potential in the apical dendrite (top) and soma (bottom). Simulations were first run using the original conductances from Traub et al. ("default", gray), as well as replacement conductances for the A-type potassium model (left, two different replacements, light and dark blue), delayed rectifier potassium model (middle, red and orange), or both (right, purple and magenta). **B**: Model from **A** was simulated for 1000msec with a 1.5nA current step with one of 243 A-type-labeled models (blue), one of 188 delayed rectifier-labeled models (red), or combinations of both (purple). Summary measures are shown for total number of spikes, total number of calcium spikes, mean inter-spike interval (ISI) and coefficient of variation (CV) of ISI during the 1000msec period. Black arrows represent the simulation results for the default model.

First, it provides a tool for modellers looking to incorporate existing models into their simulations. Second, by providing methods to measure the similarity amongst models and experiments in the same topology, we provide a framework for future large scale comparisons with standardized experiments.

To this end, we performed an exhaustive analysis of all major voltage-dependent and calcium-dependent ion channel models in the NEURON language available on ModelDB. In a first step, we catalogued the associated literature, taking note of various metadata, including relations between channel models that were derived from each other (pedigree). In a second step, we performed a comparison of ion channel model kinetics through standardized voltage clamp protocols and subsequent cluster

analysis of the resulting current responses. The topologies created in this way allow us to draw conclusions about the amount of model variability and diversity, as well as the computational and experimental origins of each model.

In doing so, we reduced the number of ion channel models with unique response profiles by approximately 50%, from 2378 to 1132, and clustered the remaining models into 211 different clusters, with a representative model for each one. We thus dramatically reduced the number of choices for a modeler looking to incorporate new channels into their models, without reducing the variety of the available channels.

Our findings are presented in an annotated, interactive web-interface [ICGenealogy, 2017], which we hope will act as a complement to existing resources such as ModelDB. It allows for filterable search of individual ion channel models and the comparison of channel models by the similarity of their response kinetics. The underlying database contains metadata for each channel model, relational information between models, and quantitative information reflecting the dynamics of the models, and is openly accessible by an API.

In the effort to make our resource compatible with experimental data and new models, we offer the possibility to upload and assess the similarity of experimentally recorded current traces (as well as new model traces) in the same topology. We show an example of the use of this comparison through the analysis of an unclassified ion channel model, as well as an experimentally recorded voltage-dependent potassium current from *Drosophila melanogaster*. We thus provide a framework for the direct comparison of experiments and models to facilitate experimentally constrained modeling and eventually enable broad comparisons between experiment and published models.

2.2 Results

2.2.1 Ion channel model metadata and ancestor-descendant relationships

We performed an analysis of 2378 voltage- and calcium-dependent ion channel models (".mod" files) in the *NEURON* simulation environment [Hines and Carnevale,

2001, Carnevale and Hines, 2006] available for download on *ModelDB* [Hines et al., 2004]. In a first step, we manually extracted "metadata" information from the journal articles associated with the 395 relevant ModelDB entries (Fig. 2.2A) – reference information (Ref. Info, including author(s) of the model code), ion channel information (I.C. Info: ion selectivity, gating mode, subtype), system information (Sys. Info: brain area, neuron type, neuron region, animal model), as well as additional comments (Other: e.g. temperature constraints, see Methods). Based on ion type and gating dependence, the majority of all ion channel models fell into five classes (Fig. 2.2B): voltage-dependent potassium (Kv), voltage-dependent sodium (Nav), voltage-dependent calcium (Cav), calcium-dependent potassium (KCa) and hyperpolarization-activated cation (Ih) channel models.

Despite standardized ion channel nomenclature [Ashburner et al., 2000, Chandy, 1991, Ertel et al., 2000, Goldin et al., 2000, Ranjan et al., 2011], labels are not consistently used in model annotations. We took note of the subtypes as specified in the journal article or model file itself, and arrived at a cumulative number of 74 (32 Kv, 19 Nav, 20 Cav, 11 KCa, 5 Ih) individual subtype labels (Fig. 2.2B). Note that these labels do not necessarily correspond to distinct ion channel subtypes in a physiological or genetic sense. Instead, we simply consider them as labels, allowing for the possibility of overlap or redundancy between different names. For example, voltage-dependent potassium channel models (931 models) were split into *A-type*, *delayed rectifier (dr)*, *Hodgkin-Huxley (HH)*, *m-type*, *not specified*, *slow*, *fast*, *inward rectifier (IR)*, *D*, *2*, *leak*, and 21 other less frequent types. Other ion type classes followed similarly heterogeneous subtype label specifications (cf. Suppl. Table 2.1). Additional metadata categories also showed a broad variety across channel models. The most commonly emulated cell types were *pyramidal*, followed by *interneuron*, *granule cell*, and *basket cell* (Fig. 2.2C). Prominent brain areas included *hippocampus*, *cortex*, *basal ganglia* and *cerebellum* (Fig. 2.2D).

We also established family relations between channel model entries (Fig. 2.2A, bottom, Fig. 2.3). Model A was registered as a "descendant" of an older "ancestor" model B if the publication corresponding to A cited the publication for B as the

source of its channel dynamics. In cases of missing or vague heritage, we tried to infer information from the model files and the *ModelDB* entry. A connection was also recorded if two files displayed a sufficient resemblance to one another (Methods). We denoted the groups of models connected by these relations as "families".

Family relations can be visualized in genealogical trees (Fig. 2.3 for Kv, see suppl. figures for other ion type classes). A prominent feature of the genealogical trees is the presence of large families, often with a highly-cited hub model that has

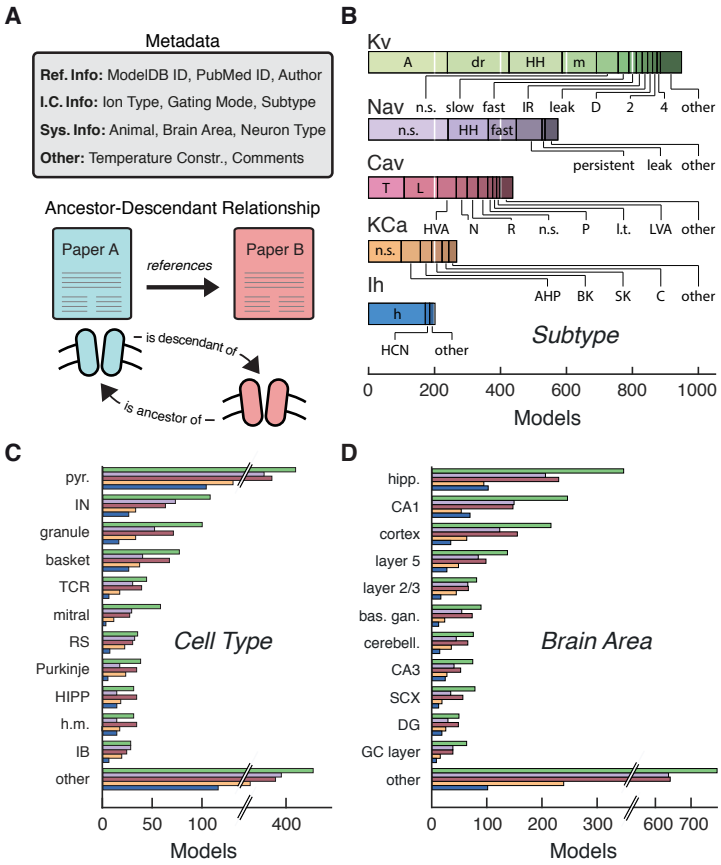


Figure 2.2: Ion channel models can be categorized by metadata and ancestor-descendant relationships. **A**: Metadata were manually extracted from ModelDB and associated journal articles (top). Ancestor-descendant relationship (bottom) was established between different models (see main text for description). **B**: Models were divided into five classes based on ion type: voltage-dependent potassium (Kv), voltage-dependent sodium (Nav), voltage-dependent calcium (Cav), calcium-dependent potassium (KCa), and hyperpolarization-activated cation (Ih). Each class is divided into subtypes, ordered from left to right according to group size. Uncommon subtypes are grouped together (other). **C, D**: Histogram of cell types and brain areas for each ion type, ordered from top to bottom by the number of models. Colors as in **B**.

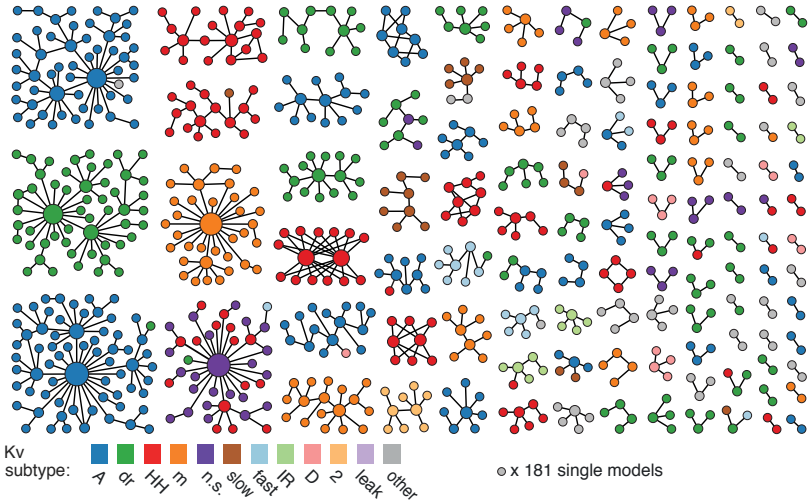


Figure 2.3: Graph of ancestor-descendant relationships for Kv models. Pedigree graph displaying families of the Kv class, sorted by family size. Each node represents one model, colored by subtype, and edges represent ancestral relations between models (Fig. 2.2, panel **A**, bottom). The size of each node is scaled according to the number of edges. A total of 750 models are displayed, with unconnected models (181 total) not shown.

---

many descendants. All five ion type classes contain at least one family of 30 or more models. On the other hand, there are at least 45 families in each ion type class with 3 or fewer members. The large amount of isolated single models and small families implies that many models have been created *de novo*, i.e., without directly using a formalism and parameters from any previous model. Alternatively, models may have been translated from other simulation environments into *NEURON* and thus only appear to have no ancestor (see website metadata *comments*, ICGenealogy [2017]).

The pedigrees featured a mix between consistent and sometimes heterogeneous subtype labels across the genealogical trees. For example, the Kv class (Fig. 2.3) comprised several large families with homogeneous subtype labeling: two families of mostly *A-type* models (67 and 54 members each), one family of *delayed rectifier* models (53 members), one family of *m-type* models (35 members), and a family containing *HH* and *not specified* labels (40 members). However, some families featured several different subtypes, while the same subtype could be found across several different families. For example, for *A-type* channels alone (238 models), there were over 10 families with *A-type* as the majority subtype label

and 42 families featuring at least one *A-type* labeled model. Other ion type classes followed similar patterns.

2.2.2 Assessing model similarity through voltage clamp protocols and clustering

Ion channel model metadata and family relations cannot convey the functional similarity between models, nor potential modifications to a particular model descendant. To quantify the relationships between all channel models, we used a set of voltage-clamp simulation protocols. These and similar protocols were originally developed for the experimental characterization of real ion channels [Hodgkin and Huxley, 1952a, Willms et al., 1999, Ranjan et al., 2011]. In the context of channel models, the protocols aim to assess the spectrum of possible dynamics in a model-free manner, i.e. without explicitly taking into account the equations underlying their implementation. This allows for the comparison of models across different levels of precision, and, as we discuss later, the direct comparison with experimental data.

Using the *NEURON* programming environment [Carnevale and Hines, 2006, Hines and Carnevale, 2001], each ion channel model was placed individually into a model soma with predefined parameter specifications (Methods), and its current responses to five voltage-clamp protocols were recorded (Fig. 2.4A). The protocols were designed to probe the opening and closing characteristics of ion channels, i.e. activation, deactivation and inactivation, as well as temporal dynamics during voltage ramping and repeated action potentials. Protocol parameters were adjusted for each of the five ion type classes to accommodate for differences in reversal potential, time constants, and the voltage range in which each class is known to be active – no quantitative comparisons were made between classes. Calcium-dependent potassium channel models were simulated with seven different internal calcium concentrations (see Methods). Current responses were normalized to remove the dependence on the maximum conductance value, and the normalized current responses (Fig. 2.4A) were subsampled at particular regions of interest

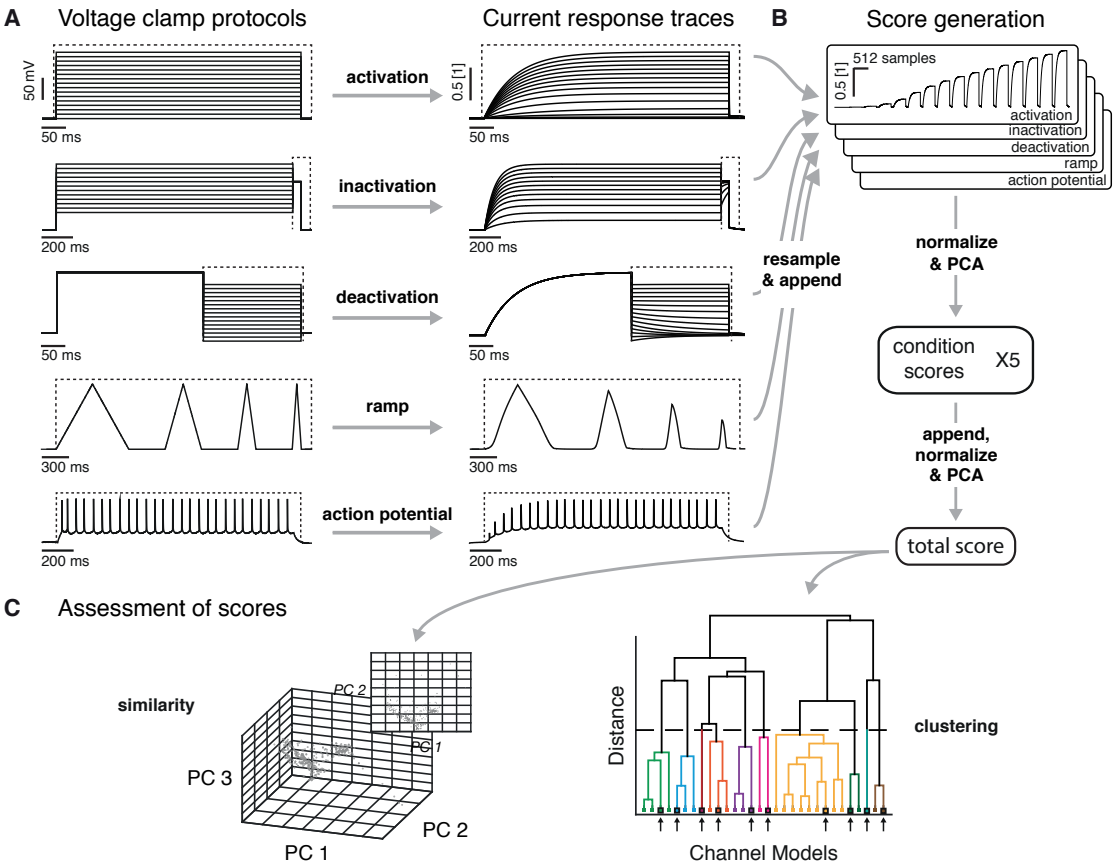


Figure 2.4: Voltage-clamp protocols for the quantitative analysis of ion channel models. **A**: Left: five voltage clamp protocols were used to characterize ion channel responses recorded in single compartment somata simulated in *NEURON* (see Figure 2.10 for full description). Multiple lines indicate a series of increasing voltage steps with the same time sequence. Right: current response traces are shown for an example model. Dashed regions indicate response times used for data analysis. **B**: Current responses were subsampled and appended, then dimensionality-reduced by principal component analysis (PCA) to form a *condition score* vector for each protocol. These score vectors were further normalized and dimensionality-reduced to form a *total score* vector. **C**: The first three principal components of the score vector are shown for Kv ion channel models (top). Scores were clustered using an agglomerative hierarchical clustering technique (bottom). Distinct clusters (noted by colors) form when a cutoff (dashed line) is introduced in the distance between hierarchical groupings, chosen based on several cluster indexes (see methods). *Cluster representative models* (bold squares with arrows) are selected as reference models for each cluster (see Methods).

(Fig. 2.4A, dashed areas), retaining 512 characteristic data points for each protocol as a stereotypic channel response trace.

In a next step we used principal components analysis (PCA) across all traces of a particular ion type (Methods) to obtain low-dimensional scores for each separate

response trace (Fig. 2.4B). We then performed a second PCA step on the normalized scores, arriving at a final score of D dimensions for each model, accounting for at least 99% of the variance across all channel models in each class (see methods for motivation and a more detailed description of these processing steps). The dimensionality D varied between 16 and 29 dimensions for the five classes (Kv: 16, Nav: 21, Cav: 29, KCa: 16, Ih: 16). The Euclidean distance between any two given model scores was termed their "similarity" (Fig. 2.4C, left).

In a final step, we used Ward's clustering method [Ward, 1963] on the model scores to establish an agglomerative hierarchy of ion channel model clusters (Fig. 2.4C, right). We determined an appropriate threshold constrained by intra-cluster variability and inter-cluster distance, through a variety of published cluster indexes (see Methods). For the Kv class, this resulted in 60 clusters with distinct responses (Fig. 2.5, Fig. 2.6). The other ion type classes divided similarly into 38, 43, 44, and 26 clusters for Nav, Cav, KCa, and Ih, respectively (see suppl. figures).

Most ancestor-descendant families fell within only one cluster, indicating consistency between the family relations collected from the papers and model behavior (Fig. 2.5A). However, a common subtype label did not guarantee a common cluster identity, as many models with the same subtype fell into several different clusters due to the diversity of their responses (Fig. 2.5B). Furthermore, ambiguously labeled channel models, e.g. *not specified* and *HH*, could now be assigned a kinetic identity. An interactive similarity view of all models in their score coordinates (Fig. 2.5C) is available on the web interface [ICGenealogy, 2017].

Relations of channel clusters to the subtype labels and family components followed relatively stereotypical patterns (Fig. 2.5B, left & middle). Identical subtype labels segregated into several clusters, yet the majority of any particular subtype label was limited to only a few homogeneously-labeled clusters. For example, for the Kv class, five clusters accounted for 90% of *A-type* channel models, and contained few other subtype labels, suggesting that *A-type* is a generally consistent label for at least five similar, yet distinct kinetic behaviors. Other clusters, containing six or less *A-type* labels, could represent outlier models, mislabeling, or the overlap

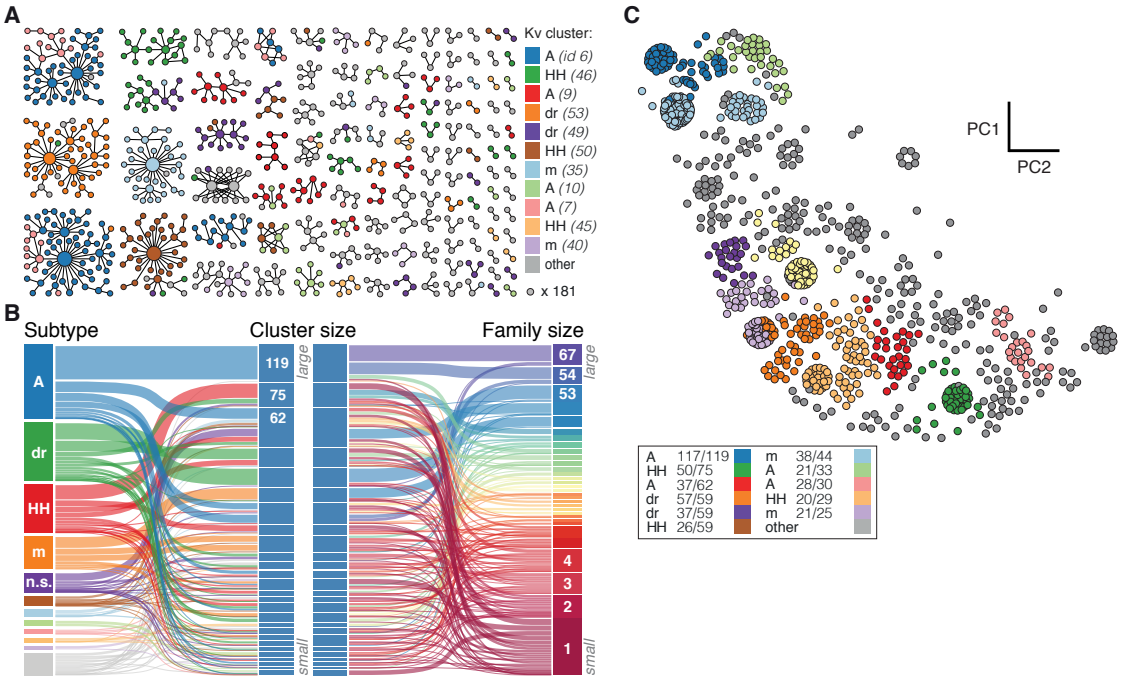


Figure 2.5: Quantitative analysis of Kv ion channel models: functional map and clusters of common behavior. **A** Pedigree graph of the Kv class (cf. Figure 2.2E), colored by membership in the 11 largest clusters in the class (named by most prevalent subtype, bottom). Membership to other clusters is indicated by grey color. Cluster ID is given in parentheses for easy comparison with website. **B**: 'Sankey' diagram for the Kv ion type class, showing the relation between subtype, cluster identification and family identification, each ordered from top to bottom by increasing group size. The 11 most common subtypes are shown in color, with all others grouped together in grey. Small families (size 1 to 6 members) are grouped together. **C**: Plot of Kv models in the first two principal components of score space. Colors indicate membership in one of the 11 largest clusters in the class, with membership to other clusters colored in grey. Clusters are named by their most common subtype, with the proportion of that subtype specified in the legend. Points lying very close to each other have been distributed around the original coordinate for visualization.

---

of *A-type* with another subtype label. The *m-type* subtype label was similarly consistent, with four clusters accounting for 84% of the models. Other subtype labels, such as *HH* and *dr*, belonged to more heterogeneously-labeled clusters, suggesting that these subtypes are not as specific to a particular kinetic behavior. Again, the four other ion type classes displayed similar topologies.

Comparing clusters with families, we also observed similar trends across all five ion type classes (Fig. 2.5B, middle & right). Large clusters were generally associated with large families, and small clusters were associated with small families.

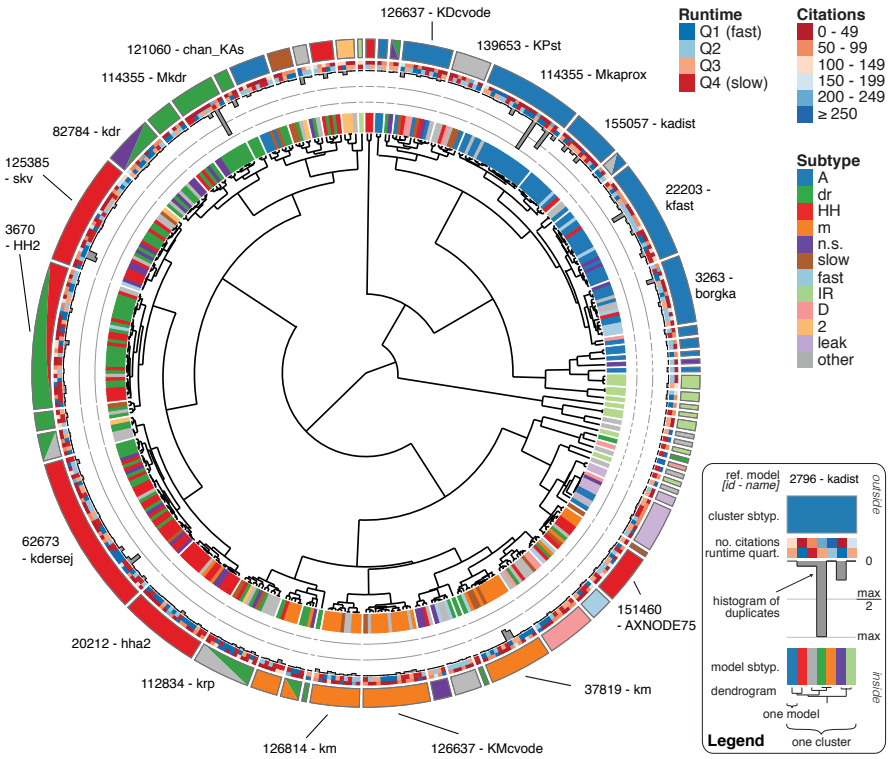


Figure 2.6: Quantitative analysis of Kv ion channel models: 'Circos' diagram of the Kv ion type class. All unique ion channel models are displayed on a ring, organized by cluster identification. From outside to inside, each segment specifies: cluster reference model (only displayed for large clusters), cluster subtype(s) (all subtype labels that contribute at least 30%), number of citations, runtime, number of duplicates, model subtype, as well as a dendrogram of cluster connections (black).

In the case of Kv, the majority (89%) of models in the largest cluster were members of the two largest families. Moreover, the six largest families featured at least 85% of models from a single cluster. Interestingly, most isolated single-model clusters for four of the five ion type classes (not KCa) fell into single-model families, supporting the idea that these models are indeed unique, and do not appear isolated simply due to missing ancestor-descendant links. However, this was not true for all single-model families, many of which were kinetically aligned to larger families and consequently fell into the same clusters.

We plot several metadata items with the quantitative clustering results together in a "Circos" plot in Fig. 2.6. Since many of the clusters were consistently made up of one or a few subtype labels, we proceeded by identifying clusters according to the most common subtype label of their constituents and we denoted the model closest

to the mean score coordinate of each cluster as its reference model. The similarity between any two reference models (and, naturally, all models within the clusters they represent), guides their relative positions along the wheel of the Circos plot and the associated cluster view of the web interface [ICGenealogy, 2017]. For the example of Kv, it can be observed that some of the *A-type*-labeled channel models (Fig. 2.6, dark blue, inner wheel) are in fact quite far from most *A-type*-named clusters (dark blue, outer wheel). Similarly, several *m-type* labeled channel models (orange, inner wheel) can be found distant from the *m-type*-named clusters (orange, outer wheel).

More generally, the relative position of different clusters along the outer wheel of the Circos plots is also representative of the kinetic similarities of the channel models in the cluster. For the Kv class, for example, representative "*121060 - chan KAs*" of an *A-type*-named cluster, has much slower inactivation than the other four *A-type*-named clusters, and is consequently closer to the *delayed rectifier*-named clusters. Representative "*140038 - kdr7*" of a *delayed-rectifier*-named cluster, is found close to the *m-type*-named cluster area, due to the fact that it has slower activation and *m-type*-like ramp and action potential behavior. These kinds of comparisons can be easily investigated with the comparison tool of the interactive web interface [ICGenealogy, 2017], which provides a simple way of exploring the dynamics of each available model without having to implement each one.

To illustrate our results, we have focused mainly on the behavior of potassium channel models, but similar results (with some peculiarities) were found for the other four ion type classes (supplementary figures). The Nav class featured many *not specified*-named clusters, as well as the largest single cluster of all classes – 236 (90 unique) of the total 553 (284) models – which also made up the majority of the three largest ancestor-descendant families (Fig. 2.11). The Cav and KCa classes were the most heterogeneously-labeled (Fig. 2.12, Fig. 2.13). For Cav, the largest family contained substantial numbers of members of four different clusters, although smaller families have more homogeneous cluster membership. For KCa, like Nav, there were some clusters with *not specified* as the most prominent label, and *AHP* was found throughout many areas of the wheel, suggesting nonspecificity.

Unsurprisingly, the Ih class was the most homogeneous, as it already represents a subgroup of cation channels (Fig. 2.14).

2.2.3 Validity and variability of available models

Given that there are several types of behavior for any given subtype, this raises two main questions, which are at the heart of our study: (1) which ion channel models can be considered "valid" models? i.e., do small differences in ion channel dynamics substantially affect the outcome of a modelling study? and (2) where does the variability in models of the same subtype come from? Unfortunately, the first question is out of the scope of this current study – both because we lack the experimental data to directly compare models to biology, and because the relative validity of a model depends very much on the particular modelling question at hand, and cannot be generalized. However, the second question is indeed something that we can explore using the information collected from our database.

We first hypothesized that the variability in models could arise from different systems and neuron types being modelled. However, comparing the clustering results to metadata, such as brain area and neuron type, we find no clear assignment of metadata values for any given cluster (Fig. 2.7). The behavior of groups of ion channel models defined by common subtype, neuron type and brain area (Fig. 2.7A, plotted data points) is often more diverse than that of models defined by a common cluster (dashed line; see Fig. 2.15 for a comparison which includes animal model). Furthermore, nearly every cluster contains some models that have been used in pyramidal cells, and nearly every cluster contains models that were designed for cortical or hippocampal models (Fig. 2.7B). Thus, the diversity in model kinetics is not only due to the diversity of the modeled systems and neuron types, but in many cases there are several viable choices.

We next looked into the experimental sources of the models on the database, which were recorded and added to the "model comments" in the metadata collection. Models come from various different experimental studies, and upon each reuse,

are sometimes modified in light of new experiments. We again focus on A-type-labelled models here as an example. First of all, we see sometimes the same experimental origin being found in more than one cluster, such as clusters 6 and 7, both containing models originally from the same source [Hoffman et al., 1997], but with slight differences in implementation, potentially due to extra experimental information, or model fitting. Second of all, we see multiple experimental origins for similar or identical subtypes in the same system. For example, three clusters contain A-type models used in cerebellar Purkinje cells from three different sources, and interestingly each one is labelled with a different subtype (A-type, D, and 4) but have very similar behavior. Additionally, three models for A-type currents in olfactory bulb mitral cells are found in three different clusters and come from three different references. Consistently, we find that models defined by common families (connected directly or indirectly through ancestor-descendant relationships) can occasionally fall into different clusters (Figure 4A). Thus we see that many models, even of the same ionic current in a similar or identical system, are based on various experimental studies and previous models, ultimately leading to different parameter choices and implementations.

This leads to the question of whether or not the diversity in *models* reflects a diversity in *biology* (rather than error in translation into models). Substantial experimental evidence exists for ion channel heterogeneity, both in expression and regulation [Marder and Goaillard, 2006, Schulz et al., 2008], as well as heterogeneity in ion channel distributions for cells with similar response profiles [Prinz et al., 2004]. Therefore, experimental data based on averaging responses across a population may result in a misleading characterization of a neuron's physiology [Golowasch et al., 2002]. Furthermore, such mechanisms as splice variants, differential subunit combination, and post-translational modification (e.g., phosphorylation) may result in different response profiles for the same cell type [Schulz et al., 2008, Li et al., 2007, Campiglio and Flucher, 2015, Levitan, 1994, Misonou et al., 2004].

One of the main goals of this study was to understand the relationship between different models in order to test their validity. While we do substantially simplify

the number of available models down from over 2300 to roughly 200 groups with similar kinetics, we cannot make strong claims about which of these groups, or which particular model in each group is the best choice. This question would again require ion channel recording data that is not currently available, as well as careful comparison with available models.

2.2.4 Automated comparison of new ion channel models and experimental data

Our analysis framework, accessible through the web interface, enables the automated analysis of new ion channel models as well as experimental data (Figure 2.8A). To illustrate this process, we uploaded and tested a previously uncatalogued Kv model from a hippocampus CA1 pyramidal cell model (*kad.mod* from Hsu et al. [2015]; ModelDB ID no. 184054). We compared its scores and response traces to the presently available 931 Kv models (Figure 2.8B) and determined its relation to previous ion channel implementations. We found that the ion channel model fits well within a cluster of mostly pyramidal A-type-labeled ion channel models used in simulations of rodent hippocampus, thereby verifying the assumed characteristics of the model.

The framework can also be used for the comparison of experimental data and models. To illustrate, we uploaded and tested an experimental dataset of recordings (using the same voltage-clamp protocols as in Fig. 2.4) from a cell type in the olfactory system of *Drosophila melanogaster* (refer to Podlaski et al. [2017] for details on experimental procedures). This cell type proved very suitable to our purposes, as a particular voltage-gated potassium current (Shal/Kv4, resembling an A-type current) appears to dominate the current response in this cell when sodium channels are blocked [Gasque et al., 2005]. By comparing this experimental data in the same 'score' space as the database of models, we found a close match to an existing model, and thus characterizes the behavior of the ion channel as similar to a mammalian Kv4 ion channel (Figure 2.8C; Fineberg et al. [2012]; *Kv4_csi*, ModelDB i.d. 145672).

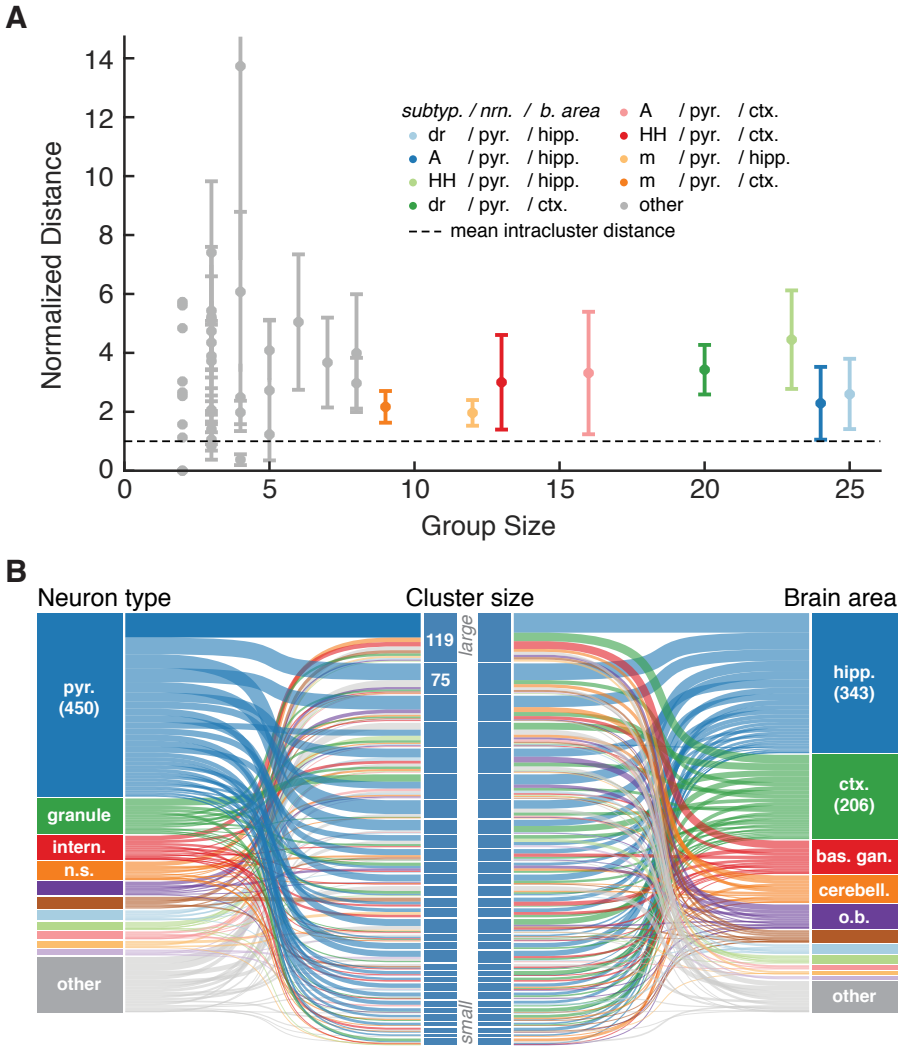


Figure 2.7: Ion channel model groups defined by common subtype, neuron type and brain area show variability in behavior. **A**: Kv models are grouped by common subtype, neuron type, and brain area. The mean pairwise distance in score space between all models within a group is plotted against group size, with errorbars corresponding to the first and third quartiles (middle 50% of distances). The largest eight groups are shown in color. Distances are normalized relative to the mean pairwise distance between models in the same *cluster* (dashed line; averaged across all clusters with at least one pairwise distance greater than zero, 43 of 60 clusters). Most groups have a larger mean distance than the average cluster (above the dashed line). **B**: 'Sankey' relational diagram for metadata, neuron type, cluster identification and brain area, for the Kv ion type class. Bars are stacked histograms, i.e. the height of bars indicate the relative number of models. Left column: partition of channels by 11 most prevalent neuron types. Middle column: partition of channels into assigned clusters. Right column: partition of channels by 11 most prevalent brain areas. Links between columns are colored according to neuron type (left) and brain area (right). Numbers in left/right column are number of channels per label.

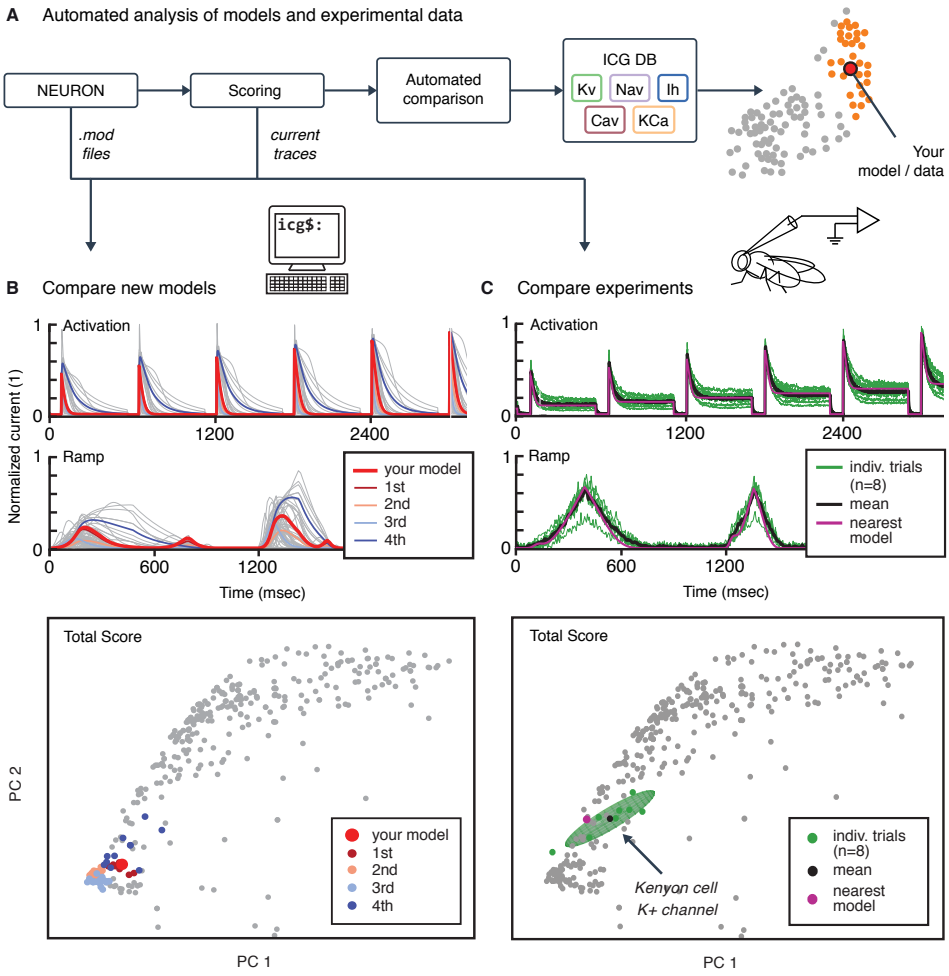


Figure 2.8: Automated analysis of new models and experimental data. **A**: Flowchart of data processing steps involved in automated comparison. Source code for model files are uploaded to the website, and current responses are automatically generated and processed to compute scores, which are compared to all models in the resource (illustrated in **B**). Additionally, raw current traces obtained experimentally (or from models in other languages) can be uploaded and analyzed directly (illustrated in **C**). **B**: Example analysis and comparison of a new ion channel model (*kad.mod* from Hsu et al. [2015]; ModelDB ID no. 184054). Top: Segments of the current response traces (red) for activation (voltage steps 10 - 60mV) and ramp protocols (first half), along with the closest four clusters (other colors: mean currents, grey lines: individual currents). Bottom: first two principal components of score space. **C**: Example analysis of in vivo recordings of a K+ current from Drosophila Kenyon cells (see Podlaski et al. [2017] for details) and comparison to ICG resource. Top: mean (n=8 recordings, black) and individual recordings (green). Bottom: mean (black dot) and individual experimental recordings (green dots) plotted in the first two principal components of score space (ellipsoid illustrates the variance across individual recordings). Comparison is made to the nearest (in score space) ion channel model in the resource (magenta; *Kv4_csi*, ModelDB i.d. 145672). Grey dots in **B** and **C** are scores of Kv channel models in the resource.

2.3 Discussion

In this work, we presented a meta-analysis of a large database of voltage- and calcium-dependent ion channel models available on *ModelDB* [Hines et al., 2004]. By combining metadata extracted from publications with a quantitative characterization of dynamics, we were able to provide detailed information about the identity of each model, as well as a validation of purported model behavior. Along with other recent efforts in neuroinformatics to deal with the increasing amounts of data, publications and models [Akil et al., 2011, Ferguson et al., 2014, Grillner, 2014, Tripathy et al., 2014, Hines et al., 2004, Gleeson et al., 2013], we hope that our framework will facilitate a better understanding of ion channel function and how it leads to complex neuronal behavior. We provide an interactive browser [ICGenealogy, 2017], which acts as a complement to existing resources such as ModelDB. It allows for the exploration and comparison of ion channel models in views similar to many of the figures of this chapter (Figs. 2.9 and 2.16).

Our work represents a first step towards a better understanding of the diversity of biophysical models used in neuroscience research. The voltage-clamp protocols used in this study were designed to efficiently probe the kinetics of all ion channel types considered here [Hodgkin and Huxley, 1952b, Willms et al., 1999, Ranjan et al., 2011], regardless of implementation. While there is evidence that simple step and ramp current pulses are sufficient to probe the underlying kinetics of neurons [Druckmann et al., 2011], we note that the addition of other protocols may be better suited to separate certain models, and can certainly be added. Additionally, our study can be extended beyond the selection of ion channel models considered here. We limited our analysis to voltage-dependent and calcium-dependent ion channel models coded in the *NEURON* language, but, given the appropriate protocols, other types of ion-channels and other simulators can be included (or even simulator-independent formats, e.g., NeuroML; Cannon et al. [2014]).

Our resource provides a substantially reduced collection of candidate ion channel models to consider when building a new neuronal simulation (1132 unique models of 2378, clustered into 211 groups). However, we stress that it is extremely difficult

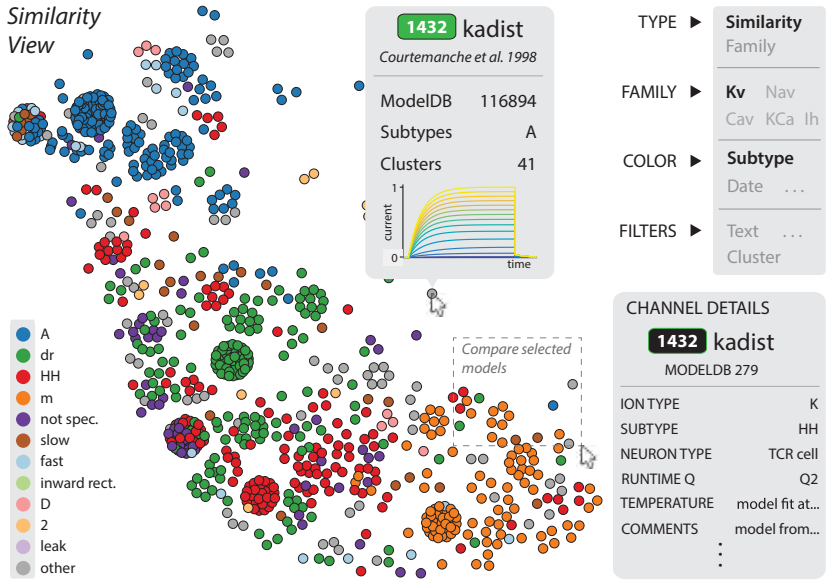


Figure 2.9: The ICGenealogy website allows for the interactive visualization of all data and analysis on the resource [ICGenealogy, 2017] (http://icg.neurotheory.ox.ac.uk). Schematic of similarity view on website. Channel models of the Kv family are displayed in the first two principal components of score space, colored by subtype (legend on left). Hovering over models brings up information tooltip (center), and clicking on a model displays Channel Details (bottom right). Selected models can be compared by click-and-drag. See Fig. 2.16 for other browser views.

---

to assess the validity of different models. As already mentioned, the variability in behavior of ionic currents [Marder and Goaillard, 2006] suggests that there is no clear answer to the question of which ion channel model (and parameters) to use for a given neuronal simulation. Thus, the relevant distinctions between model behaviors is also difficult to determine – models assigned to the same cluster may still possess differences that are crucial for neuronal function. While we were careful to validate our clustering results with several established measures, we cannot escape a certain degree of ambiguity. Since the responses of different channels vary slowly and continuously rather than in discrete steps along the dimensions of the manifold of scores (see e.g. Figure 2.5C), the data may also be amenable to more sophisticated clustering and machine-learning approaches. To this end, the raw response data and scores have been made publicly available [ICGenealogy, 2017].

Perhaps one solution to the question of variability comes from previous studies suggesting that particular neuron types can be seen as occupying particular regions

of parameters space, rather than as individual representative points [Goldman et al., 2001, O'Leary et al., 2014]. This is a promising concept and is currently an area of active research. In this vein, our database would lend itself to systematic analyses of the co-variability of sets of published ion channel models that are able to elicit desired behaviors in conductance-based neuron models (see also Figure 2.1), in line with previous work [Prinz et al., 2003].

One of the motivations behind the voltage-clamp protocols used in this study is the possibility of direct comparison with experiments. We showed one preliminary comparison in this chapter of both the classification of a new model, as well as experimental data (Fig. 2.8). Other resources have also attempted to standardize our understanding of ion channels, with Channelpedia [Ranjan et al., 2011] as one prominent example. While our motivations for this work do overlap with these other projects, the difference here is that we wanted to describe the entire space of published ion channel models, rather than focusing on a select few channel models for a given type. However, while it is beyond the scope of the current study to integrate ion channel information from IUPAR/BPS Guide to Pharmacology [Pawson et al., 2014], Channelpedia or other sources, these are important future steps which would help standardize nomenclature. In conclusion, our resource not only provides a tool for computational neuroscientists in building biophysical neuron models, but also presents a first step towards a unified functional map of ion channel dynamics in model and experiment.

2.4 Methods

2.4.1 Pyramidal cell model (Figure 1)

A detailed multicompartmental model of a layer 2/3 pyramidal cell [Traub et al., 2003] was downloaded from ModelDB (Hines et al. [2004]; accession no. 20756) and simulated in the NEURON language [Hines and Carnevale, 2001, Carnevale and Hines, 2006]. The model consisted of 74 compartments (68 somato-dendritic and 6 axonal) as well as 12 ionic conductances: leak (g_L), transient (inactivating) Na+ (g_{NaF}), persistent (noninactivating) Na+ (g_{NaP}), delayed rectifier K+ (g_{KDR}),

transient inactivating K+ (g_{KA}), slowly activating and inactivating K+ (g_{K2}), muscarinic receptor-suppressed K+ (g_{KM}), fast voltage- and calcium-dependent K+ (g_{KC}), a slow calcium-dependent K+ (g_{KAHP}), low-threshold inactivating Ca2+ (g_{CaT}), high-threshold non-inactivating Ca2+ (g_{CaH}), hyperpolarization-activating cation conductance (g_{Ih}). See Traub et al. [2003] for more details on the ion channels and simulation details.

Following Fig. 2 of Traub et al. [2003], the neuron model was stimulated with a current-step input injected into the apical dendrite as follows: 400 msec at -0.15 nA, followed by 1000 msec at 1.5 nA. The following four measures were computed for each spike train: total number of spikes, total number of calcium spikes, mean inter-spike interval (ISI) and coefficient of variation of ISI.

The gray traces in Figure 2.1A show the default behavior of the neuron model in response to injected input. The simulation was repreated over several model instances with different ion channel models downloaded from ModelDB for g_{KA} (243 models), g_{KDR} (188 models), or both (441 of the total 45684 random combinations).

2.4.2 The *ModelDB* database, NEURON language and ion channel subtype nomenclature

At the time of analysis, the *ModelDB* database [Hines et al., 2004] contained 496 model entries coded in the NEURON language [Carnevale and Hines, 2006, Hines and Carnevale, 2001]. Customizable ion channel models are coded in *NEURON* in *.mod* files (with suffix ".mod") [Hines and Carnevale, 2000]. We downloaded all corresponding *.mod* files for all NEURON entries on *ModelDB*, and renamed them as *ID_name.mod*, where *ID* is the *ModelDB* ID (accession number, unique to each model submission), and *name* is the original name of the *.mod* file. 46 *.mod* files contained more than one current of the same ion type, and were separated into distinct files by appending the suffix "icgXY" to the name, where X was the ion type, and Y was the number of the current, beginning with 1 (e.g., 1234_kv_icgK2.mod for the second Kv current in file 1234_kv.mod). Duplicate file names from the same ModelDB entry were added separately and given unique names by appending a

version number to the name – e.g. "_v2" for the second file. The total number of files collected from the database was 3495.

2.4.3 Collection of metadata and ancestor-descendant relationships

Associated metadata information was collected about each *.mod* file using information in the corresponding journal articles and file code. Database fields include the following:

- **ModelDB ID**. Identification number associated with each entry on *ModelDB*. All channels from the same entry have the same number.
- **PubMed ID**. PubMed citation ID of journal articles associated with this channel's *ModelDB* entry; may contain multiple elements, and can be empty for a select few *ModelDB* entries for which no articles were found on PubMed.
- **ion type**. The ion type, or permeability, of the channel model, as listed in the journal article and *.mod* file. The following ion types were analyzed: potassium (K), sodium (Na), calcium (Ca), nonspecific (NS). If models contained more than one current, all ion types were recorded separately. Other ion types were registered but not included in the analysis.
- **gating mode/mechanism**. The dynamic simulation variable that modulates the kinetics of the model, such as voltage (v), calcium (ca), voltage and calcium (v/ca), sodium (na), chloride (cl), light (o), and g-protein coupled (g). Only v channel models were included in the analysis, with the exception of ca and v/ca models exclusively for the K ion type.
- **subtype**. The listed ion channel type, as detailed in the journal article or the *.mod* file. Subtypes were listed as mentioned without conformation to any naming convention, e.g., [Ashburner et al., 2000, Yu and Catterall, 2004]. If no subtype was given, then the subtype was recorded as *not specified*. A full list of all recorded subtypes is found in Figure 2.1.

- **author**. Listed author(s) of the *.mod* file (programmers). If authors were not specified in the *.mod* file or on *ModelDB*, we recorded the field as *not specified*.
- **animal**. The animal model (and age, if specified) emulated in simulations, either stated explicitly, or inferred from the journal article.
- **brain area/layer**. The emulated brain area and layer of the simulation, as stated explicitly, or inferred from the journal article.
- **neuron type**. The emulated neuron type of the simulation. May be several types, or listed as *general* if no neuron type was specified.
- **neuron region**. The neuron region that the ion channel is found in, divided into dendrites, soma, axon, axon hillock, or specific areas of dendrites or axon.
- **comments**. Comments from the *.mod* file itself and any other information about the channel and model from the journal article, such as previous models or experimental data that were used to constrain the model.
- **runtime** [**ms**]. Elapsed CPU time for running 10 repetitions of a single voltage-clamp protocol (action-potential). In plots and on the web interface, we simplify model runtimes by assignment to one of four quartiles of the distribution of runtimes of all models in each class.
- **temperature**. Details about the model's temperature dependence, and also the temperature at which simulations and/or experiments were performed as described in the journal article.
- **citations**. Estimated number of citations as available through Google Scholar, scraped monthly to update the entries.

.mod files were placed into one of five groups: voltage-dependent potassium (Kv), voltage-dependent sodium (Nav), voltage-dependent calcium (Cav), calcium-dependent potassium (KCa), and hyperpolarization-activated cation (Ih). The calcium-dependent potassium group contained both ca and v/ca channel models

without any distinction. These five groups accounted for 2378 files of the total 3495 downloaded from ModelDB. Mod files that did not fit this description were omitted from the analysis. This included pumps and active dynamics (290), receptor models (370), and models with other gating dependencies or ion types (68), as well as files that do not function as ion channels (366).

We note that the voltage-dependence of ion channels may stem from different underlying mechanisms, including traditional voltage-gated ion channels that contain a voltage-sensitive domain [Catterall, 1995], as well as dependence that occurs indirectly through interaction with other molecules such as polyamines and magnesium [Nichols and Lopatin, 1997], or intracellular signalling cascades [Kase and Imoto, 2012]. The ion channel models considered here are often agnostic to the biophysical mechanism, and different types of models can be used to model an arbitrary voltage-conductance relationship [Destexhe and Huguenard, 2000a]. Therefore, voltage-dependence as discussed in this work refers to the functional relationship between voltage and channel conductance, and does not generally depend on any particular biological mechanism.

Kv Subtype	Count
A	238
delayed rectifier (dr)	187
Hodgkin-Huxley (HH)	155
m	105
not specified	63
slow	32
fast	23
inward rectifier (IR, IR2, IRK)	17
D	15
2	14
leak	11
4	10
1	8
RP	7
low threshold	6
high threshold	6
3	5
3.1	5
anomalous rectifier (AR)	4
x	4
1.2	3
persistent	2
ito	2
other	1
transient	1
2.1	1
KCNQ	1
7	1
HVA	1
udr	1
LVA	1
x1	1
Total Models	**931**

Nav Subtype	Count
not specified	235
Hodgkin-Huxley (HH)	115
fast	82
persistent	78
resurgent	8
leak	6
1	4
1.6	4
slow	4
transient	3
1.7	3
1.9	3
1.2	2
T-type	1
1.3	1
P-type	1
2	1
low threshold	1
1.8	1
Total Models	**553**

KCa Subtype	Count
not specified	99
AHP	58
BK	34
SK	32
C	20
2	8
CT	6
HH	5
fast	3
slow	2
mAHP	2
Total Models	**269**

Cav Subtype	Count
T-type	109
L-type	99
HVA	58
N-type	33
R-type	33
not specified	25
P-type	15
low threshold	12
LVA	10
high threshold	9
HH	6
TNL	5
E	4
Q	4
NPQ	3
leak	2
active flux	1
2.1	1
P/N	1
slow	1
Total Models	**431**

Ih Subtype	Count
h	171
HCN	13
anomalous rectifier (AR)	7
TNC	2
HH	1
Total Models	**194**

Table 2.1: Subtypes of each of the five classes currently available in the resource (Kv, Nav, Cav, KCa, Ih), sorted by the frequency of their occurrence.

The genealogy of ion channel models was defined by an ancestor-descendant relationship. Each channel model was linked to previous models if a relationship was listed in the journal article. We denoted the groups of models connected by these relations as "families". The details of this relationship in the journal articles was often vague. When no information was listed about previous kinetics, both in the journal article and model files themselves, the channels were assumed to have no ancestors and to have been created de novo. However, in many situations obvious similarity in *.mod* file code as verified by a `diff` command was sufficient to link models to previous ones. In these cases, relations were established even when they were not stated in the journal articles. This task was done by hand, and as such is prone to mistakes. We repeated the collection of metadata, including ancestry relations, a second time in order to correct for potential errors.

2.4.4 Voltage-clamp protocols and data processing

Mod files were run individually in a *NEURON* simulation by generating a single soma compartment of length and diameter equal to $20\,\mu m$ and cytoplasmic resistivity of $150\,\Omega cm$. A passive conductance was set to $3.334 \cdot 10^{-5} S/cm^2$. The simulation temperature was set to 37 °C. Reversal potentials were specified separately for each ion type. Some models (172 files) featured explicit calculation of the reversal potential, so internal and external concentration values were added as extra variables to make these equivalent. Parameter values for reversal potential and ion concentrations can be found in Table 2.3.

A particular model was placed in the soma and a series of five voltage clamp experiments were run (Figure 2.10-2), with the current output being recorded. Based on the desired effect of each protocol, only particular sections of the protocols were used in comparing the kinetics (Figure 2.10, dashed lines; also noted in Figure 2.2). The activation protocol featured a single voltage step level, meant to capture the activation kinetics of the model. The inactivation protocol featured a varying voltage step for a long duration, followed by a second fixed voltage step, measuring the inactivation due to the first step. The deactivation protocol featured a single

	Ion Type	V_0	V_1	V_2	ΔV	T_1	T_2	T_3	T_A^*	T_B^*				
Act	Kv	-80	-80	70	10	100	500	100	100	700				
	Nav	-80	-80	70	10	20	50	30	18	100				
	Cav	-80	-80	70	10	100	500	100	98	700				
	KCa	-80	-80	70	10	100	500	100	95	605				
	Ih	-40	-150	0	10	100	2000	100	95	2105				
	Ion Type	V_0	V_1	V_2	V_3	ΔV	T_1	T_2	T_3	T_4	T_A^*	T_B^*		
Inact	Kv	-80	-40	70	30	10	100	1500	50	100	1600	1700		
	Nav	-80	-40	70	30	10	100	1500	50	100	1580	1750		
	Cav	-80	-40	70	30	10	100	1500	50	100	1580	1750		
	KCa	-80	-40	70	30	10	100	1500	50	100	1595	1700		
	Ih	-40	-150	-40	-120	10	100	1000	300	100	1095	1405		
	Ion Type	V_0	V_1	V_2	V_3	ΔV	T_1	T_2	T_3	T_4	T_A^*	T_B^*		
Deact	Kv	-80	70	-100	40	10	100	300	200	100	400	600		
	Nav	-80	70	-100	40	10	20	10	30	20	29	80		
	Cav	-80	70	-100	40	10	100	300	200	100	380	700		
	KCa	-80	70	-100	40	10	100	300	200	100	395	605		
	Ih	-40	-140	-110	0	10	100	1500	500	400	1595	2105		
	Ion Type	V_0	V_1	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_A^*	T_B^*
Ramp	Kv	-80	70	100	800	400	400	400	200	400	100	100	100	2800
	Nav	-80	70	100	800	400	400	400	200	400	100	100	98	2800
	Cav	-80	70	100	800	400	400	400	200	400	100	100	98	2800
	KCa	-80	70	100	800	400	400	400	200	400	100	100	100	2800
	Ih	-80	70	100	800	400	400	400	200	400	100	100	100	2800
	Ion Type	T_1	T_A^*	T_B^*										
AP	Kv	1800	100	1800										
	Nav	1800	98	1800										
	Cav	1800	98	1800										
	KCa	1800	95	1655										
	Ih	1800	95	1655										

Table 2.2: Voltage-clamp protocol parameters for the five ion type classes. Times are stated in units of ms, voltages in units of mV. See Figure 2.10 for graphical description. Items T_A^* and T_B^* represent the starting and ending times, respectively, of the regions used for analysis (dashed areas in Figure 2.4 of the main text, as well as Figure 2.10).

voltage step at a high voltage, followed by a second voltage step of varying amplitude, meant to measure the deactivation kinetics that occur as the voltage is changed from one level to the other. The ramp protocol featured a series of four up and down ramping voltages, at different slopes. Finally, the action potential protocol features voltage deflections as recorded from the soma of a neuron exhibiting a regular spiking pattern. This was recorded from a L2/3 pyramidal neuron of P14 mouse somatosensory cortex (R. Ranjan, unpublished). Each of the five ion type groups featured different voltage values and durations based on differences in time constants, reversal potentials, and voltage ranges at which each ion channel class

	E_{rev} (mV)	$[ion]_{in}$ (mM)	$[ion]_{out}$ (mM)
Kv	-86.7	85.0	3.3152396
Nav	50.0	21.0	136.3753955
Cav	135.0	8.1929e-5	2.0
KCa	-86.7	85.0	3.3152396
Ih	-45.0	-	-

Table 2.3: Parameters for reversal potential and inside and outside concentrations used in simulation protocols for five ion type classes. Ionic concentrations were not used for Ih currents.

is known to be active – no quantitative comparisons were made between classes. Additionally, calcium gated channel models were simulated at seven different calcium concentrations based on known concentrations [Neher and Sakaba, 2008]. Values were expressed in concentration as $10^{-x}mM$, with x taking the following values: $2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0$. Voltage-clamp protocols are available for download from the ICG website [ICGenealogy, 2017].

A substantial number of *.mod* files (952 files) were modified to work with the procedure, in one of the following ways: (1) reversal potential was renamed and made a global variable to be accessed from *.hoc* file, (2) NONSPECIFIC CURRENT was changed to a USEION statement with the correct ion type (3) extra functions and/or data were included through *.inc* files, data tables or extra *.mod* files (4) max conductance was made nonzero (arbitrarily set to 1.0) (5) file was split into multiple files for each current present (6) POINTER variables were removed (7) internal temperature initialization was removed, and temperature dependence was set to use the global variable "celsius". The changed files will be made available upon publication.

All ion channel models were taken from the *ModelDB* repository, as published, with small changes as noted above. This included the assumption that model parameters as chosen by the authors were set to reproduce a given, desired, dynamical behavior which matches experimental data or other constraints. We did not consider changing the internal parameters of given models – this would prevent any feasible comparison of the models, since most given models would be able to generate a large variety of behaviors under changing parameter settings (data not shown).

A small number of *.mod* files were omitted from the analysis due to problems with running the simulation protocols (44 files, Figure 2.4). These included files that did not compile for unknown reasons, and files that produced abnormal oscillations, or extreme values. The count of these files was 16 for Kv, 20 for Nav, 5 for Cav, 2 for KCa, and 1 for Ih.

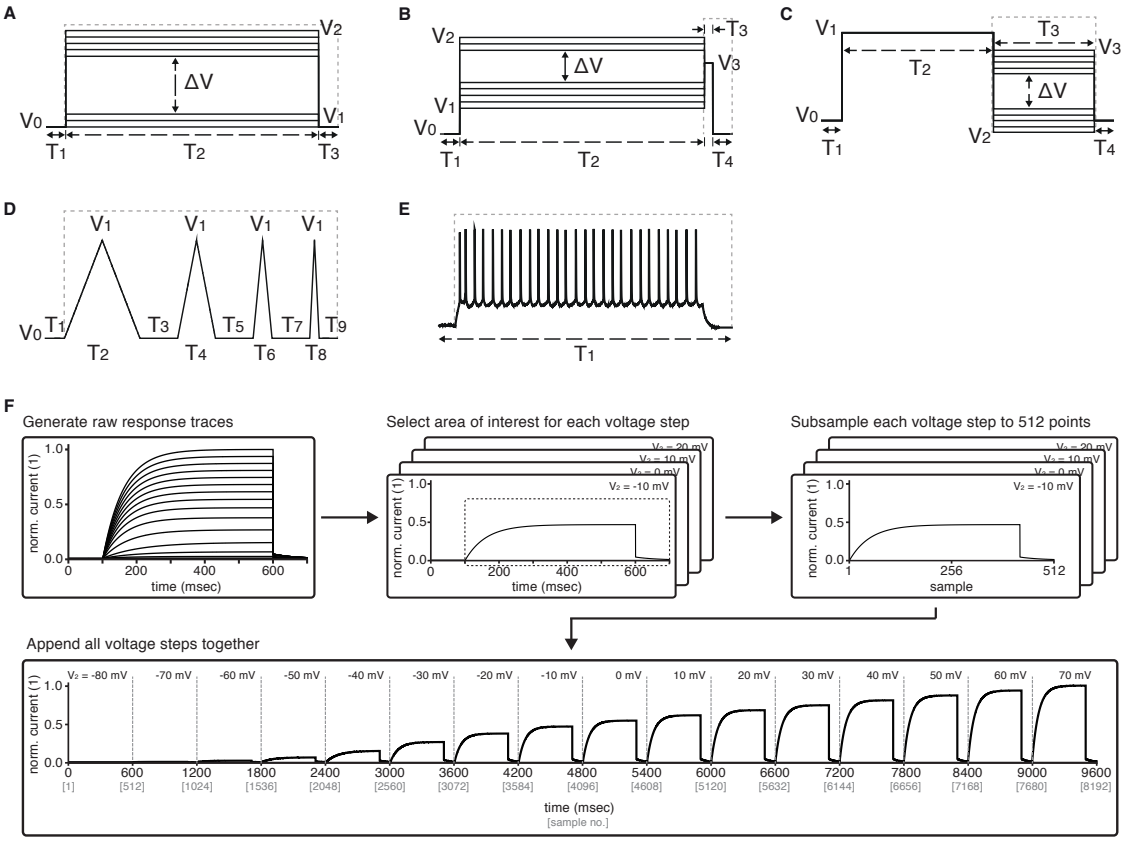


Figure 2.10: Graphical description of the five voltage-clamp protocols used for ion channel model analysis: activation (**A**), inactivation (**B**), deactivation (**C**), ramp (**D**), action potential (**E**). See Figure 2.2 for values of variables used in simulations. Dashed lines represent the approximate regions used for analysis. **F** Preprocessing stage of analysis – example shown for Kv *activation* protocol. Raw current response traces are cut to include only the region inside the dashed lines (as in panels *A*-*E*) for each of the (sixteen, for *activation*) voltage steps of the protocol. Then, each of the resulting traces is separately subsampled to obtain 512 data points. Finally, the subsampled traces are appended to form a final vector of size $512 * 16 = 8192$. Other protocols for Kv and the other four classes are processed similarly, with the following two comments: (1) ramp and action potential protocols do not feature multiple voltage steps, and so only contain 512 datapoints total; (2) KCa current traces are run at seven different calcium concentrations, which are processed separately for each concentration as shown here, and then appended together – e.g., KCa activation will contain $512 * 16 * 7 = 57344$ data points.

Simulations were run using the *NEURON* language [Carnevale and Hines, 2014]. *NEURON* models were injected with the five different voltage clamp protocols described above and integrated at a timestep of $dt = 5e^{-2}ms$. The resulting current traces were processed in the following manner. First, inward currents (Na^{+} and Ca^{2+}; represented as negative deflections from baseline) were flipped by multiplying

	ModelDB ID	mod name	Issue
Kv	127992	HHcn.mod	Produces abnormal oscillations for all protocols
	127992	HHcnf.mod	Produces abnormal oscillations for all protocols
	141272	hh58F1.mod	Produces abnormal current trace responses
	141272	hh58F1ss.mod	Produces abnormal current trace responses
	155601	markov.mod	Produces abnormal oscillations for all protocols
	155602	markov.mod	Produces abnormal oscillations for all protocols
	135838	GK.mod	Tab data insufficient for simulation protocol
	114637	iA.mod	Unknown Error during run(): Segmentation Violation
	140828	kir.mod	Table not specified in hoc func table - no table provided, not used in simulation
Nav	114637	iap.mod	Produces abnormal oscillations in AP protocol
	151443	Naf_No.mod	Produces abnormal oscillations for all protocols
	153280	ch_NavPVBC.mod	Produces abnormal oscillations for all protocols
	84589	naf.mod	Produces abnormal oscillations in AP protocol
	3263	nahh.mod	Produces abnormal oscillations for all protocols
	150446	NafTraub1994sd.mod	Produces abnormal oscillations for all protocols
	127351	nax.mod	Produces abnormal oscillations for all protocols
	147867	nahh.mod	Produces abnormal oscillations for all protocols
	150239	nax.mod	Produces abnormal oscillations for all protocols
	141272	hh58F1.mod	Produces abnormal current trace responses
	141272	hh58F1ss.mod	Produces abnormal current trace responses
	155601	markov.mod	Current traces are mostly zero, does not seem to work
	155602	markov.mod	Current traces are mostly zero, does not seem to work
	127992	HHcn.mod	Current traces are mostly zero, does not seem to work
	127992	HHcnf.mod	Current traces are mostly zero, does not seem to work
	135838	GNa.mod	Tab data insufficient for simulation protocol
Cav	135838	GCa.mod	Tab data insufficient for simulation protocol
KCa	139654	kca.mod	Skipped due to mod file format (point process)
	20007	KctBG99.mod	Produces extreme large current values, unknown reason
IH	144520	isi.mod	Unknown error during run()

Table 2.4: List of omitted files for each of the five ion type classes, with a brief description of the reason for omission.

the entire trace by -1. Next, all current traces were normalized by dividing by the maximum trace value in order to make the result invariant to the max conductance parameter (e.g., $\bar{g}$ in Hodgkin-Huxley sodium current $I_{Na} = \bar{g}m^3h(V_m - E_{Na})$). We reasoned that the maximum current amplitude depends on the number of channel models in a particular area and was thus not related to the kinetic behavior of the channels. The traces were then subsampled at a resolution of 512 data points within the regions of interest stated above. Finally, for protocols containing graded steps (activation, inactivation, deactivation) the subsampled responses across all c graded voltage steps were appended into one representative vector of length $L = 512 * c$. For calcium gated channels, we performed a similar procedure for each of the k calcium concentrations separately, and then appended them into one representative vector of length $L = 512 * k * c$. See Figure 2.10F for a schematic of this process.

2.4.5 Similarity measure and clustering

Several postprocessing steps were applied to the raw current response traces in order to obtain a low-dimensional representation for each ion channel model. Generally, these steps were the following: (1) the subsampled and appended current

responses for each protocol were Z-scored by subtracting the mean and dividing by the standard deviation; (2) principal components analysis (PCA) [Ramsay and Silverman, 2005, Lewicki, 1998] was applied separately to each of the five protocols (each one being a $N \times L$ dimensional data matrix, with N being the number of channel models for a particular ion type) to obtain *condition scores*, with dimensionality chosen to capture 99% of the variability; (3) condition scores were each normalized by their respective standard deviations and appended together into a final score vector; (4) PCA was applied a second time to the final score vector, and dimensionality was again chosen to capture 99% of the variability.

There were several reasons for the particular choice of steps listed above. First of all, response traces were Z-scored in order to focus the analysis on correlations between different temporal dimensions rather than finding particular areas of the protocols with high variance. We found empirically that this produced better results, because otherwise the PCA analysis focused too much of the weight on a small number of dimensions. Secondly, the protocols were first dimensionality reduced separately and normalized (steps 2 and 3) in order to ensure equal weighting of each of them in the final score vector. This accounted for the fact that the protocols may have different inherent variability, as well as clear differences in dimensionality. Finally, due to the initial PCA step done separately for each protocol, it was necessary to do a second PCA step to remove redundancies across protocols.

Since response traces were relatively noise-free, a high PCA dimensionality can be chosen to capture current response dimensions that are rare across the population of models. The precise value of the variance criterion in both PCA steps, although slightly changing the resulting scores, did not affect the clustering results reported above. Similarity between two channel models was then defined as the Euclidean distance of their dimensionality reduced scores.

In summary, the principal components calculated in the first step represented response curves along which the current traces were projected to yield intermediate scores. The second transformation was a linear mixing matrix, which combined these intermediate scores. Final scores had between 16 and 29 dimensions depending

on the family analyzed, which additionally allowed the efficient storage of the characteristics of the thus compressed response properties in our resource. The linear PCA transformations, once calculated, can be applied to additional channel models and their current responses and allow us to efficiently score new channels and easily evaluate them against all other channels in the resource [ICGenealogy, 2017].

For clustering of channel scores, we used Ward's minimal variance linkage [Ward, 1963] for hierarchical clustering, as implemented in the MATLAB Statistics Toolbox (R2015A, The MathWorks Inc., Natick, MA). This method can be used to produce a division of the set of all channel models into an arbitrary number of "similar" clusters, the number of which has to be constrained by internal criteria (we assumed no a-priori existence of classes in this dataset) [Halkidi et al., 2001]. Generally, the algorithm as follows: all models begin as separate clusters, and then they are grouped together by finding the pair of clusters which are closest to one another (in Euclidean space) and defining a new cluster as the combination of the two. This is repeated until a cutoff number of clusters is achieved, which is usually determined empirically. To this end we employed a range of internal clustering evaluation measures, which indicate the emergence of an appropriate number of clusters. Although the evaluation of these measures requires some heuristics, they have been well established and can guide the decision as to which number of clusters to choose. Concretely, these are: the Silhouette criterion [Rousseeuw, 1987], the Dunn index [Dunn, 1973], the Davies-Bouldin index [Davies and Bouldin, 1979], and the Calinski-Harabasz measure [Caliński and Harabasz, 1974], also implemented in the MATLAB Statistics Toolbox (R2015A, The MathWorks Inc., Natick, MA). For the Dunn index, the Silhouette index and the Calinksi-Harabasz measure, high values indicate mostly compact and well-separated clusters. The Davies-Bouldin index also indicates compactness and separation, however for low values. For details and reviews on these clustering indexes see e.g., Milligan and Cooper [1985] and Halkidi et al. [2001].

2.4.6 Generation of figures

'Circos' plots displaying the clustering results together with genealogical links were generated using the Circos visualization tool [Krzywinski et al., 2009], combined with TreeDyn [Chevenet et al., 2006] to create circular dendrograms (Figure 2.5D).

The Circos plot is a visualization technique that enables the comparison of functional similarity to metadata information for each cluster separately. All *unique* models of a given class are arranged in a circle (numbers of duplicate models are shown in the histogram along the circle). *Plot rings*: For each model, the following information is shown (from inside to outside; refer to legend): clustering dendrogram, subtype label, histogram count of duplicate models, model runtime information, number of citations of the accompanying paper, and most common subtype label(s) of each cluster (all subtypes that contribute 30% or more). Location along the circle corresponds to functional (clustering) information, whereas color corresponds to metadata information, such as subtype label. *Location*: position along the circle was established by the circular dendrogram at the center. This dendrogram was created with an agglomerative hierarchical clustering algorithm as defined above, and arranges models in such a way that similar models are in adjacency, and all models in the same cluster appear in one continuous group. The outer ring of the plot denotes the extent of each individual cluster. Groups of models defined by cluster were visually displaced from others by adding a small white space between clusters. *Color*: Three color legends accompanying the graph define the color relationships plotted. The two large rings on the inside and outside are colored by subtype label (of individual models and of clusters, respectively), of which the 11 most common are displayed, with all others in gray. Two smaller rings just inside of the outer ring, denoting number of citations and runtime, are colored on a red-blue scale.

Pedigree plots were generated using Gephi [Bastian et al., 2009], and then manipulated and ordered manually for visualization (Figure 2.2E, Figure 2.5A). Coloring was chosen according to subtype label as well as cluster identity. 'Sankey' diagrams [Sankey, 1898, Schmidt, 2008] were generated in Javascript and D3.js [Bostock et al., 2011] (Figure 2.5B, Figure 2.7B). Subtype coloring was chosen

as for the pedigree plots. Subtype labels, clusters and families were arranged from top to bottom by size.

2.5 Supplementary materials and figures

2.5.1 Abbreviations

Ion channel subtypes:

A: A-type, dr: delayed rectifier, HH: Hodgkin-Huxley, m: m-type, n.s.: not specified, IR: inward rectifier, HVA: high-voltage activating, N: N-type, R: R-type, P: P-type, l.t.: low threshold, LVA: low-voltage activating, AHP: after-hyperpolarization, BK: big conductance, SK: small conductance, HCN: Hyperpolarization-activated cyclic nucleotide-gated, per./pers.: persistent, res.: resurgent. T: T-type, L: L-type, HVA: high-voltage activated, N: N-type, R: R-type, n.s.: not specified, P: P-type, l.t.: low threshold, LVA: low-voltage activated, h.t.: high threshold, n.s.: not specified, AHP: after-hyperpolarization, BK: big conductance, SK: small conductance, HH: Hodgkin-Huxley, mAHP: medium AHP. h: hyperpolarization-activated, HCN: hyperpolarization-activated cyclic nucleotide-gated, AR: anomalous rectifier, TNC: tonic nonspecific current.

Cell types:

pyr.: pyramidal, IN, intern.: interneuron, TCR: thalamocortical relay, RS: regular spiking, HIPP: hilar perforant-path associated, h.m.: hilar mossy, IB: intrinsic bursting, n.s.: not specified, GC: granule cell.

Brain areas:

hipp: hippocampus, bas. gan.: basal ganglia, cerebell.: cerebellum, SCX: somatosensory cortex, DG: dentate gyrus, o.b.: olfactory bulb.

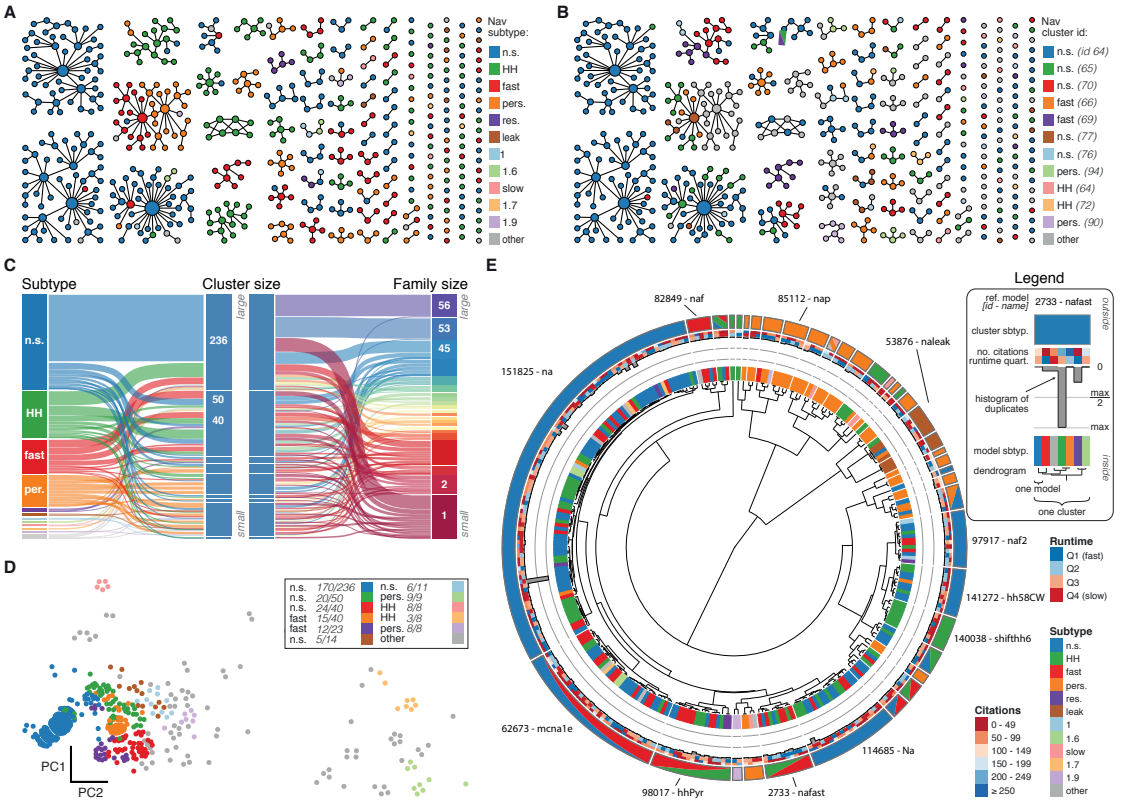


Figure 2.11: Nav class genealogy and clustering. **A**: Families of the Nav class, ordered from left to right by family size. Each dot represents an ion channel model and edges represent family relationships. Colors indicate the 11 most prevalent subtypes in the class, with other subtypes colored in grey. **B** Families of the Nav class, as in panel *A*. Colors indicate membership in one of the 11 largest clusters in the class (named by most prevalent subtype, right hand side), with membership to other clusters colored in grey. Cluster ID is given for easy comparison with website. **C** 'Sankey' diagram for the Nav ion type class, showing the relation between subtype, cluster identification and family identification, each ordered from top to bottom by increasing size. The 11 most common subtypes are shown, with others being grouped together (grey). Small families (size 1 to 5 members) are grouped together. **D**: Plot of Nav models in the first two principal components of score space. Colors indicate membership in one of the 11 largest clusters in the class, with membership to other clusters colored in grey. Legend indicates the most common subtype in each cluster, and the proportion of models with that subtype. Points lying very close to each other have been distributed around the original coordinate for visualization purposes. **E**:'Circos' diagram of the Nav ion type class. All unique ion channel models are displayed on a ring, organized by cluster identification. From outside to inside, each segment specifies: cluster reference model (only displayed for large clusters), cluster subtype, number of citations, runtime, number of duplicates, model subtype, as well as a dendrogram of cluster connections (black) and family relations (grey).

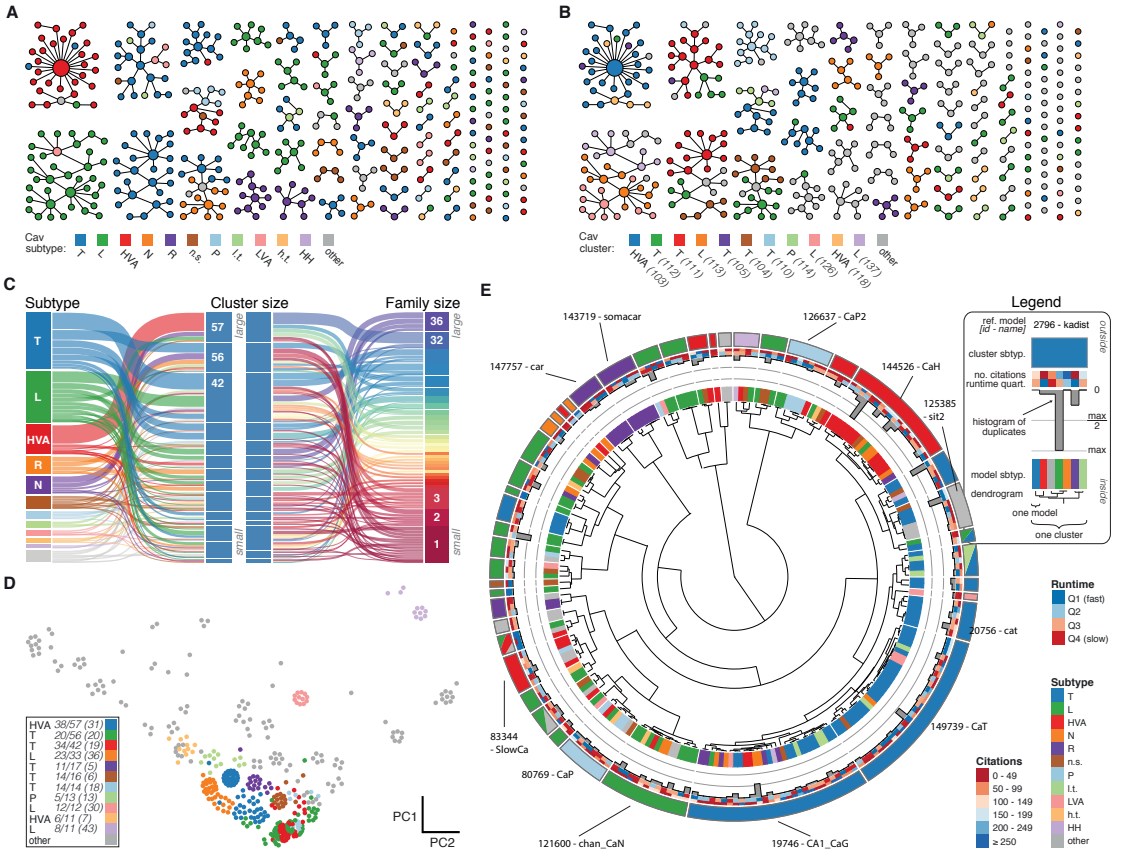


Figure 2.12: Cav class genealogy and clustering. **A**: Families of the Cav class, ordered from left to right by family size. Each dot represents an ion channel model and edges represent family relationships. Colors indicate the 11 most prevalent subtypes in the class, with other subtypes colored in grey. **B** Families of the Cav class, as in panel *A*. Colors indicate membership in one of the 11 largest clusters in the class (named by most prevalent subtype, right hand side), with membership to other clusters colored in grey. Cluster ID is given for easy comparison with website. **C** 'Sankey' diagram for the Cav ion type class, showing the relation between subtype, cluster identification and family identification, each ordered from top to bottom by increasing size. The 11 most common subtypes are shown, with others being grouped together (grey). Small families (size 1 to 5 members) are grouped together. **D**: Plot of Cav models in the first two principal components of score space. Colors indicate membership in one of the 11 largest clusters in the class, with membership to other clusters colored in grey. Legend indicates the most common subtype in each cluster, and the proportion of models with that subtype. Points lying very close to each other have been distributed around the original coordinate for visualization purposes. **E**:'Circos' diagram of the Nav ion type class. All unique ion channel models are displayed on a ring, organized by cluster identification. From outside to inside, each segment specifies: cluster reference model (only displayed for large clusters), cluster subtype, number of citations, runtime, number of duplicates, model subtype, as well as a dendrogram of cluster connections (black) and family relations (grey).

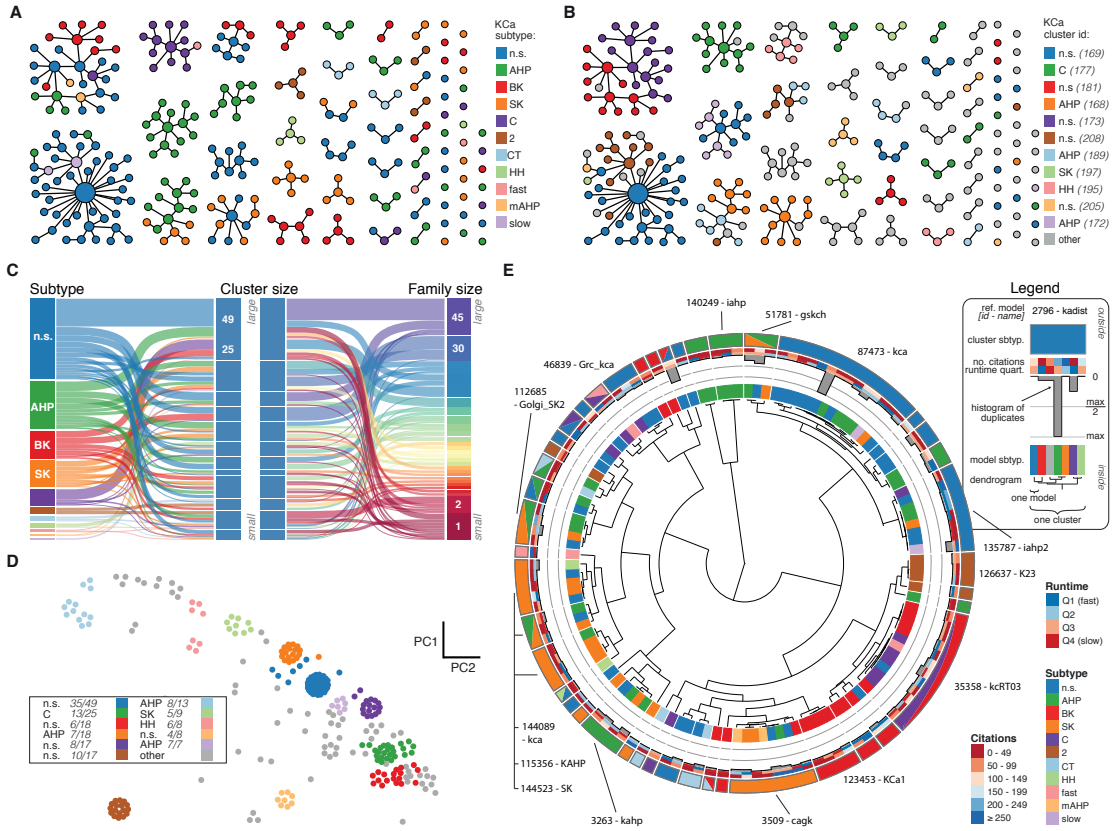


Figure 2.13: KCa class genealogy and clustering. **A**: Families of the KCa class, ordered from left to right by family size. Each dot represents an ion channel model and edges represent family relationships. Colors indicate the 11 subtypes in the class. **B** Families of the KCa class, as in panel *A*. Colors indicate membership in one of the 11 largest clusters in the class (named by most prevalent subtype, right hand side), with membership to other clusters colored in grey. Cluster ID is given for easy comparison with website. **C** 'Sankey' diagram for the KCa ion type class, showing the relation between subtype, cluster identification and family identification, each ordered from top to bottom by increasing size. Small families (size 1 to 2 members) are grouped together. **D**: Plot of KCa models in the first two principal components of score space. Colors indicate membership in one of the 11 largest clusters in the class, with membership to other clusters colored in grey. Legend indicates the most common subtype in each cluster, and the proportion of models with that subtype. Points lying very close to each other have been distributed around the original coordinate for visualization purposes. **E**: 'Circos' diagram of the KCa ion type class. All unique ion channel models are displayed on a ring, organized by cluster identification. From outside to inside, each segment specifies: cluster reference model (only displayed for large clusters), cluster subtype, number of citations, runtime, number of duplicates, model subtype, as well as a dendrogram of cluster connections (black) and family relations (grey).

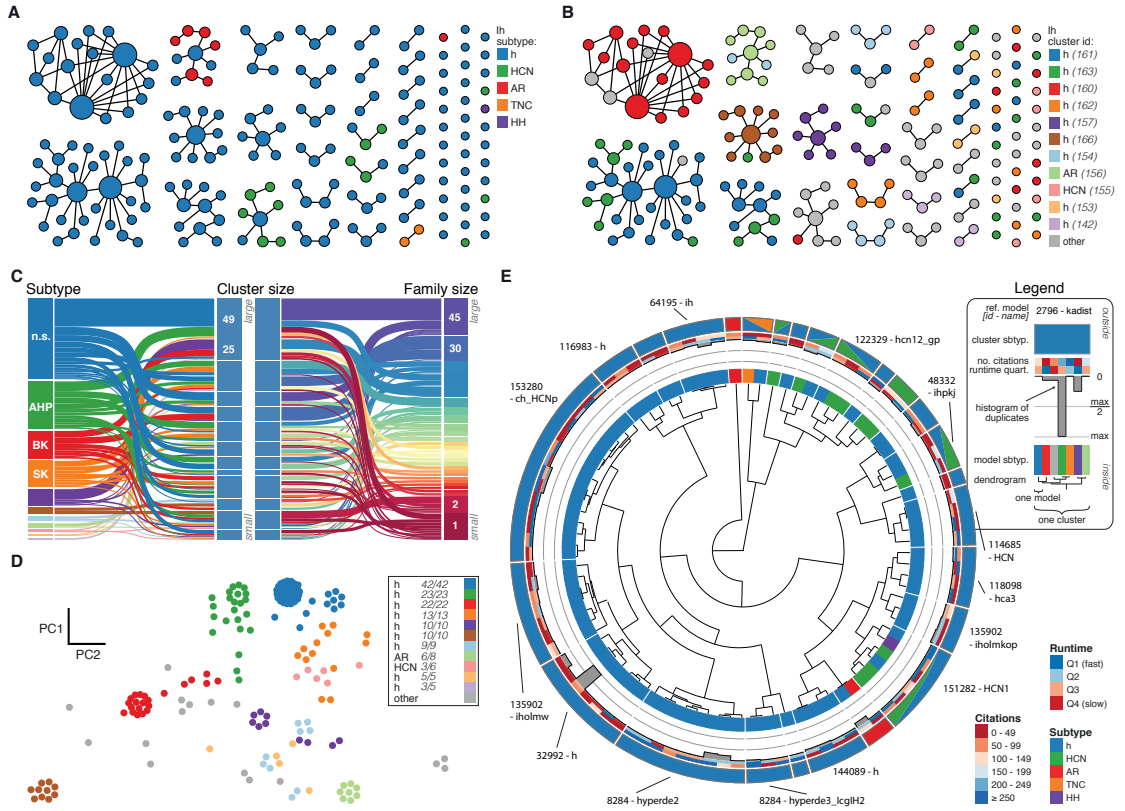


Figure 2.14: IH class genealogy and clustering. **A**: Families of the IH class, ordered from left to right by family size. Each dot represents an ion channel model and edges represent family relationships. Colors indicate the 11 subtypes in the class. **B** Families of the IH class, as in panel *A*. Colors indicate membership in one of the 11 largest clusters in the class (named by most prevalent subtype, right hand side), with membership to other clusters colored in grey. Cluster ID is given for easy comparison with website. **C** 'Sankey' diagram for the KCa ion type class, showing the relation between subtype, cluster identification and family identification, each ordered from top to bottom by increasing size. Small families (size 1 to 2 members) are grouped together. **D**: Plot of IH models in the first two principal components of score space. Colors indicate membership in one of the 11 largest clusters in the class, with membership to other clusters colored in grey. Legend indicates the most common subtype in each cluster, and the proportion of models with that subtype. Points lying very close to each other have been distributed around the original coordinate for visualization purposes. **E**: 'Circos' diagram of the IH ion type class. All unique ion channel models are displayed on a ring, organized by cluster identification. From outside to inside, each segment specifies: cluster reference model (only displayed for large clusters), cluster subtype, number of citations, runtime, number of duplicates, model subtype, as well as a dendrogram of cluster connections (black) and family relations (grey).

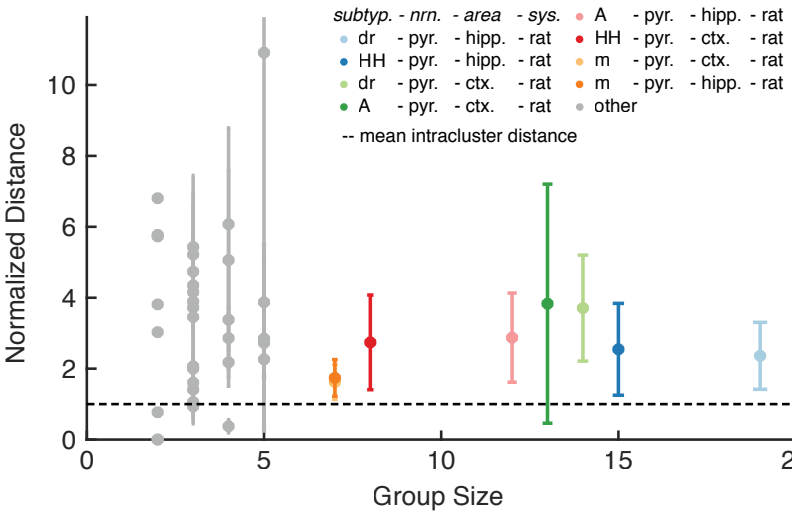


Figure 2.15: Variability of groups defined by common subtype, neuron type, brain area and animal species for Kv ion channel models. The mean pairwise distance in score space between all models within a group is plotted against group size, with errorbars corresponding to the first and third quartiles (middle 50% of distances). The largest eight groups are shown in color. Distances are normalized relative to the mean pairwise distance between models in the same *cluster* (dashed line; averaged across all clusters with at least one pairwise distance greater than zero, 43 of 60 clusters).

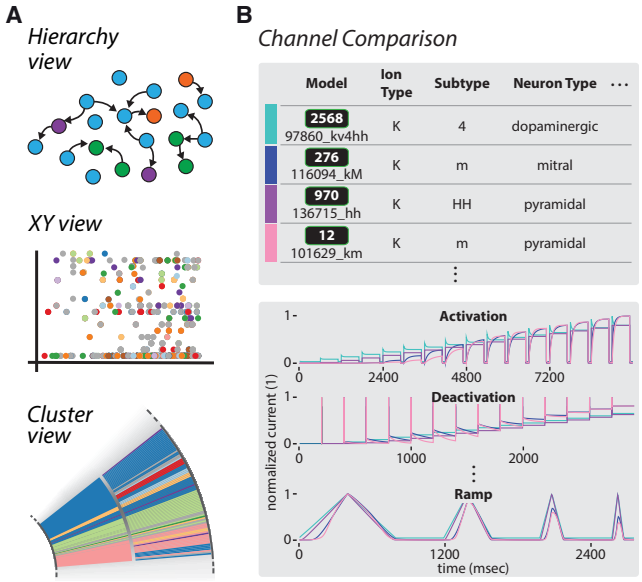


Figure 2.16: Additional browser views on the ICGenealogy website. [ICGenealogy, 2017] (http://icg.neurotheory.ox.ac.uk). **A**: Schematic of similarity view on website. Channel models of the Kv family are displayed in the first two principal components of score space, colored by subtype (legend on left). Hovering over models brings up information tooltip (center), and clicking on a model displays Channel Details (bottom right). Selected models can be compared by click-and-drag (see instruction manual SOM for more details). **B**: Schematic of other three views available on the website. Hierarchy view: models are displayed in a graph with edges that represent family relations. XY view: any two selected metadata are plotted against each other. Cluster view: models are organized in a ring partitioned by clusters. **C**: The channel comparison displays selected channels side-by-side with metadata (top) and current responses (bottom).

3

A standardized model for voltage-gated ion channel dynamics

Abstract

Biophysical neuron modeling has become widespread in neuroscience research, with the combination of diverse ion channel kinetics and morphologies being used to explain various single-neuron properties and responses. However, there is no standard by which ion channel models are made, making it very difficult to relate models to each other and to experimental data. Here, we revive the idea of a standard model for ion channels based on a thermodynamic interpretation of the Hodgkin-Huxley formulation, and apply it to a recently curated database of ion channel models. We show that this standard formulation fits a large percentage of voltage-gated models found in the database (over 2000 models), and produces responses to voltage-clamp protocols and single-neuron simulations with high fidelity. Importantly, the standard formalism enables easy comparison of models based on parameter settings, and can be used to make new observations about the space of ion channel kinetics found in neurons. Overall, this standard formulation should facilitate better understanding and comparisons among ion channel models.

3.1 Introduction

Single neurons form the building blocks of neuronal signal processing, with a diversity of different morphologies and activity profiles across different brain areas and species [Hille et al., 2001]. Processing at the level of single neurons is largely due to the diversity and arrangement of ion channels at different areas of the membrane, leading to interesting and complicated effects [Kandel et al., 2000]. Computational modelling has proven essential in understanding the complex interaction of different ion channels and their effects on input and output [Herz et al., 2006], starting with Hodgkin and Huxley's characterization of action potential generation in the squid giant axon [Hodgkin and Huxley, 1952b].

In the Hodgkin-Huxley (HH) formalism, voltage-dependent ion channel dynamics are described using activation and inactivation gating variables, which have different *steady-state* values and *time constants* , depending upon the membrane potential. The exact shapes of these dependencies may be deduced from data, or can be formulated based upon thermodynamic considerations [Destexhe and Huguenard, 2000c, Sigg, 2014]. Other formalisms have also been introduced to model ion channel dynamics, such as Markov models [Fink and Noble, 2009], which are able to capture more accurately the stochastic nature of ion channel kinetics, as well as additional dependencies that may arise between different states.

In the past few decades, computational models of neurons and neural circuits have utilized these ion channel formalisms to model the dynamics of hundreds of neuron and network models, comprising thousands of ion channel currents with different functional forms describing the steady-state and time constants (Fig. 3.1; models taken from ModelDB, Hines et al. [2004]). However, the lack of standardization in these model descriptions, as well as the sheer abundance of models makes it difficult to validate, reproduce, and compare them.

To address this, we recently created a new database of ion channel models coded in the NEURON language [Hines and Carnevale, 2000, Carnevale and Hines, 2006] which featured a more quantitative approach to comparing models [Podlaski et al., 2017]. Notably, we analyzed the database in a model-free manner – we did not

concern ourselves with the explicit model formulations, but rather analyzed the current responses to voltage-clamp experiments. While this approach does have its benefits, it suffers from two main issues. First of all, while the comparison of different models based on responses provides an interesting view into the diversity of neuronal ion channel kinetics (see previous chapter), there is no way of relating differences in dynamics to particular parameters of the models. Second of all, the database makes no strides towards standardization of model formulations, nor better programming practices when using ion channel models.

Standardization is a hallmark of open science, and for computational modelling, it facilitates the understanding and reuse of models [McDougal et al., 2016]. While several previous studies have discussed general frameworks for ion channel descriptions [Destexhe and Huguenard, 2000b, Sigg, 2014], this has not been adapted by the field – instead model descriptions often use arbitrary functional forms to describe the dynamics. While this enables flexibility in fitting experimental data, it makes it difficult to understand model behavior and variability.

In this work, we revisit the idea of a standardized thermodynamic model for ion channel dynamics, following Destexhe and Huguenard [2000b]. We fit this model to approximately 2500 models on the ICGenealogy (ICG) database [Podlaski et al., 2017], validating adequate model fits through direct comparisons with steady-state and time constant curves as well as voltage-clamp protocols. We further validate this formalism through an example single-neuron model simulation. Due to the common formalism, we are able to easily describe the space of models in terms of parameters.

Overall, this work acts as an important follow-up to our previous paper on the ICG database, and acts to make this work more useful for the field. Importantly, a common formalism also enables much more efficient translation between model languages.

3.2 Results

In the following, we fit a standardized thermodynamic ion channel model formulation based on the Hodgkin-Huxley model to a collection of 2500 ion channel

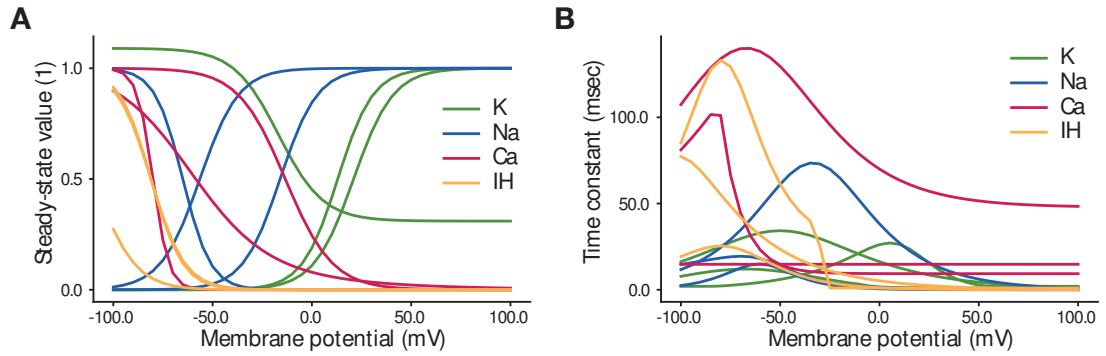


Figure 3.1: Steady-state and time constant curves of gating variables for a random selection of voltage gated K, Na, Ca and Ih channels. **A**: Steady-state curves for activation and inactivation curves show sigmoidal shape. **B**: Time constant curves are generally bell-shaped, but may feature other idiosyncracies such as discontinuities. Lines are colored according to the ion type (see legend).

---

models coded in the NEURON language [Hines and Carnevale, 2001] on the ICG database (originally from ModelDB; [Hines et al., 2004]). All models are voltage-gated and can be categorized into four classes based on ion channel species: potassium (K; 1212 models), sodium (Na; 703 models), calcium (Ca; 524 models), and nonspecific cation (Ih; 241 models). Note that in contrast to the previous chapter, we do not consider calcium-dependent ion channel dynamics here (KCa).

3.2.1 standardized model formulation

In the Hodgkin-Huxley formalism [Hodgkin and Huxley, 1952a, Koch, 2004, Gerstner et al., 2014a], the current $I(t)$ due to a particular type of ion channel is modelled as

$$I(t) = -\bar{g}m(t)^p h(t)^q (V(t) - E_{\text{rev}}), \tag{3.1}$$

where $\bar{g}$ is the maximum conductance of the ion channel, E_{rev} is the reversal potential of the ionic species, $V(t)$ is the membrane potential of the neuron, and $m(t)$ and $h(t)$ are gating variables for activation and inactivation, respectively (with powers p and q). Note that we write this generic form with one activation and one inactivation gate, but some channels do not have any inactivation gate, while others may have

several [Hille et al., 2001]. Each gating variable x has dynamics that follow

$$\tau_x(V)\frac{dx(t)}{dt} = x_\infty(V) - x(t), \tag{3.2}$$

where $x_\infty(V)$ is the steady-state activation (or inactivation) function, and $\tau_x(V)$ is the time constant function. The form of the steady-state activation and inactivation curves is generally sigmoidal in shape (Fig. 3.1A), whereas the time constant curves are more heterogeneous (Fig. 3.1B) – many are bell-shaped, but some are constant, some are bi- or multi-modal, and others feature discontinuities.

Following Destexhe and Huguenard [2000b] (see also Destexhe and Huguenard [2000c]), we formulate a generic functional form for the steady-state and time constant functions (see methods), which we envision should be general enough to describe the vast majority of ion channel dynamics. The steady-state function is described as a modified sigmoid function with four parameters (a,b,c,d)

$$x_\infty(V) = \frac{c}{1 + e^{-aV+b}} + d, \tag{3.3}$$

where a determines the slope, b determines the voltage offset, c scales the function, and d is an additive offset. Note that the sign of a distinguishes between activation and inactivation curves.

The time constant function has the form

$$\tau_x(V) = \frac{A}{e^{-[b_1(V-V_H)+c_1(V-V_H)^2+d_1(V-V_H)^3]} + e^{[b_2(V-V_H)+c_2(V-V_H)^2+d_2(V-V_H)^3]}}, \tag{3.4}$$

with eight parameters (A,V_H,b_1,c_1,d_1,b_2,c_2,d_2). This corresponds to a third-order thermodynamic model. These model formulations were chosen with considerations about model flexibility and number of parameters (see methods). In the following section we briefly discuss model choice and comparison with other candidate models.

3.2.2 Direct model fitting

We extracted data points for steady-state activation, inactivation and time-constant curves by running voltage-clamp protocols on the NEURON .mod files and reading out the gating variables directly. This extraction procedure worked

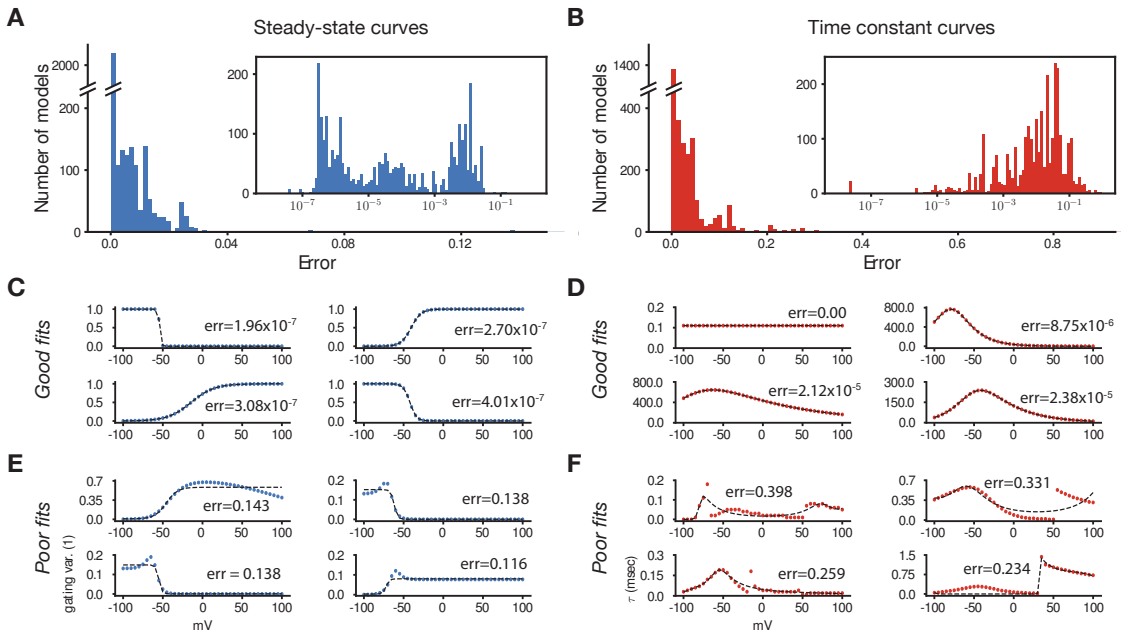


Figure 3.2: Direct standardized model fitting results. **A**: Histogram of errors (see Methods Eqn. (3.9) for definition) for steady-state (**A**) as well as time constant curves (**B**) across all ion types (inset in log-scale). **C,D**: Examples of steady-state (**C**) and time constant (**D**) curves with good fits (low error). **E,F**: Examples of steady-state (**E**) and time constant (**F**) curves with poor fits (high error).

---

for approximately 84% of models, with others being excluded either because they failed to run, or because the variables displayed aberrant behavior.

We then directly fit the parameters of the desired functional forms (the modified sigmoid in Eqn. (3.3) and third-order thermodynamic fomulation in Eqn. (3.4)) using standard nonlinear curve fitting methods (see methods). The error in the fit was measured as the normalized l_2-norm of the difference between the data points and the fit itself (see equation 3.9, methods). As seen in Figure 3.2A, the modified sigmoidal function fit the steady-state activation and inactivation curves very well for the vast majority of models (mean error of 0.0035 across all models). Four good fit and poor fit examples each are shown in 3.2C and D, respectively. As can be seen from the poor fits, some activation and inactivation curves are qualitatively not sigmoidal, which explains the high error. These account for only a handful of models (less than ten), which is small enough that we did not think it reasonable to increase the complexity of the model to accommodate them (but see Discussion).

As shown in Fig. 3.2B, the fits to the time constant curves also yielded low error, but overall performed worse (0.033 mean error). Good fits normally featured smooth curves that were sigmoidal or unimodal in shape (Fig. 3.2D), whereas bad fits often corresponded to functions with bimodal curves or discontinuities (Fig. 3.2E).

We compared the results obtained for the modified sigmoid steady-state function and third-order time constant function with other candidate model formulations (see methods for details; Fig. 3.3), looking at mean fit error and percentage of models in which the fitting procedure did not converge ("No fit %"). The modified sigmoid function (m-sig) for steady-state curves was compared with a standard sigmoid function (sig_0; $c, d = 0$ in Eqn. (3.3)). As can be seen in Fig. 3.3A, the inclusion of the multiplicative factor and additive offset reduced the mean error substantially in all four ion type classes, without causing any fitting issues when compared to a standard sigmoid.

The result of comparing time constant formulations was less straightforward (Fig. 3.3B; note that mean error is plotted on a logarithmic scale). We compared the third-order thermodynamic formulation with first-, second-, and fourth-order formulations (see Methods). The benefit of the second-order formulation over the first-order formulation is clear in all ion type classes (especially for K and Ca), and does not affect model fitting. The third-order model further reduces error for all four ion types, but increases the percentage of models which cannot be fit. This is especially the case for potassium (K) models, though we have not thoroughly explored the reasons behind this peculiarity. Finally, the fourth order model offers slightly reduced gains in mean error (except for IH, where it is again substantial), and further increases the number of models which are not fit. Across the four ion type classes considered (K, Na, Ca, IH), fits were comparable, with similar mean performance (Fig. 3.3A top, B top). However, both for steady-state curves as well as time-constants, the sodium (Na) ion channel models proved to be the most difficult to fit.

The final choice for model formulation depended on several factors relating to model complexity (number of parameters), error reduction, and model coverage (number of models successfully fit). In this work, our main priority was maintaining

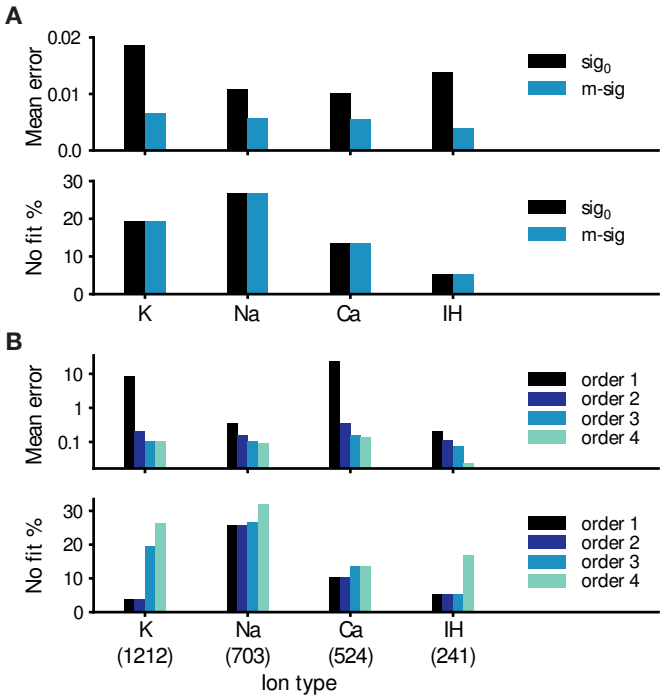


Figure 3.3: Comparison of direct fitting across ion type for different model formulations. **A**: Histogram of mean error (mean across all steady-state and time constant errors for a particular model, *top*) and percent of models in which fitting procedure did not converge (no fit %, *bottom*) for two choices of steady-state model formulation: a standard sigmoid (sig_0, *black*) and a modified sigmoid with offset and multiplicative factor (m-sig, *blue*). **B**: Histogram of mean error (*top*) and percent of models in which fitting procedure did not converge (*bottom*) for thermodynamic model formulation for maximum order of one to four. Total number of models in each ion type class is listed below the names.

---

fidelity with the original model. For the steady-state curves, it was clear that the modified sigmoid offered a practical benefit, with the addition of two extra parameters and no reduction in model coverage. For the time-constant formulation, the third-order model proved to be a good choice in comparison with the fourth-order model because it substantially reduced the error compared with the second-order model, while only decreasing the model coverage modestly (with the exception of K). For simplicity, we also chose to maintain the same model formulation for each ion type class. For the remainder of this chapter, we only consider results for the modified sigmoid and third-order time constant functional forms.

In general, it may appear counterintuitive for model coverage to decrease (increasing no fit %) as the model complexity increases. We note that this simply

reflects the inability of the fitting algorithm to find a good solution as the number of parameters increases, and not the ability of the model to capture the diversity of curves seen in the data (which would surely improve with model complexity). Better initialization choices for the third-order and fourth-order models may remedy this, which we plan to explore in future work.

3.2.3 Comparison using voltage-clamp protocols

The direct model fitting procedure in the previous section worked well for the majority of models – however, it is difficult to assess the true validity of the standardized model fits without considering the behavior of the models in physiological conditions. As a first step in this direction, we decided to compare the standardized models with the original models in a set of five voltage-clamp protocols previously used on the database of models [Podlaski et al., 2017].

We created a standardized ion channel .mod file for each original file using the fits obtained to the steady-state and time constant curves (Fig. 3.4A, left). Gating powers (p and q in Eqn. (3.1)) were extracted directly from the original model files. We then simulated each model separately for each of the five voltage-clamp protocols (Fig. 3.4A, center). The protocols, comprising activation, inactivation, and deactivation voltage steps, as well as ramping and action potential-mimicking traces, are meant to probe the full dynamic range of most ion channels, and differ for each ion type class (for more details and discussion, see previous chapter). The current responses for each standardized model were compared with those of the original model to again compute a normalized error measure based on the normalized l_2-norm of the difference between response traces, which we refer to as the *ICGenealogy* or *ICG protocol error* (3.4A, right; see methods).

As can be seen in Fig. 3.4B, the responses of the standardized models to the voltage-clamp protocols were similar to the original models, with most errors close to zero. However, the errors in general were much higher than those for the direct model fitting, suggesting that the voltage-clamp protocols provide a more sensitive measure of fidelity with the original models.

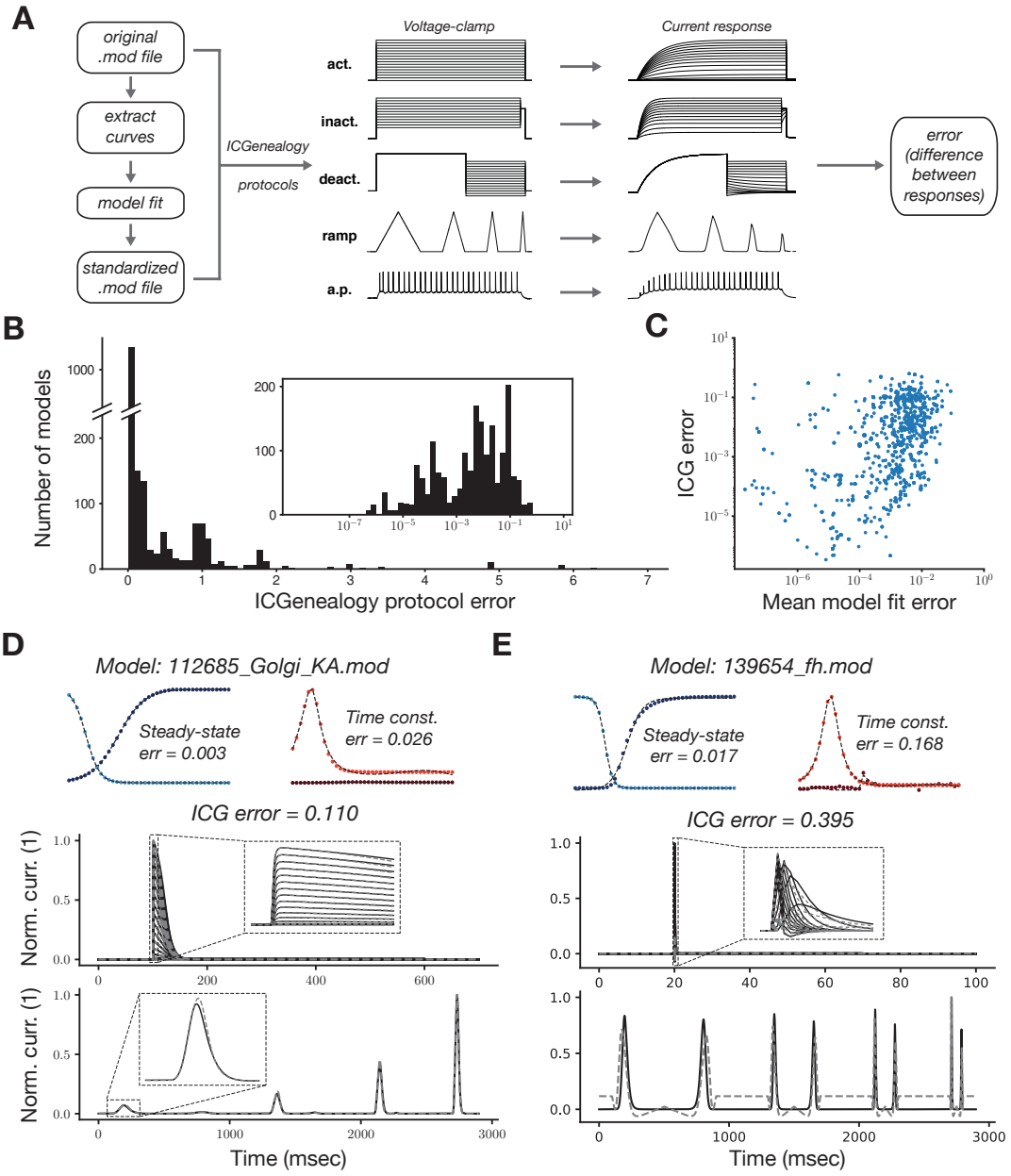


Figure 3.4: Validation of standardized model formulation through ICGenealogy voltage-clamp protocols. **A**: Schematic of ICG protocols applied to standardized model validation. Original mod files are used to extract rates and fit a standardized mod file (left). Both mod files are run through five voltage-clamp protocols (center), and the difference between the responses is computed (right). **B**: Histogram of errors (see methods for definition) across models of all ion types. **C**: Comparison of mean direct model fitting error versus ICGenealogy protocol error for each individual model. $R^2 \approx 0.012$. **D,E**: Comparison procedure for two example models. Extracted rates and direct model fits are shown for activation and inactivation gates (top), as well as comparison of activation and ramp ICG protocols (bottom).

One natural question that arises is whether or not the direct model fitting error was predictive of the ICGenealogy protocol error. When plotting the two together (Fig. 3.4C), we see that though a slight trend exists, the two errors do not correlate very much ($R^2 \approx 0.012$). Note that here we use the mean model fit error across all steady-state and time constant curves of a particular model. The lack of correlation suggests that there may be particular areas of the steady-state and time constant curves which are more important for model performance than others. For example, voltage-step protocols (activation, inactivation and deactivation) focus in on a *single* voltage level, and thus a very small area of the steady-state and time constant functions, which means that any discrepancies that arise there will be amplified.

We show two examples to illustrate the protocol comparison, for an A-type potassium channel (Fig. 3.4D) and a fast sodium channel (Fig. 3.4E). The standardized A-type potassium model shows a very good fit to the model curves (Fig. 3.4D, top), and while the ICG protocol responses are very close to those of the original model, some discrepancies still arise. As for the fast sodium current, the standardized model fit is worse (especially for the time constant functions), which is also seen in the ICG protocol responses (Fig. 3.4E). Whether or not these differences, small or large, would make a difference in capturing the desired behavior of a full neuron model is not as clear. We briefly delve into this question in the next section.

We make two more observations relating back to Fig. 3.4C. First of all, models with the largest mean model fit error tend to have higher ICG error – this is reassuring, because it suggests that models which are poorly fit will of course not match the current responses in voltage-clamp. However, low mean model fit error can correspond to a wide array of ICG errors, sometimes extremely high. This indicated to us that there may be some issues when comparing the response traces. For example, upon inspecting some particular files, we found artifacts having to do with the simulator and integration method used. See Discussion for more details about this.

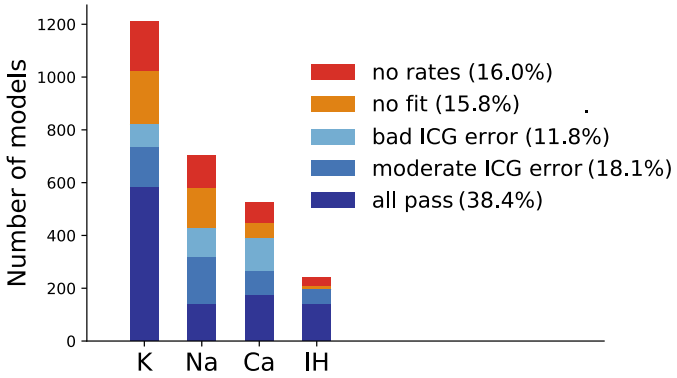


Figure 3.5: Current status of standardized model fitting procedure. Histograms for each ion type are shown with color indicating models for which no rates were extracted (red), no direct fit was obtained (orange), ICG protocols did not run (beige), poor ICG error (> 1.0; light blue), moderate ICG error (> 0.1; medium blue), and low ICG error (all pass; err< 0.1; dark blue.

3.2.4 Current status

As of the final draft of this thesis, we are currently making good progress towards the formulation of a standardized model for each entry in the ICGenealogy database. Shown in Fig. 3.5 is the breakdown for all models of each ion type. There are still over 30% of models for which we either cannot extract steady-state and time constant values, or for which the fitting procedure doesn't converge (Fig. 3.5 red and orange, respectively). This may require manual inspection of some or all of these models.

Out of the models which we are able to fit and simulate through the voltage-clamp protocols, a substantial portion of these models display poor error for the ICGenealogy protocols. While this may be due to a poor direct model fit for some models, we suspect that other issues may be causing problems for the other models, as eluded to above.

3.2.5 Realistic neuron and network simulations

While the voltage-clamp protocols provide a much more physiologically-realistic measure of the fidelity of the standardized model formulations, we see two issues with using them as the final word on model fidelity. First of all, as already mentioned, these protocols may stress different parts of the model differently, thus

amplifying potentially small errors to a large degree, whereas in real neuronal simulations the membrane potential does not normally sit at a single value for hundred of milliseconds. Second of all, each of these models was created for a particular functional purpose in mind – the ICGenealogy protocols are agnostic to this purpose, and thus the error does not reflect the validity of the standardized model as applied to the original intention of the simulation.

The true test of the standardized model formulation is to put it into a full neuron simulation (preferably an identical or similar one for which the model was intended) to see if it produces the same results. Ideally, to test the fidelity of each model on ICGenealogy, we would download any available code from ModelDB and simulate the full neuron or network models for as many models as we are able to fit a standardized model.

We have not accomplished this yet, and so instead we show this procedure for one example model, taken from Traub et al. [2003] (the identical model to the one used in Fig. 2.1 from the previous chapter). This model contains eleven total active ionic conductances, eight of which we were successfully able to fit with a standardized model (Fig. 3.6A; fits shown in Fig. 3.7). We compare the default model with two modified models – one in which eight of the voltage-dependent currents are replaced with standardized models (Fig. 3.6 blue), and another in which only six standardized models are used (Fig. 3.6 red; omitting KM and CaL).

To make the comparison, we specifically look at two figures from the original paper (Figures 2 and 4 in Traub et al. [2003]). In the first figure (Fig. 3.6B), a direct current step stimulation is given to the dendrite while recording from an apical dendritic compartment (top) and the soma (bottom). As can be seen in the default model, depending upon the input level, the cell may fire a single burst of action potentials (coinciding with a dendritic spike), or several. When comparing with the standardized model behavior, we see that the model in which all eight currents were replaced is not able to reproduce the multiple bursts upon high current injection, whereas the model with only six currents is able to qualitatively capture the behavior. Though the fits to all models are relatively good (Fig. 3.7), it would appear as though

very small ion channel differences (perhaps specifically with KM and CAL models) makes a difference as to what input level switches the spiking behavior of the cell.

We then simulated another experiment in which four current step levels were injected into the soma, resulting in spiking behavior that switches from bursting to regular spiking (Fig. 3.6C; compare with Traub et al. [2003], Fig. 4). Compared with the default model, we see that both standardized models perform qualitatively similarly, with some differences, especially at the highest current level (top). Thus, once again, it appears as though small differences in kinetics may have substantial effects on the results when put into realistic single neuron simulations.

Going forward, this example raises the question of what it means for a standardized model to maintain the fidelity of a particular ion channel model. An important point to note is that many of these single-neuron models have many parameters, which makes them very flexible (presumably reflecting the flexibility of real neurons). It is possible that differences in responses due small changes in ion channel dynamics can be counteracted by changing other parameters, such as the maximum conductance value for each ion channel in each compartment. Thus, small details in ion channel formulations may not be crucial for single-neuron functionality, even though they clearly lead to differences in simulation outcomes. However, the inherent complexity of the models also makes them extremely sensitive, which makes comparison difficult. If a single-neuron model displays the same *qualitative* behavior after replacing some ion channels, does that mean that this standardized model can be seen as equivalent? Or, instead, is it necessary that all aspects of the simulation are *quantitatively* identical? These are important questions to be answered in future work.

3.2.6 Future work: clustering in parameter space and comparison with metadata

One benefit of obtaining standardized model fits which does not depend upon the absolute fidelity of the models (provided the fits are moderately close) is the exploration of the resulting parameter space of models. One validation step that

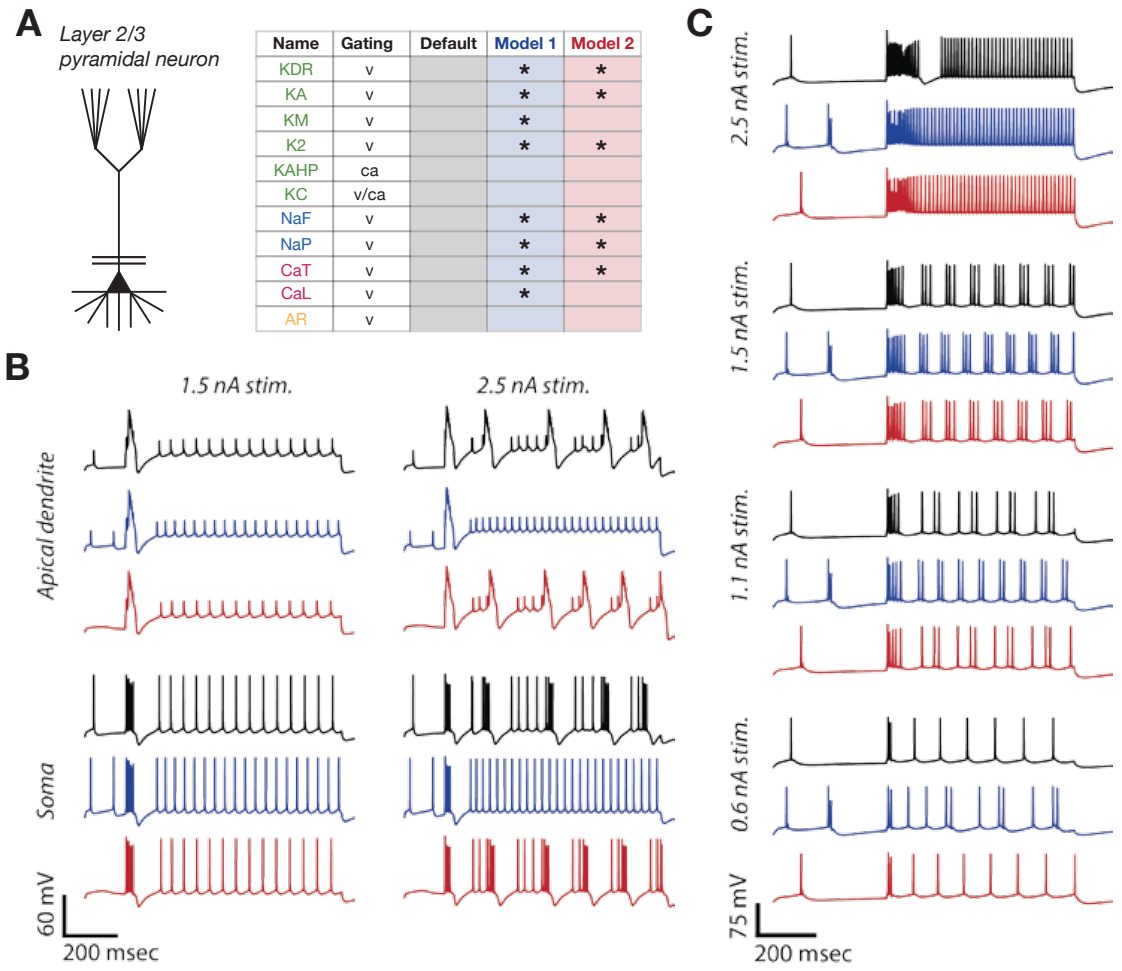


Figure 3.6: Test of standardized models in full single-neuron simulation. **A**: A morphologically-extended layer 2/3 pyramidal neuron model (left) with eleven active conductances (right) was downloaded from ModelDB (Traub et al. [2003]; ModelDB accession no. 20756) and simulated. Right: the model was simulated using default ion channel models (black), with eight models replaced with best-fit standardized models (Mod. 1, blue), and with six models replaced with best-fit standardized models (Mod. 2, red). Asterisks indicate models replaced with standardized models. **B**: Reproduction of Figure 2 from Traub et al. [2003] for default model and models in which some currents were switched with the standardized formulation (blue, red). Voltage traces show recordings in apical dendrite (top) and soma (bottom) during current stimulation of $1.5nA$ (left) and $2.5nA$ (right). **C**: Reproduction of Figure 4 from Traub et al. [2003] for default model and models in which some currents were switched with the standardized model (blue, red). Voltage traces show somatic recordings for current stimulation of $2.5nA$ (top) to $0.6nA$ (bottom).

should be taken is the comparison of parameter space with the functional space obtained directly from the ICGenealogy voltage-clamp protocols (e.g., Figs. 2.5, 2.6). We should see a similar "distance" between models in either space, as well as a similar clustering into functional groups.

Furthermore, a comparison of parameters with metadata items from the ICGenealogy database is also possible. The ICGenealogy database contains information about neuron type, brain area, animal model, and many other potentially informative criteria, which may span particular regions of the parameter space. Though we did not find much correlation between functional clustering and metadata items such as neuron type (see Fig. 2.7 from the previous chapter), perhaps a different picture would emerge when comparing directly with parameters.

We have not explored these avenues yet, but they remain exciting opportunities for future research. Before they can be fully exploited, however, a better overall model coverage would need to be achieved.

3.3 Discussion

In this work, we have presented a framework for fitting a standardized ion channel model to a recently curated database of approximately 2500 voltage-dependent ion channel models (ICGenealogy, Podlaski et al. [2017]) comprising four ion type classes (K, Na, Ca, IH). We showed that the standardized model formulation is flexible enough to fit the steady-state and time constant curves of the majority of models with low error, and maintains reasonable fidelity when simulating them in voltage-clamp protocols. Simulation of an example single-neuron model with six to eight of the ionic currents replaced with standardized models results in qualitatively similar behavior, showing that this formulation is a promising standardization for the field.

The model equations chosen were loosely based on a thermodynamic formulation [Destexhe and Huguenard, 2000b]. The steady-state activation and inactivation curves were modelled with a modified sigmoid function with a multiplicative constant and additive offset, and the time constant curves were modelled with a third-order thermodynamic formulation. Though these proved to be reasonably successful for

the results shown here, it may be the case that other formulations, especially for the time constant functions, work better. One interesting point of note is that we separate the steady-state and time constant curves as having separate parameters, whereas this is not really true in reality (see discussion in methods).

The resulting errors from the ICG voltage-clamp protocols suggest that direct model fitting errors are not necessarily predictive of errors seen in more realistic simulations. Though we discuss the possibility that these protocols may weight different areas of the fit differently, it is also possible that some discrepancies between the models may be due to numerical errors and simulation choices that were made both for the ICGenealogy database and for the standardized model fitting procedure. For example, the ICGenealogy protocols require that current responses are normalized in order to be compared, so that the maximum conductance does not matter. However, many of the response traces (especially for voltage-step protocols which feature discontinuities) feature artifacts which may then corrupt this normalization. We are currently devising strategies to overcome these issues, which should substantially reduce the error measurements from the voltage-clamp protocols.

We showed an example of a simulation of a full single-neuron model, in which the standardized ion channel models led to qualitatively similar behavior to the original model. However, the differences between these results raises the question about how to assess the fidelity of the particular standardized model. This may not have an easy answer, and may depend upon the particular question at hand. One aspect that is clear is that models which depend upon strong fine-tuning of ionic conductances may be affected by arbitrarily small differences in formulation.

One of the main motivations behind this work, as well as that of the previous chapter, is the lack of standardization in the field of biophysical models. Indeed, there are countless ways of formulating ion channel kinetics, and many of the resulting behaviors may also depend upon simulator idiosyncracies. One promising research area in neuroinformatics which is attempting to remedy this is the formulation of simulator-independent model description languages, such as NeuroML [Cannon

et al., 2014]. One very exciting prospect about this work is that a standardized parameter description of ion channel models will allow easy conversion into other languages. It is also our intention to perform a functional translation of all models into NeuroML, which should be straightforward with this common formalism.

We see the results from this chapter as an extention to the work put into the original creation of the ICGenealogy database. The standardized model information will ideally be added to the website, providing an additional dimension of exploration for browsing and comparing models. This also provides a new avenue for model fitting. New fitting algorithms based on likelihood-free inference are able to fit parameters of biophysical models in a standardized way [Lueckmann et al., 2017]. We will begin to explore this in the context of fitting simple ion channel models as well. If successful, then we should be able to produce an ion channel model fit to arbitrary experimental data, which can also be made available on the ICGenealogy website.

3.4 Methods

3.4.1 Details of standardized model formulation

Steady-state curves

Model equations were loosely based on a thermodynamic formulation of ion channel dynamics [Tsien and Noble, 1969, Hill and Chen, 1972, Stevens, 1978, Destexhe and Huguenard, 2000b]. Steady-state activation and inactivation curves are for the most part sigmoidal, with the majority of curves having a maximum and minimum value of 1 and 0, respectively (Fig. 3.1). A small, but appreciable number of models (approximately 4%) featured maximum and/or minimum values which took on arbitrary values between 1 and 0, suggesting that these channels do not fully activate or inactivate. These considerations motivated the following standard equation for steady-state activation and inactivation equations (repeated from Eq. 3.3)

$$x_\infty(V) = \frac{c}{1 + e^{-aV+b}} + d, \tag{3.5}$$

for each gating variable x, with four free parameters (a,b,c,d). This resembles a first-order (linear) thermodynamic formulation [Destexhe and Huguenard, 2000b], with the addition of a multiplicative constant (c), and additive offset (d).

Due to the small number of models requiring the multiplicative and additive parameters in the steady-state formulation, we also considered a standard sigmoid function

$$x_\infty(V) = \frac{1}{1 + e^{-aV+b}}, \tag{3.6}$$

with two parameters (a,b). See Fig. 3.3A for a comparison of the two.

Time constant curves

Time constant curves for activation and inactivation are much more varied in their shapes (Fig. 3.1). Here, we used a standard nonlinear thermodynamic formulation [Destexhe and Huguenard, 2000b] with up to fourth-order terms

$$\tau_x(V) = \frac{A}{e^{-[b_1(V-V_H)+c_1(V-V_H)^2+d_1(V-V_H)^3+e_1(V-V_H)^4]} + e^{[b_2(V-V_H)+c_2(V-V_H)^2+d_2(V-V_H)^3+e_2(V-V_H)^4]}}, \tag{3.7}$$

with ten parameters (A,V_H,b_1,c_1,d_1,e_1,b_2,c_2,d_2,e_2). We compared the effect of adding different number of terms by considering formulations with different numbers of terms. The first-order model sets parameters c_1,d_1,e_1,c_2,d_2, and e_2 to zero, and so has only four parameters. The second-order model has six parameters, with d_1,e_1,d_2, and e_2 set to zero. The third-order model, which is used for the majority of results presented in this chapter, sets parameters e_1 and e_2 to zero, and so has a total of eight parameters. See Fig. 3.3B for a comparison of fits.

Temperature dependence

Ion channels display a stereotypical dependence upon temperature, which manifests itself in the Hodgkin-Huxley model as a multiplicative factor on the rates [Hille et al., 2001]. This has no effect on the steady-state curves, but the time constant function should be multiplied by a factor $1/t_{\text{adj}}$ defined as

$$t_{adj} = Q_{10}^{\Delta T/10}, \tag{3.8}$$

where Q_{10} is defined as the ratio of rates when the temperature has changed by ten degrees, and $\Delta T = (T - T_{\text{orig}})$, i.e., the difference between the original temperature of the kinetics and the temperature used in simulation. In addition to the time constant, temperature is known to also affect the maximum conductance, as well as the reversal potential of each channel [Hille et al., 2001].

In the ion channel models described in ModelDB and ICGenealogy, at least 42% of the models include temperature dependence in addition to the voltage-dependence. This temperature dependence may be based on the experimental recordings of the channels, however this is not guaranteed. We numerically estimated the Q_{10} value of models by running them at 5°C and 37°C. For models with no temperature dependence, we keep $Q_{10} = 1$.

For the results presented in this chapter, we do not explore temperature dependence of the models, and set the simulation temperature to 37°C for all simulations. In future work, especially if making standardized models available to the community, this temperature dependence will be added.

3.4.2 Direct model fitting

Steady-state and time constant curves were numerically extracted from each model file by running activation voltage-step protocols in NEURON and directly reading out the resulting time courses of the state variables. A set of 41 points were extracted for each curve, corresponding to voltage values between −100mV and 100mV, with 5mV intervals (see e.g., Fig. 3.2C,D). This curve extraction procedure was successful for 84% of models.

Direct model fitting was performed in Python using the scipy.curve_fit function. This function performs iterative nonlinear least squares optimization via the Levenberg-Marquardt algorithm [Press et al., 2007], which uses a combination of the Gauss-Newton algorithm and gradient descent. Model fitting was performed iteratively through different model classes, in the following way: for the steady-state curves, first a normal sigmoid was fit (with parameters a,b from (3.6)), and then the modified sigmoid was fit using the previous fit as initial conditions for the

next one (with $c, d = 0$). Note also that initial conditions for a, b were generated uniformly at random between 0 and 1 for the sigmoid fit, but the initial sign of the slope parameter was chosen to correspond with activation (positive) or inactivation (negative). This was found to substantially improve the fitting results.

Similarly, all time constant curves were first fit with the first-order formulation, and then each subsequent order terms were added using the previous fit as the initial conditions, with all other parameters set to zero. This was again empirically found to improve the fits slightly, as random initial conditions would occasionally not converge. Initial parameters for the first-order fit were chosen in the following way: V_H was set to the voltage value at which the curve was maximum, A was set to the mean of the curve, and all other parameters were initialized to zero. This essentially sets the initial fit to a straight line, which proved crucial for fitting some time constant curves which were indeed constant over voltage. When a random initialization was used, the fitting procedure was not able to fit the horizontal lines.

Error between the data points (x_i) and the model fit ($\hat{x}_i$) was calculated as the l_2-norm of the difference between the data points divided by the $l2$-norm of the data itself

$$\text{Error} = \frac{||\mathbf{x} - \hat{\mathbf{x}}||_2}{||\mathbf{x}||_2} = \frac{\sqrt{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}}{\sqrt{\sum_{i=1}^{N} x_i^2}} \tag{3.9}$$

3.4.3 Voltage-clamp protocols

Generation of standardized NEURON .mod files based on direct model fit

Standardized model .mod files were created by feeding the parameter values into a template .mod file which was responsible for simulating the dynamics of all gating variables of a particular ion channel according to the model formulations defined above. Due to numerical issues pertaining to the NEURON simulator, the maximum conductance ($\bar{g}$) was set to $1.0 S/cm^2$ for K models, and $10^{-3} S/cm^2$ for Na, Ca and IH models. These were set based on empirical explorations, and we have not thoroughly explored the reasons behind the numerical differences

observed. It appears to be linked to the range of time constant values that are present in the different ion types.

ICGenealogy voltage-clamp protocols

We refer the reader to the previous chapter for details on the voltage-clamp protocols (e.g., see Fig. 2.10 and Table 2.2), and only provide a brief description here.

Each .mod file was placed into a single soma compartment with a small passive conductance. A series of five voltage-clamp protocols were run: activation, inactivation, deactivation, ramp, and action potential. The activation, inactivation and deactivation protocols featured a set of voltage steps of varying amplitude and duration. The ramp protocol featured four triangular "ramping" voltage traces in which the voltage linearly increases from a baseline value to a maximum, and then decreases back to baseline again. Finally, the action potential protocol featured a voltage trace from an experimentally-recored current-clamp experiment (see previous chapter for details).

The current response traces for each model were recorded and normalized by dividing by the maximum value of each protocol, which accounts for different $\bar{g}$ values in the original models. The traces were then appended together to get one large data vector, which was used to compare the original model with the standardized model. The error between responses was measured analogously to (3.9), and is referred to as the *ICGenealogy* or *ICG protocol error*.

3.4.4 Single-neuron simulation

The pyramidal neuron model from Traub et al. [2003] was downloaded from ModelDB (Hines et al. [2004]; accession number 20756). It contains eleven ionic conductances: leak, transient (inactivating) Na+ (NaF), persistent (noninactivating) Na+ (NaP), delayed rectifier K+ (KDR), transient inactivating K+ (KA), slowly activating and inactivating K+ (K2), muscarinic receptor-suppressed K+ (KM), fast voltage- and calcium-dependent K+ (KC), a slow calcium-dependent K+ (KAHP), low-threshold inactivating Ca2+ (CaT), high-threshold non-inactivating Ca2+

(CaL), hyperpolarization-activating cation conductance (AR). See the description in the previous chapter or the original paper [Traub et al., 2003] for more details.

In addition to the original model, we created two modified versions of the model which featured eight standardized ion channel models (Mod. 1; KDR, KA, KM, K2, NaF, NaP, CaL, and CaT) and six standardized ion channel models (Mod. 2; KDR, KA, K2, NaF, NaP, and CaT), respectively.

We reproduced two of the figures from the original study [Traub et al., 2003] using the available code on ModelDB. In Fig. 3.6B, a current step input was applied to the apical dendrite with amplitude $1.5nA$ or $2.5nA$ for $1000msec$, and the voltage was recorded from the dendrite and soma (Fig. 2 in Traub et al. [2003]). In a second simulation, a somatic current step was applied for $1000msec$ at four amplitudes: $2.5nA$, $1.5nA$, $1.1nA$ and $0.6nA$, and the voltage was recorded from the soma (Fig. 4 in Traub et al. [2003]).

3.5 Supplementary figures

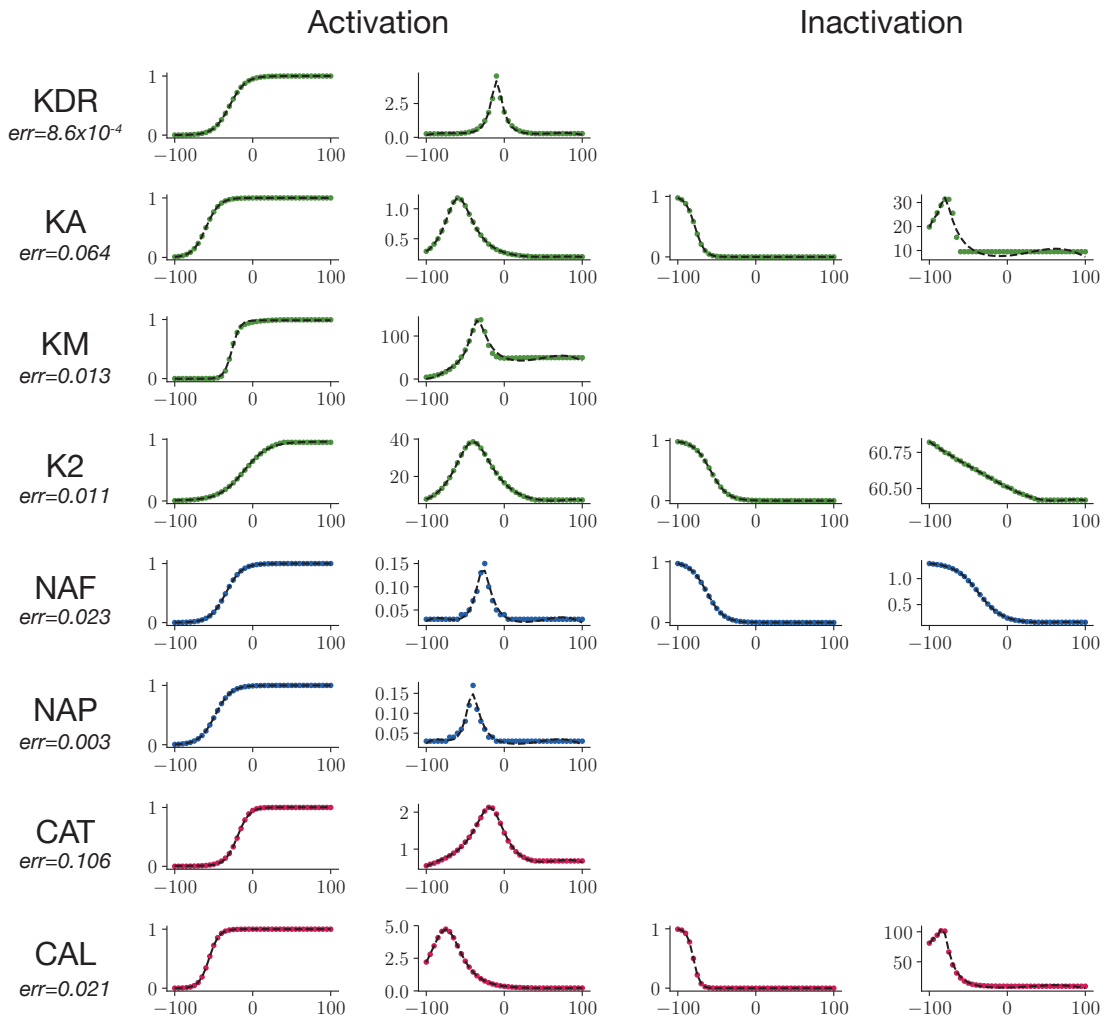


Figure 3.7: Direct standardized model fits for full single-neuron simulation. Extracted rates (points) and direct model fits (black dotted lines) for eight voltage-dependent ion channel models from Traub et al. [2003]. Color as in Fig. 1, indicating ion type.

Part II

Models of context-dependent processing in neuronal networks

4

Introduction: neural circuits and memory

Up to now, this thesis has focused on the micro-scale components of neural processing, i.e., single neurons and the dynamics of their constituent ion channels. At this low level, we can describe and capture the vast diversity of activity profiles seen in single neurons. The characterization of individual neuronal properties and responses has been the subject of much study over the past century (reviewed in Shepherd [2015]). More recently, in part due to advancements in recording techniques for studying the brain, the focus in neuroscience has shifted away from the single neuron towards the population, and indeed some have suggested the neural network as the new fundamental unit of nervous system function [Yuste, 2015]. Neurons are embedded in large networks of interacting units, and it is likely that many of the computations performed in the brain will only be understood at the network level..

In the remainder of this thesis, I will take a higher-level perspective, and study how networks of neurons may be organized to perform particular functions, with a special focus on memory and context-dependent processing. This introduction is divided into two parts. The first part is a general introduction to neuronal circuits and computational models that try to explain their dynamics. The second part introduces memory as one particular function of brain circuits. There, I provide a background on the psychology and neuroscience of memory, and then

on to computational models of memory and context-dependence, which is the focus of the next chapter.

4.1 Neuronal network dynamics, organization and function

4.1.1 Neuronal circuits of the brain

The mammalian brain contains on the order of hundreds of millions (for rodents; Herculano-Houzel et al. [2006]) to hundreds of billions (for humans and elephants; Herculano-Houzel [2012]) of neurons, organized into different areas with distinct connectivity patterns and cell types. Many of these neurons connect to thousands of other cells (up to hundreds of thousands for Purkinje neurons of the cerebellum; Napper and Harvey [1988]), and receive a constant barrage of input from presynaptic partners [Destexhe et al., 2003], both in local circuits and from other areas [Harris and Shepherd, 2015]. While the proposition of understanding this remarkable complexity sounds like a daunting task, many breakthroughs have been made over the past century or two. Here, I briefly describe some findings and possible principles of circuit function, starting with a high-level view of brain areas, then moving down to the the single neuron, and finishing with the neural circuit and population dynamics.

Coarse organization of the brain

The idea that the brain can be divided up into different areas that perform different functions can be traced back to early theories of *phrenology* by Franz Joseph Gall in the early 19th century, who suggested that different areas of the brain would grow (along with the skull) as skills are developed (reviewed in Zola-Morgan [1995]). Though this was disproven, later work on cortical language centres by Broca and Wernicke [Geschwind, 1970], among others, provided substantial evidence that many functions are *anatomically localized* in the brain [Kandel et al., 2000]. It should be noted, however, that even at this time, Wernicke and others believed that different components of a particular behavior could be *distributed* across different areas.

In the intervening years, we have now discovered much more about different brain areas and their purported functions. The cortex is traditionally divided into four areas – the frontal, parietal, temporal and occipital lobes – broadly responsible for planning and movement, somatosensation, audition, and vision, respectively [Kandel et al., 2000]. Functional roles have been attributed to other areas as well, such as the hippocampus for memory (Scoville and Milner [1957]; described more in the next section), and the cerebellum for fine control of movements [Ito, 1972]. In sensory areas, and most notably vision, a general hierarchical picture of processing has emerged, in which sensory information is transformed through a series of different areas, and then is directed into higher level association areas [Felleman and Van Essen, 1991, Riesenhuber and Poggio, 1999, DiCarlo et al., 2012].

However, despite the prevailing feedforward view of sensory processing, most cortical areas contain a substantial amount of feedback connections from other areas, the function of which is not well understood [Harris, 2013]. Furthermore, recent work is making it increasingly clear that many associative and motor areas communicate substantially with even primary sensory areas (e.g., Marques et al. [2018], Saleem et al. [2018]), suggesting that there is still much to understand about what is being processed and represented in these areas.

Single neurons and receptive fields

Much of the understanding about the localization of function described in the previous section has come from single-neuron recordings. Indeed, arguably, much of what we know about the brain in general comes from the characterization of single-neuron responses [Yuste, 2015].

The view from the single-neuron level has had a major place in neuroscience research since the seminal research by Hubel and Wiesel [Hubel and Wiesel, 1959] to characterize the receptive fields of neurons in the primary visual cortex (V1). Hubel and Wiesel discovered “simple cells”, which respond to edges at particular orientations in localized spatial areas, and “complex cells”, which have similar characteristics, but display a degree of spatial invariance. Later work showed similar

characterizations of auditory spectrotemporal coding [Aertsen and Johannesma, 1981] and spatial coding cells in the hippocampus [O'Keefe and Dostrovsky, 1971], among others. These simple response descriptions fit well within the feedforward processing view described above. One current promising area of research is the comparison of response profiles from experimental data with large-scale feedforward artificial neural networks trained with supervised learning [Yamins and DiCarlo, 2016]. This has enabled a better understanding of higher-level sensory areas, which have evaded simple descriptions in previous research.

While substantial progress has been made on this front, evidence exists that this is not the entire picture of sensory processing [Fairhall, 2014]. Phenomena such as response normalization and surround suppression suggest that neuronal responses within a hierarchical stage cannot be fully described through receptive fields alone, as each neuron's activity is not entirely independent of the others [Shapley et al., 2007, Ozeki et al., 2009]. Furthermore, for primary visual cortex, most neurons *do not* display the stereotypical simple and complex receptive fields as reported by Hubel and Wiesel, and all cells are extremely variable on a trial-to-trial basis [Carandini et al., 2005]. Finally, as mentioned above, top-down and long-range connections coming from other brain areas may influence neuronal activity as well.

Local recurrent networks: excitation, inhibition and canonical circuits

If single neuron responses are insufficient to explain the computations arising at different levels of the cortical hiearchy, then it is likely that recurrent excitatory and inhibitory connections play important roles. In the cortex, the majority of input to each cell comes from the local circuit [Douglas and Martin, 2007].

Most circuits are composed of excitatory neurons, which have a positive effect on postsynaptic partners (through glutamate in mammals), and inhibitory interneurons, which have a negative effect (normally through GABA) [Kandel et al., 2000]. *Dale's law* states that each neuron may only have one of these effects, thus creating two separate populations of neurons in each circuit, where inhibitory neurons normally accounts for approximately 10-30% of the population (depending upon the species

and brain area) [Kandel et al., 2000, Beaulieu, 1993, Gabbott and Bacon, 1996, Gabbott et al., 1997]. The interaction of excitation and inhibition has been the focus of much recent study. Due to the positive feedback nature of recurrent excitation, it was theorized that inhibition should act to "balance" excitation [Van Vreeswijk and Sompolinsky, 1996], which was later shown in experiments [Moore and Nelson, 1998, Shu et al., 2003, Wehr and Zador, 2003]. This may have interesting implications for circuit dynamics, and may explain surround suppression as well as neuronal variability [Ozeki et al., 2009, Hennequin et al., 2018]. Furthermore, balanced inhibition with a slight time lag allows short *windows of opportunity* for signals to propagate, possibly leading to precise spike-timing [Haider and McCormick, 2009] or transient amplification [Murphy and Miller, 2009].

More generally, many experimental studies and theories have put forward the idea of canonical circuits – stereotypical connectivity patterns which are found throughout the cortex (or other areas) to provide a diversity of functions [Douglas and Martin, 2007, Plebe, 2018]. This perhaps began with the "discovery" of the cortical column [Mountcastle, 1957], a common architecture for the arrangement of cells throughout the layers of cortex. The definitive canonical circuit of the cortex remains elusive, but several principles have been proposed which are found in many areas, such as normalization [Carandini and Heeger, 2012] and amplification [Douglas et al., 1995].

Of note more recently is the distinction between different inhibitory cell types, which appear to play specific functional roles in cortical circuits [Isaacson and Scanziani, 2011]. The majority of cortical inhibitory neurons can be divided into one of three types based on genetic expression: parvalbumin-positive (PV+), somatostatin-positive (SST+), and vasoactive intestinal peptide-positive (VIP+; sometimes classified as 5HT3aR-positive) [Harris and Shepherd, 2015, Tremblay et al., 2016]. These neuron classes typically show different synaptic target localizations onto excitatory neurons, with, e.g., PV+ neurons primarily targeting perisomatic areas and SST+ neurons targeting areas of the apical dendrites [Harris and Shepherd, 2015]. A major area of current research is dissecting specific inhibitory motifs, determining which neurons mediate them, and how pervasive they are in

different areas of cortex. For example, one well-studied motif is thalamocortical feedforward inhibition, which has been studied in somatosensory, visual and auditory cortex, and appears to be primarily mediated by PV+ neurons in layer 4 [Feldmeyer et al., 2018]. Recent studies in somatosensory cortex have shown that SST+ interneurons may also contribute to feedforward inhibition from the thalamus at different timescales [Feldmeyer et al., 2018], but it is unclear if this holds for other sensory areas. Another prominent example relates to VIP+ neurons, which have been found to predominantly target other inhibitory neurons, and have thus been implicated in disinhibition [Letzkus et al., 2015]. Locomotion has been shown to increase excitatory firing rates in the primary visual cortex, and is likely due to increased VIP+ inhibition onto SST+ neurons, thereby reducing inhibition onto excitatory cells [Wood et al., 2017]. Interestingly, this does not seem to be the case in auditory cortex, where excitatory firing rates tend to decrease with locomotion [Wood et al., 2017].

Neuronal population dynamics

The push to understand network function has popularized a dynamical systems view of neural circuits [Sussillo, 2014, Evan et al., 2018]. In this light, recurrent connectivity and single-neuron properties describe how neural activity evolves over time as a "dynamical engine" [Shenoy et al., 2013], with inhibition acting as a control mechanism [Hennequin et al., 2017].

Dimensionality reduction techniques are now commonly used on neuronal population data to understand the variables being represented in the system and how they evolve over time [Kobak et al., 2016, Gao et al., 2017]. This view differs from the previous approaches in that it does not focus on the single-neuron receptive field. Instead, each neuron participates in many aspects of the signal [Rigotti et al., 2013]. This has led to some network-level descriptions of well-characterized brain areas, in which neurons previously thought to be untuned actually contribute substantially to dynamics and representation [Zylberberg, 2017, Meshulam et al., 2017, Stefanini et al., 2018].

To summarize, neuronal processing in the brain is achieved through the dynamics of intricately-connected networks of different neuron types, which receive input both locally, as well as from other brain areas. To understand this complexity, computational models are becoming increasingly useful, both to understand the dynamics, as well as to propose relevant computations. In the next section, I briefly describe the models employed to do this.

4.1.2 Abstract descriptions of neuronal dynamics

The first part of this thesis introduced detailed models of single neurons, such as the Hodgkin-Huxley [Hodgkin and Huxley, 1952b] and integrate-and-fire (IAF) models [Gerstner et al., 2014b]. These models differ in their biological realism, but generally share the characteristic that they explicitly model action potential (AP), or spike events, and are thus called spiking neuron models. While these models can be used to study network-level phenomena [Brunel, 2000, Vogels and Abbott, 2005], often more abstract single-neuron dynamics are employed [Dayan and Abbott, 2001, Gerstner et al., 2014b]. I briefly describe these here.

Firing-rate models

The activity of a neuron over time can be described as a sequence of spikes using a *neural response function* [Dayan and Abbott, 2001]:

$$\rho(t) = \sum_{i=1}^{n} \delta(t - t_i), \tag{4.1}$$

where each spike is represented by a Dirac delta function, $\delta(\cdot)$. Since the integral of the Dirac delta is unity, we can integrate over a neural response function to obtain a count of the number of spikes that a neuron has fired in a given period of time. Neural responses are typically variable, and so we can describe a neuron's *firing rate*, $r(t)$, as the integral over the trial-averaged response function:

$$r(t) = \frac{1}{\Delta t} \int_{t}^{t+\Delta t} d\tau \langle \rho(\tau) \rangle, \tag{4.2}$$

where $\langle \cdot \rangle$ denotes an average over trials [Dayan and Abbott, 2001]. Thus a neuron's firing rate generally describes the intensity at which it fires spikes at a particular time.

To describe a neuron solely by its firing rate is to suggest that neurons convey signals only by the intensity of spikes, and not by their particular timing (*rate coding*; Brette [2015a]). From this perspective, neurons can be seen as stochastic processes (perhaps Poisson processes), in which on individual trials, spikes act as a noisy realization of an underlying rate [Gerstner et al., 2014b]. The transformation from input to output can then be characterized not by a spiking neuron model, but by a frequency-input (f-I) curve, which describes the firing-rate frequency (f) of a neuron given a particular input (I). The f-I curve of a LIF spiking neuron model is shown in Fig. 4.1A, which displays remarkable similarity to f-I curves of real neurons [Gerstner et al., 2014b].

Taking this to the network level, *firing-rate models* describe the dynamics of populations of neurons using coupled equations of continuous variables which may represent the firing-rate of each neuron, or more commonly each neuron's average input (or average membrane potential) [Wilson and Cowan, 1973, Dayan and Abbott, 2001, Gerstner et al., 2014b]. A typical formulation of a firing-rate network of neurons has dynamics described by the following equation:

$$\frac{d\mathbf{x}(\mathbf{t})}{dt} = -\mathbf{x}(t) + \mathbf{W}\phi(\mathbf{x}(t)) + \mathbf{I}(t), \tag{4.3}$$

where $\mathbf{x}(t)$ is a vector of each neuron's average membrane potential, $\mathbf{W}$ is the synaptic connectivity matrix, $\phi(\cdot)$ is the input-output nonlinearity which turns $\mathbf{x}(t)$ into firing rate, and $\mathbf{I}(t)$ is the time-dependent external input given to the network.

A firing-rate neuronal network model takes the rate coding assumption further by not only assuming that each neuron is well-described by its trial-averaged firing rate, but also that the effect that individual spike patterns of one neuron have on the rest of the network is well approximated by its trial-averaged behavior as well [Dayan and Abbott, 2001]. This relationship can be shown formally (see Dayan and Abbott [2001] Chs. 1 and 7 and references therein). Though firing-rate networks can have input-output nonlinearities which resemble real neuronal f-I curves (Fig. 4.1A), they are often more abstract, and describe activity relative to a baseline value (such as a hyperbolic tangent function; Fig. 4.1B). Perhaps the simplest of

these is the so-called "sign" function, which turns input into a binary output of $+1$ or -1. This exceedingly simple formulation thus describes each neuron as either active or inactive. A classic model that uses this formulation is the Hopfield model, which I describe in more detail at the end of this chapter.

Due to the inherent variability seen in experimental recordings of cortical neurons, rate models appear to be a good abstraction to describe neural circuits in the brain [Britten et al., 1992, Shadlen and Newsome, 1994]. However, other studies suggest that rate coding may be too simplistic, and individual spike times are needed to describe neuronal signal processing [Thorpe et al., 2001]. In either case, rate networks abstract away a lot of the details of real spiking neurons. For example, although arbitrary nonlinearities can be composed to fit the steady-state firing-rates of neurons given particular inputs, rate models may have very different dynamics and transients compared with spiking networks, in general [Gerstner et al., 2014b].

One intriguing proposition suggests that abstract rate networks can be used to describe not individual neurons, but the dynamics of a neural circuit in a lower dimensional, latent space [Pandarinath et al., 2018]. This would also apply to networks of binary neurons, such as Hopfield networks. Considering that this is the case, an interesting area of development is the formulation of analytical tools in order to translate between these different network descriptions [Abbott et al., 2016].

The true utility of firing-rate models of neuronal dynamics lies in their simplicity [Gerstner et al., 2014b]. They are analytically tractable, and allow simple descriptions of networks. They crucially enable us to also describe interesting computations which derive for the most part from connectivity matrices, which means that we can employ linear algebra and linear systems theory to these problems.

Computation in neuronal networks

Neuronal network models act as powerful tools for performing computations, and may also inform us about the dynamics and representations in neuronal circuits of the brain [Barak, 2017]. Network models have been used to describe and provide insights about such phenomena as signal propagation [Vogels and Abbott, 2005], decision

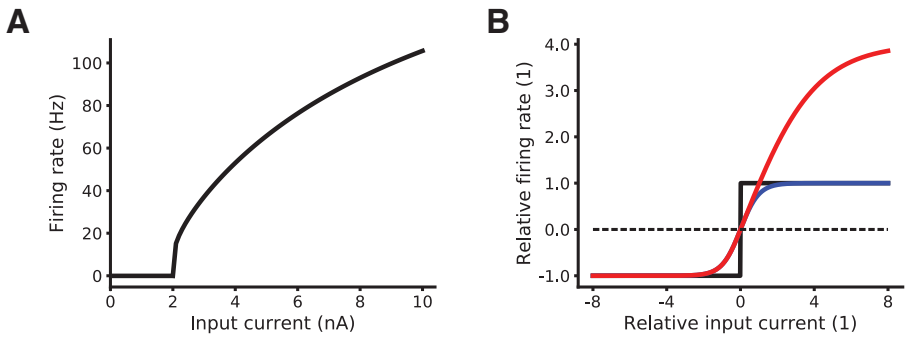


Figure 4.1: Neuronal input-output response functions. **A**: Frequency-input (f-I) curve of a leaky integrate-and-fire neuron. **B**: Input-output functions for abstract neuron models which represent signals relative to a *baseline* level – Heaviside (binary) function in black, a tanh function in blue, and an asymmetric tanh function as in Rajan et al. [2010] and Hennequin et al. [2014].

---

making [Wang, 2002], motor control [Hennequin et al., 2014], and context-dependent processing [Mante et al., 2013]. Many of these models employ supervised learning techniques in order to train the networks to do arbitrary computations [Sussillo and Abbott, 2009, Laje and Buonomano, 2013, DePasquale et al., 2018]. One particular network function which has received a great deal of interest both experimentally and theoretically is associative memory. In the next section, I describe this in detail.

4.2 Memory

One concrete function of many of the circuits in our brains is that of learning and memory. Indeed, as humans, we can recall many experiences throughout our lives, as well as a whole host of facts about the natural world and the people that inhabit it. We also display other forms of learning shared across animals, such as learning a motor skill like running, and associating a punishing stimulus with an effect.

In fact, the word *association* is key for most theories of memory – basic associative learning is the way that organisms learn about causal relationships in the world [Bouton, 2007]. This concept has a rich history dating back to Aristotle and later William James, who proposed that the act of remembering was simply the appropriate linking of a series of items together to form one common memory item

[Gluck et al., 2008]. For example, imagine encountering a banana, and encoding all of the different sensory percepts that it might evoke – color and shape, taste and smell, and more abstract concepts like food and fruit. The memory of a banana is composed of all of these components, and crucially, the association between all of them. This is the basis of what is called *associative memory*.

In this section, I take a slight digression away from neuronal circuits to describe the foundations of memory research. I then return to the circuit description to discuss implementations of memory in neuronal networks.

4.2.1 The psychology of learning and memory

Learning and memory has had a rich history in psychology research, and it has been clear from early on that memory comes in different forms, has different timescales, and relies on different brain areas. It is now well established that there are multiple *memory systems* which act in parallel in any given situation to store particular aspects about a task [Thompson and Kim, 1996, Eichenbaum and Cohen, 2004]. In general, memory can be divided into declarative (explicit) and nondeclarative (implicit) memory (Fig. 4.2).

Implicit memory

Classical conditioning is the most basic form of implicit memory, which was initially studied by Pavlov [Pavlov, 2010]. In these tasks, an animal typically learns to associate an arbitrary stimulus (unconditioned stimulus; US) with a biologically-relevant stimulus (conditioned stimulus; CS) that leads to a behavioral response (conditioned response; CR) [Gluck et al., 2008]. In Pavlov's original experiments, he used various auditory cues such as a buzzer as the unconditioned stimulus, which would reliably predict food (CS), causing a dog to salivate (CR). Classical conditioning continues to be studied, with most experiments focusing on one of two paradigms: the eyeblink reflex response, in which rabbits are trained to associate a tone or light with an air puff to their eye, and the fear conditioning

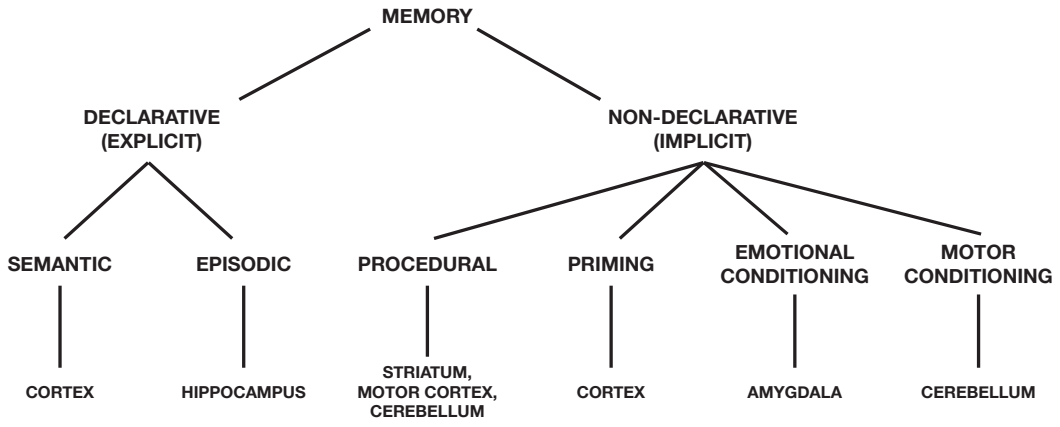


Figure 4.2: Memory systems of the brain. Adapted from Thompson and Kim [1996].

paradigm, in which a tone is associated with an aversive stimulus, such as an electric shock [Bouton, 2007].

The brain areas involved in classical conditioning depend upon the exact nature of the task, and likely involve several areas in the limbic system (including the amygdala, hippocampus and thalamus, among others), brainstem, and possibly areas of the cortex [Thompson et al., 1983]. Those involving a conditioned reflex response such as an eyeblink typically involve the cerebellum, whereas those involving fear will incorporate the amygdala [Gluck et al., 2008]. There is also substantial evidence of hippocampal activation during classical conditioning, which becomes necessary for performance when delays are introduced between the US and CS [Thompson and Kim, 1996].

Following Pavlov, other researchers such as Thorndike, Watson and Skinner introduced *operant (instrumental) conditioning*, in which the organism learns to make (or refrain from making) an action in order to influence the outcome of a task [Gluck et al., 2008]. An example paradigm is the Skinner box, in which a rat learn to operate a lever (or other implement) to receive a food reward. The ideas originally conceived in classical and operant conditioning have led to the modern study of reinforcement learning. Neuroscience research is now focused on finding the neural substrates linking stimulus-response learning, as well as response-outcome prediction.

Evidence suggests that the basal ganglia, and particularly the dorsal striatum, is involved in associating stimuli with responses [O'Doherty et al., 2004], whereas the orbitofrontal cortex (OFC) appears to be involved in prediction of outcomes [Schoenbaum et al., 2009]. Reward prediction in general has also been linked to the dopamine system in the ventral tegmental area (VTA) [Schultz et al., 1997].

Other types of implicit memory include *procedural learning*, which describes the unconscious learning of a skill such as playing a sport or a game like chess, and *priming*, in which exposure to a stimulus may affect the response to a later stimulus [Thompson and Kim, 1996]. These types of learning may involve various areas of the cortex, as well as the cerebellum and basal ganglia [Thompson and Kim, 1996].

Declarative memory

Declarative, or explicit memory refers to information that is consciously accessible, divided into knowledge about facts, called *semantic memory*, and knowledge about particular events, called *episodic memory* [Tulving et al., 1972, Tulving, 2002]. Perhaps the first formal study of this type of memory was done by Ebbinghaus, who studied his own ability to memorize words from a list, and how this recall degrades with time [Ebbinghaus, 2013]. He was able to make some key insights about declarative memory, such as the fact that the semantic meaning behind words influenced the ability to remember them [Gluck et al., 2008].

Later research proposed further details about the structure of declarative memory in the brain. Tolman introduced the concept of the *cognitive map*, suggesting that spatial information is acquired such that relations and distances between things in the world are preserved [Tolman, 1951]. Bartlett further developed this idea, by proposing that we develop *schemas*, which act as scaffolds to incorporate new knowledge [Bartlett, 1995]. Recent work has proposed the cognitive map as a general structure for arbitrary knowledge as a way of structuring semantic knowledge [Constantinescu et al., 2016].

Much of what we know about our declarative memory system is thanks to a single patient named Henry Molaison, known as H.M., who underwent a bilateral removal

of his medial temporal lobes, or hippocampus, in an attempt to treat uncontrollable seizures [Scoville and Milner, 1957, Gluck et al., 2008]. Due to this operation, he developed severe anterograde amnesia (inability to form new memories), and moderate retrograde amnesia (inability to remember old memories). Importantly, H.M. was able to learn new motor skills (procedural memory) and was affected by prior exposure to stimuli (priming), which led to the suggestion that these were separate memory systems [Cohen and Squire, 1980]. Furthermore, H.M. had full mental capabilities and access to his semantic memory, suggesting an integral role for the hippocampus in incorporating new episodic memories into semantic memory.

This led to several hypotheses about the organization of declarative memory in the brain. One such influential theory, called complementary learning systems theory, suggested that information is slowly integrated into semantic memory in the neocortex from the hippocampus, which is able to learn rapidly based on single episodes [McClelland et al., 1995]. Current experimental work is attempting to understand how the cortex interacts with the hippocampus and prefrontal cortex to control memory formation [Eichenbaum, 2017].

Cognitive control and working memory

A final form of memory relates to short-term information that we maintain for brief periods of time, called *working memory* [Gluck et al., 2008]. This type of memory has very limited capacity, and was proposed by George Miller in the 1950s to have a limit of 7 ± 2 items [Miller, 1956]. Working memory is inherently linked with the declarative memory systems of the brain, as these are the types of memories that can be consciously manipulated in short-term memory. An early influential model of working memory, due to Baddeley [Baddeley and Hitch, 1974], suggested that working memory consists of a visuospatial sketchpad for manipulating visual and spatial information, and a phonological loop to do the same for auditory and verbal memories. These are controlled by a central executive which controls which items are selected to be represented in working memory.

Substantial research has pointed to the prefrontal cortex as the locus of working memory and cognitive control [Gluck et al., 2008].

The subtleties of memory

As already mentioned, the fact that the brain possesses different parallel memory systems suggests that several systems may be active at any given time during a task, and may interact with one another. For example, in operant conditioning and the reinforcement learning literature, this fact has manifested itself as *model-free* and *model-based* learning [Dayan and Daw, 2008] (this distinction can also be applied to classical conditioning, Dayan and Berridge [2014]). Model-free learning is normally associated with the dopamine system of the ventral tegmental area as described above, whereas model-based learning appears to recruit the cognitive map structure of the declarative memory system, including the hippocampus and prefrontal cortex [Doll et al., 2012]. Furthermore, it has even been suggested that there is a third memory system in place which relies solely on single or very few trials stored in episodic memory [Lengyel and Dayan, 2008].

Subtleties emerge in learning and recall even within a single memory system. Many factors influence the ability of an organism to learn a new memory. In classical conditioning, phenomena known as *overshadowing* and *blocking* describe how the salience or timing of one cue may affect learning of another cue [Gluck et al., 2008]. For the declarative memory system, as already mentioned, memories are better learned when they fit in with prior knowledge and existing schemas [Bartlett, 1995, Ebbinghaus, 2013].

Memory retrieval also depends on many factors [Gluck et al., 2008]. First of all, retrieval may come in different forms based on how the memory is probed (here I am focusing on declarative memory in particular). *Recognition* refers to the scenario in which the original stimulus that caused the memory is presented again, and a subject merely needs to remember that the stimulus had been seen before. *Cued recall* refers to a case in which the subject is given a contextual cue that prompts retrieval of the correct memory. Finally, *free recall* refers to the free

ability to retrieve any unprompted memory. Intuitively, the more cues that are given the more likely the correct memory is to be retrieved.

This highlights a very important point, which is that *context* is a very important determinant for both how things are learned, as well as how they are later retrieved, and even relearned (or forgotten) [Smith, 1979, Bouton, 1993]. Consider the case of *extinction* in classical and operant conditioning, which describes the process of unlearning when a stimulus is presented without the reinforcer (or an action no longer produces the expected outcome). Often, if the background context differs slightly from the initial period of learning, a second memory (called an extinction memory) will be learned on top of the original memory, thus counteracting its effect but not removing it [Quirk and Mueller, 2008]. This is an issue for people suffering from post-traumatic stress disorder [Shin and Liberzon, 2010]. This examples stresses a critical points about the stability of memories – it appears as though there is a period of time when new memories are learned when they are vulnerable to change. These memories become stable through a process of *consolidation*. Furthermore, old memories that are successfully retrieved may undergo a similar process called *reconsolidation* and will become vulnerable again [Alberini, 2005]. Thus, each time a memory is recalled, it may be changed. This may serve useful purposes such as to restructure the memory so as to reduce interference with other memories.

4.2.2 The neuroscience of learning and memory

The neural substrates of learning and memory have been a major topic of neuroscience research since the early days of the field. Generally speaking, retrieval of a memory is considered to be equivalent to the activation of a neuron (or more often a population of neurons) which *represents* the memory [Milner, 1996], sometimes called an *engram* [Poo et al., 2016].

Synaptic plasticity

One of the most pervasive ideas in neuroscience, due to Donald Hebb, is an attempt to explain how learning actually modifies neural circuits to encode

new memories. Hebb proposed that when one neuron A consistently takes part in the activation of another neuron B, then the efficiency by which neuron A affects neuron B should increase [Hebb, 1949] (often paraphrased as "cells that fire together, wire together"; Shatz [1992]). Currently, Hebbian learning simply refers to synaptic plasticity rules which depend on a co-activation of presynaptic and postsynaptic activity [Gerstner et al., 2014a]. This idea continues to have a strong influence on the field and guides much (but not all) of the research on learning and memory in neuroscience.

Experimental studies on plasticity have shown some evidence for Hebb's postulate starting in the early 1970s, where it was discovered that high-frequency stimulation of neurons in the hippocampus caused long-lasting increases in the amplitude of post-synaptic potentials, known as long-term potentiation (LTP; Bliss and Lømo [1973]). This was also found in the cortex [Bear and Kirkwood, 1993]. Long-term depression was found later [Lynch et al., 1977, Dudek and Bear, 1992]. It was later discovered that plasticity depends upon the relative timing of pre- and post-synaptic spikes, known as spike-timing dependent plasticity (STDP) [Bi and Poo, 1998, Markram et al., 1997].

It is now well established that different forms of synaptic plasticity exist in the brain, which may depend on cell type, brain area and the age of the organism [Abbott and Nelson, 2000]. Other forms of plasticity, such as those relating to homeostasis have also been found [Turrigiano, 2012]. Finally, it appears as though neuromodulators and other molecular pathways also influence plasticity [Gu, 2002]. The current picture of synaptic plasticity and learning in the brain is that of a sophisticated system which is concerned with many variables. Overall, one major theme that has emerged is how the brain deals with quickly integrating new information while keeping old memories stable and accurate (the *plasticity-stability* dilemma; Abraham and Robins [2005]).

Despite all of the complexity, much research is directed towards demonstrating Hebb's postulate and finding evidence of a memory engram [Josselyn et al., 2015]. This engram may take on different forms, but is usually theorized to involve a

persistent state, either at the activity or molecular level [Chaudhuri and Fiete, 2016]. Besides synaptic changes, competing theories of memory suggest that other cellular changes may be the locus of plasticity [Titley et al., 2017].

Network-level phenomena

Network-level phenomena may also influence the onset and expression of plasticity and memory activation. Inhibition is a strong candidate for both blocking plasticity from occurring and memory expression [Hennequin et al., 2017]. Some recent work has suggested that inhibition acts to form parallel engrams of inhibition [Barron et al., 2016]. Circuit mechanisms such as these may play a role in where memories are stored, and other aspects related to context-dependence [Holtmaat and Caroni, 2016, Kuchibhotla et al., 2017]

4.2.3 Computational models of associative memory

Historical perspective

Early models of learning and memory focused on explaining results seen in classical conditioning experiments. One such influential model, called the *Rescorla-Wagner* model was designed to predict the strengths of weights that associate unconditioned stimuli with an unconditioned stimulus [Rescorla et al., 1972]. This model was able to explain several behavioral results seen in classical conditioning experiments, such as extinction, blocking and overshadowing [Dayan and Abbott, 2001].

The basis for models of learning and memory at the neuronal level arguably began in the 40s and 50s with simple binary neurons such as the McCulloch-Pitts and perceptron models [McCulloch and Pitts, 1943, Rosenblatt, 1958]. These ideas developed into what became known as *connectionist* models of neuronal processing [Rumelhart and McClelland, 1986], in which computations are distributed across networks of connected units. Throughout this time, people became interested in associative content-addressable memory [Hertz et al., 1991], in which memories form as the association between different inputs [Grossberg, 1968, Willshaw et al., 1969]. Early models of this form related memory in networks to physical systems, such as

magnetic systems [Cragg and Temperley, 1955, Little, 1974, Hopfield, 1982]. Most of these studies modelled memories as particular network states (e.g., the activation of particular sets of neurons), which are stable and reliably respond to particular inputs.

Types of memory networks

More recently, the theoretical neuroscience field has delved into several aspects of what is actually needed for a memory to exist in a circuit. This is nicely expressed in a recent review from Chaudhuri and Fiete [2016], where the requirements for memory are the following:

- States that persist over time
- Sufficient capacity for memory states
- Different inputs should trigger different memory states
- States should be robust to noise, and different memories should not interfere
- Stored memories should be correctly retrievable

Several of these requirements are in conflict with one another (such as having high capacity yet robustness and low interference), which makes this problem non-trivial. These requirements can be satisfied in different ways (Fig. 4.3). I will discuss a few of the most common mechanisms below.

Bistable switches and discrete attractors

Perhaps the most common form of memory in computational models is the bistable switch, which relies on positive feedback and a nonlinear unit, which causes stable states that may either be activated or inactivated depending upon the input (Fig. 4.3B). The most well-known example of this type of memory is the Hopfield network [Hopfield, 1982, Little, 1974, Cohen and Grossberg, 1983], in which several discrete attractor states are stored in the connections between neurons. This model is the focus of the next chapter, and so I discuss this model in more detail below in section 4.2.4.

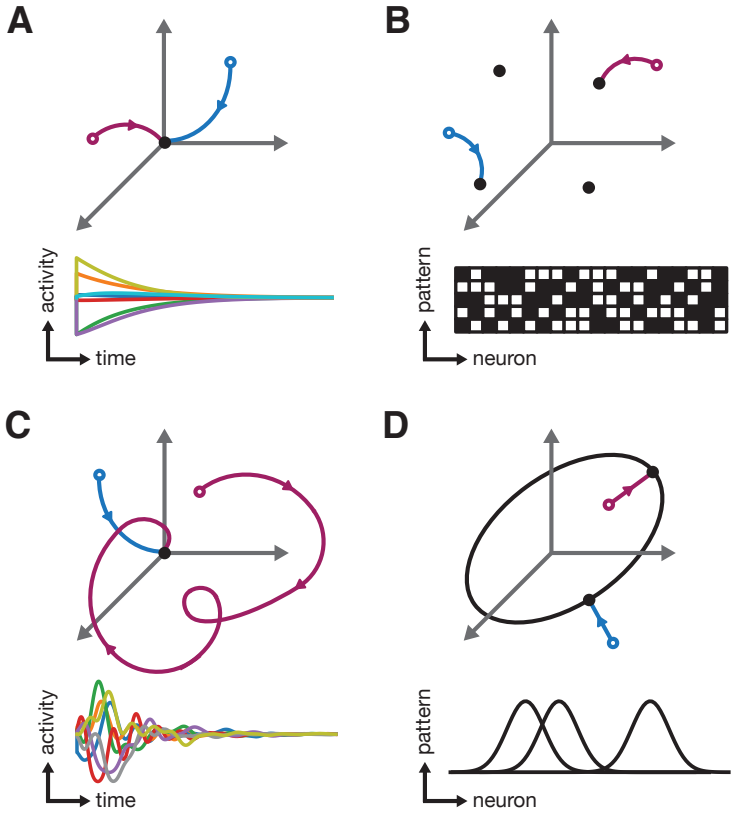


Figure 4.3: Types of transient and attractor memories. **A**:Weak to moderate-strength connectivity can produce long transients returning back to a single fixed point at zero. **B**: Strong low-dimensional connectivity can embed discrete attractors into the state-space, such that network activity flows to the closest one. **C**: Strong balanced connectivity produces rich transients which may amplify particular input directions selectively. **D**: Structured networks may display continuous attractor states. Adapted from [Chaudhuri and Fiete, 2016] Fig. 2.

Discrete attractors display several interesting properties which make them appealing as memory storage mechanisms. For example, they are very robust to external noise or perturbations in connectivity strength, due to the fact that memories are coded as strong attractors with saturating activity, and nearby states will move to the attractor [Chaudhuri and Fiete, 2016]. Second of all, these types of networks display a high memory capacity (see below for more details).

There are many examples of bistable switches in biology [Cherry and Adler, 2000], and there is some evidence that persistent activity during working memory may have a similar mechanism [Zylberberg and Strowbridge, 2017]. Despite the

popularity of the Hopfield model, there is no conclusive evidence that this type of activity exists in the brain [Harris, 2005, Wallace and Kerr, 2010].

Continuous attractors

Instead of storing a set of separate discrete attractors, positive feedback can also be tuned to produce a continuum of stable states, i.e., a line attractor (Chaudhuri and Fiete [2016]; Fig. 4.3D). This type of model has been used to describe orientation tuning in the visual cortex [Ben-Yishai et al., 1995], stability in the oculomotor system [Seung, 1996], and working memory in the prefrontal cortex [Wimmer et al., 2014]. Continuous attractors are not as robust to noise as discrete attractors, and may display drifts along the attractor due to external noise or weight perturbations [Burak and Fiete, 2012, Wimmer et al., 2014].

Long transients and feedforward structures

In contrast to the idea of memory as a stable, persistent state, some studies have proposed memory as transient patterns of activation [Rabinovich et al., 2006, Chaudhuri and Fiete, 2016]. Networks with weak and stable connectivity can be tuned to produce slowly-decaying transients taht decay back to baseline (Fig. 4.3A). These types of networks may rely on positive feedback Maass et al. [2002] or negative derivative feedback Lim and Goldman [2013]). Other networks may rely on stronger excitatory connections balanced by strong inhibition to create feedforward pathways with rich transient dynamics (Fig. 4.3C; Goldman [2009], Hennequin et al. [2014]).

Such networks always return to baseline, and so memories gradually decay depending upon the intrinsic timescales of the dynamics. Noise is integrated in the system over time, and so also contributes to memory decay, though it will have different effects depending on if the direction corresponds to fast or slow timescales Lim and Goldman [2012]. Evidence for this type of activity has been seen in studies of the primate motor system [Churchland et al., 2012] and insect olfaction [Bazhenov et al., 2001].

4.2.4 The Hopfield model

Models of associative memory date back to the 1960s and 1970s, with important contributions made by Willshaw, Grossberg, Little and others [Grossberg, 1968, Willshaw et al., 1969, Little, 1974]. In the early 1980s, Hopfield revisited this topic and stressed the relation of associative memory models to physical systems with *energy functions*, where each memory acts as a stable attractor in the system [Hopfield, 1982]. An energy function, also known as a Lyapunov function, is a real-valued function of all states of the network which specifies how the dynamics will evolve – they will always move towards states of low energy[Steven, 1994]. The model consists of a fully-connected network of N binary neurons, taking on values of -1, for inactive, and $+1$ for active. These units are denoted S_i for neuron i, with dynamics

$$S_i = \text{sgn}\Big(\sum_j J_{ij} S_j\Big) \tag{4.4}$$

where J_{ij} is the synaptic connectivity between neurons i and j, and

$$\text{sgn}(x) = \begin{cases} 1 & \text{if } x \geq 0, \\ -1 & \text{if } x < 0. \end{cases} \tag{4.5}$$

There are two methods for updating the dynamics of the network – *synchronous* and *asynchronous* [Hertz et al., 1991]. In the synchronous scheme, all neurons are updated simultaneously using the state of the network in the previous time step. The asynchronous scheme, in contrast, updates one neuron at a time, either using a fixed order or randomly. The choice of update scheme may have important effects on the convergence of the model.

By analogy with a mathematical model of a magnetic system studied in physics, called the Ising model, Hopfield proposed the following energy function for the network

$$H = -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} J_{ij} S_i S_j. \tag{4.6}$$

It can be shown that the network will converge to a low-energy state (i.e., the energy will never increase) if the dynamics are followed according to Eqn. (4.4) above

with two constraints – the synaptic connectivity must be *symmetric* ($J_{ij} = J_{ji}$) and the updates must be made *asynchronously*.

The network stores a set of p memory patterns, denoted ξ_i^μ for the ith unit of pattern μ, using the following learning rule:

$$J_{ij} = \frac{1}{N} \sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu, \tag{4.7}$$

with self connections often being set to zero ($J_{ij} = 0$ for $i = j$). This learning rule is often referred to as a "Hebbian" learning rule, as it depends upon correlations between pre-synaptic and post-synaptic neurons in order to set the synaptic weight.

Hopfield observed that the capacity of the network to store stable memory patterns scaled with the network size, N. It thus makes sense to think of the memory capacity in terms of the number of patterns stored per number of neurons in the network, often called the load parameter: $\alpha = p/N$. A few years following Hopfield's paper, the *critical* memory capacity (α_c) of the network was estimated using mean field methods by Amit et al. [1985] and found to be $\alpha_c = p_{\max}/N \approx 0.138$ (in the zero noise limit). Using methods from statistical physics, researchers studied the state space of the network and discovered additional spurious low-energy states in the network dynamics such as mixtures between different memories. When the memory load is too high, these spurious states overwhelm the memory states, and the network is no longer able to perform memory recall. This phenomenon is sometimes referred to as *blackout catastrophic interference* [Amit, 1992].

4.2.5 Extensions and biological realism

Since the original formulation of the Hopfield network, many follow-up papers have been published, exploring new variants of associative memory, as well as adding additional biological constraints to the network. I briefly mention a few highlights here.

One important direction that was taken following Hopfield's original work was the search for better memory performance, often through the study of other learning rules. In the late 1980s, Gardner [1988] drew a parallel between associative memory

networks and perceptrons – each neuron in the network can be thought of as solving a series of binary classification tasks, one for each memory pattern. Through this analysis, she was able to derive the absolute maximum capacity of the Hopfield network for any arbitrary setting of the weights, $\alpha_{\max} = 2N$, and proposed an iterative perceptron learning algorithm to enable networks to reach this maximum [Gardner, 1988]. Other recent work has further expanded the capacity of associative memory networks by, e.g., proposing rules to learn an exponential number of patterns, instead of scaling linearly with network size [Hillar and Tran, 2018].

Studies have attempted to add more biological realism to Hopfield networks in several different ways. This included changes to the synaptic connectivity of the model – adding asymmetry, weight clipping or bounding, and sparsity or modularity to the connections to reflect what is known about cortical networks of neurons [Derrida et al., 1987, Sompolinsky, 1987, Hertz et al., 1991, Renart et al., 1999]. Importantly, when synaptic weights are bounded, as is almost certainly the case in the brain, the memory capacity of the network is degraded by a substantial amount [Fusi and Abbott, 2007].

Furthermore, the formulation of the binary neural activity itself was also modified, both from $\{\pm 1\}$ units to $\{0, 1\}$ units, and also moving to continuous-valued units, resembling typical recurrent firing-rate networks [Cohen and Grossberg, 1983, Hopfield, 1984]. Finally, another important development was the study of correlated patterns [Amit et al., 1987a, Hertz et al., 1991] – these are sets of patterns for which

$$\sum_{i=1}^{N} \xi_i^\mu \xi_i^\nu \neq 0, \text{ for } \mu \neq \nu. \tag{4.8}$$

The simplest case of correlated patterns is through the introduction of a bias in the probability that each unit of a pattern is $+1$ or -1. For example, low-activity networks set each memory pattern to contain a small number of active neurons ($\Pr(\xi_i^\mu = +1) \ll 0.5$). It was found that low-activity networks have a much higher capacity than the standard Hopfield network (with a slight tweak in the learning rule), scaling with $1/\ln(a)$ (where a is the activity level), owing to the reduced interference between sparse memories [Tsodyks and Feigelman, 1988].

The legacy of the Hopfield model has had a large influence on the computational neuroscience community, as well as the neuroscience community in general. Computational studies today still employ many of the concepts established in the Hopfield model and later papers (e.g., Romani et al. [2013], Brunel [2016], Rubin et al. [2017]). Recent work also attempts to study attractor-like memories in more realistic networks, such as spiking models [Amit and Tsodyks, 1991, Curti et al., 2004, Zenke et al., 2015], and those with separate excitatory and inhibitory populations [Chapeton et al., 2012]. Currently, experimental studies are attempting to find evidence for discrete attractor memory engrams in the brain [Carrillo-Reid et al., 2016, Pfeiffer and Foster, 2015], however, it is still unclear if such states exist. In any case, the simplicity of the model (as well as the lack of well-established alternatives) makes the Hopfield network still one of the most useful models to understand memory formation and recall, as well as for introducing new concepts and ideas into memory research. In the next chapter, I use this model as a basis to explore context-dependence in memory.

5

A model of context-dependent associative memory

Abstract

An increasing amount of evidence suggests that memory formation and retrieval are modulated by contextual signals, such as behavioral or emotional state. However, typical models of associative memory do not incorporate this dependency. Here we propose an extension to the Hopfield network which takes into account contextual modulation. Each memory pattern is assigned to one of a set of contexts, which is defined by a particular network state with a configuration of available neurons and active inputs per neuron. Only one context is active at any given time, thereby reducing interference from "irrelevant" memories found in other contexts, which remain dormant. We show that these context-modular Hopfield networks have substantially increased memory capacity, while retaining high robustness to noise and increased robustness to memory overloading. Network performance depends on the number of contexts, as well as the number of active neurons and inputs—when chosen optimally, the capacity is up to sixty times greater than the standard Hopfield model. Improved performance comes at the cost of limited retrieval, because only memories stored in the active context can be recalled. To address this, we propose a system in which a controller network dynamically switches the memory network to a desired contextual state before storage or retrieval. We successfully show that this system is able to bias memory retrieval based on context. Overall, our work illustrates the benefits of context-dependent memory, and may have implications for our understanding of cortical memories and their interaction with contextual signals in the prefrontal cortex and hippocampus.

5.1 Introduction

Humans and animals are remarkably adaptable in their behavioural interactions with the world [Alcock and Rubenstein, 1989]. Correspondingly, our underlying neural circuitry is inherently flexible, which enables sensory signals to be gated or attended to [Reynolds and Chelazzi, 2004, Khan and Hofer, 2018] and decision making circuits to integrate different features based on context [Mante et al., 2013]. These contextual cues may relate to anything from external sensory information and spatial location to internal emotional and cognitive states [Robertson et al., 2015]. Memory is no exception to this phenomenon, and many behavioral studies have demonstrated the effects of context on memory storage and retrieval for Pavlovian conditioning [Bouton, 1993] and free recall [Smith, 1979]. This, in turn, has led to a host of models of context-dependent memory in the psychology literature [Howard and Kahana, 2002, Sederberg et al., 2008, Gershman et al., 2017].

The brain areas and circuit mechanisms that enable contextual flexibility of memory are just beginning to be explored. Recent studies have shown evidence for inhibitory masking of cortical memories [Barron et al., 2016] as well as a role for the hippocampus, amygdala, and areas of prefrontal cortex in controlling context [Xu et al., 2016, Guskjolen et al., 2018, Cowansage et al., 2014]. However, traditional circuit-level models of auto-associative memory often do not incorporate context-dependence [Hopfield, 1982]. Instead, all memories are typically stored using the same representation, and are able to be recalled at any time given the right input (but see [Doboli et al., 2000, Perumal and Minai, 2009]).

In this work, we propose a new model of context-dependent associative memory which integrates recent experimental findings about neural circuits with theoretical studies on hierarchical and modular associative memory [Gutfreund, 1988, Tsodyks, 1990, Mari and Treves, 1998]. In this model, associative memory is modularized into overlapping contextual states which each store a separate set of memories, with only one being active at a given time (Fig. 5.1A). We show that this modular architecture, in concert with specialized inhibition which suppresses irrelevant memories, results in enhanced memory capacity as well as robustness to memory

overloading, providing a direct benefit for organizing memories around context. The key to this enhancement lies in the sparsity of memories and their interconnectivity. This is a phenomenon which has been recognized and utilized in many previous studies [Tsodyks and Feigelman, 1988, Treves and Rolls, 1991, Okada, 1996], but has not yet been applied to context-dependence.

Such a model requires that contextual signals are explicitly encoded in the brain [Robertson et al., 2015]. Evidence suggests that contextual signals interact with cortical memories through the hippocampus and prefrontal cortex [Eichenbaum et al., 1999, Smith and Mizumori, 2006, Eichenbaum and Cohen, 2014, Eichenbaum, 2017]. We incorporate this into the model by explicitly including a "context-encoding" network which interacts with the associative memory network. The context-encoding network is able to switch the associative memory network into different states, leading to a model which can dynamically gate memory expression.

We hypothesize that the circuit mechanisms behind the interaction of top-down contextual signals with memory activation relate to recruitment of specific inhibitory motifs found in cortical circuits. Recent experimental studies indicate that contextual signals may be relayed to many brain areas through top-down modulation [Harris, 2013], and control through different inhibitory cell types [Letzkus et al., 2015, Kuchibhotla et al., 2017]. Computational and theoretical studies have begun to explore these aspects for the purposes of working memory, sensory learning and gating, and decision making [Edin et al., 2009, Yang et al., 2016, Costa et al., 2017, Guerguiev et al., 2017, Sacramento et al., 2017]. We propose here that similar mechanisms can be used for associative memory in the cortex and hippocampus (Fig. 5.1B), in which different inhibitory cell types, as well as top-down excitation, may place the network into different states for storage and retrieval of memory [Holtmaat and Caroni, 2016].

Overall, this work provides a step towards linking high-level principles of context-dependent memory with lower level circuit models that often overlook this phenomenon. We show that such an organization has inherent benefits relating to memory storage and robustness. Furthermore, the interaction of the

associative memory circuit with a context-encoding network may have implications and predictions for our understanding of contextual signals in the medial prefrontal cortex and hippocampus, as well as the dynamic control of memory expression, as seen experimentally.

5.2 Results

Our model of context-dependent associative memory (Fig. 5.1A) is based on the Hopfield model [Hopfield, 1982, Tsodyks and Feigelman, 1988, Hertz et al., 1991], consisting of a recurrent network of N neurons, which can be in an elevated or suppressed state (taking values of 1 or 0, respectively; see Methods for details). We refer to it as the *context-modular* Hopfield network.

To incorporate context-dependence, we define a set of contextual states, which control neural activity and connectivity in the following two ways. First of all, each contextual state defines a corresponding subnetwork of *available neurons* (N_s of the total N neurons, with parameter $b_s = N_s/N$) with all other neurons receiving strong enough inhibition to keep them in a suppressed state. Second of all, each context defines a subset of *available inputs* per neuron (N_d of the total N_s inputs on average, with parameter $b_d = N_d/N_s$), with other connections being temporarily suppressed by strong dendritic inhibition. Contextual states are overlapping, and each neuron and connection may participate in several contexts. There are s distinct contexts, and each one has a corresponding *separate* set of p memory patterns, making $P = sp$ total memories stored in the network. Patterns are chosen uniformly at random, with each neuron independently taking on values of 0 or 1 with probability 0.5. The synaptic connectivity between neurons is symmetric, and is defined using a "Hebbian" formulation [Hopfield, 1982] (see Methods).

Only one context is active at any given time, with the corresponding contextual state imposed on the memory network externally by a separate context-encoding network, which dynamically interacts with the memory network (Fig. 5.1A). We hypothesize that contextual state is coded through top-down signals which both directly excite subsets of principal cells in the memory network, as well as inhibit

others through local inhibition (Fig. 5.1B). The distinct target areas of different inhibitory neurons provides a mechanism for the two means of controlling contextual state (Fig. 5.1B). Perisomatic inhibition, through e.g., parvalbumin-expressing (PV) interneurons, may act to define contextual subnetworks of active neurons by suppressing the firing of particular principal cells, whereas dendritic inhibition, through e.g., somatostatin-expressing (SST) interneurons, may act to define the particular set of available input connections onto each principal cell (Isaacson and Scanziani [2011]; see Discussion for more details on biological relevance).

We thus propose three possible types of contextual modularity on the network connectivity (Fig. 5.1C): subnetwork selection through PV inhibition, input selection through SST inhibition, and the combination of the two. From the perspective of the principal cells coding for the memory, we can view these mechanisms as temporarily modifying the network such that there is a different *effective* connectivity matrix for each contextual state. Subnetwork selection effectively removes specific rows and columns from the connectivity (Fig. 5.1C, left), whereas input selection removes particular entries in the connectivity matrix for each neuron, thereby making it more diluted, or sparse (Fig. 5.1C, middle). The combination of the two results in a smaller and sparser connectivity matrix (Fig. 5.1C, right), with a potentially large number of synaptic connections remaining hidden, to be used in other contexts. Importantly, for the moment we consider that contextual subnetworks and input configurations are chosen *randomly*.

In the following, we study the characteristics of each of these architectures and their effects on such properties as memory capacity and robustness. Throughout, we will make comparisons with the standard Hopfield model [Hopfield, 1982], as well as the low-activity Hopfield model [Tsodyks and Feigelman, 1988] (see Methods for details). To start, we do not explicitly model the context-encoding network, nor the specific inhibitory mechanisms, but instead focus on the resulting connectivity matrix and its properties assuming that the network is in the correct contextual state. We return to the issue of encoding context explicitly later in the section on *dynamic gating* of memory expression.

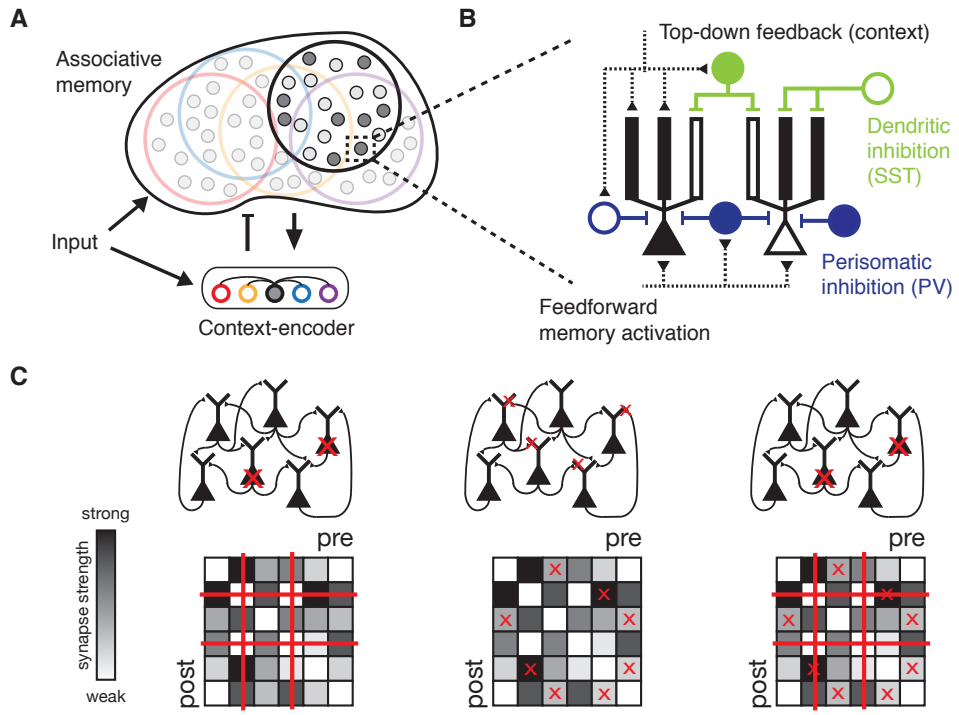


Figure 5.1: Schematic of the context-modular Hopfield network. **A**: Neurons are arranged into contextual configurations of overlapping subnetworks of *available neurons* (colored rings) and *active inputs* (not shown, see **B**), each storing a separate set of memory patterns. Only one contextual subnetwork is active at a time (black ring), as controlled by an external context-encoding network, such that memories outside of the active context remain dormant. **B**: Contextual configuration is imposed through top-down feedback, which controls local inhibition and background excitation, such that identical feedforward memory activation may cause different patterns to emerge. We propose that subnetwork selection is implemented through perisomatic inhibition (PV, blue), and input selection is implemented through dendritic inhibition (SST, green), though not implemented explicitly here. Recurrent excitatory connections are not shown. **C**: Contextual configurations change the effective connectivity matrix of the associative memory network: subnetwork selection removes particular columns and rows from the matrix (left), dendritic input selection removes particular elements from the matrix (center), and together, they will implement both effects (right).

5.2.1 Context-modular Hopfield networks with subnetwork selection exhibit enhanced memory capacity

Starting with the model of subnetwork selection only (Fig. 5.1C, left; $b_s \leq 1$, $b_d = 1$), we studied the memory capacity properties using established signal-to-noise and heuristic mean-field methods [Amit et al., 1985, Hertz et al., 1991]. Memory capacity denotes the maximum number of stored patterns divided by the network size, such that each memory pattern is a stable attractor of the

network dynamics (see Methods). Here, we have two notions of capacity. The capacity per context is defined as

$$\alpha = p/N_s, \tag{5.1}$$

i.e., the number of patterns stored in each subnetwork (p) divided by the number of active neurons in the subnetwork (N_s). For simplicity in the calculations, we impose that all subnetworks have the same size and store the same number of patterns. The total network capacity is defined as

$$\gamma = sp/N = \alpha s b_s, \tag{5.2}$$

which is the total number of patterns (sp) divided by the full network size (N), or equivalently, the capacity per context (α) times the number of contexts (s) times the ratio of subnetwork size ($b_s = N_s/N$).

We analyze the memory capacity from the perspective of a single context, and then extrapolate to determine the total network capacity by treating all subnetworks as equivalent. We illustrate the capacity calculation here using a signal-to-noise analysis (Hertz et al. [1991]; see Methods and Supplementary Materials for the mean-field derivation).

Consider that a particular memory is currently being recalled, i.e., the network is currently in one of the memory states – specifically, pattern ν of context k. We can study the stability of this memory pattern by estimating the total input that a particular neuron i will receive, which we denote as $h_i^{k\nu}$, assuming that this neuron should be in the elevated state (i.e., it should have a value of 1 – this is an arbitrary choice, the same analysis can be done assuming the opposite). It can be shown that this input is well approximated as

$$\begin{aligned} h_i^{k\nu} &\approx 1 + \mathcal{N}\Big(0, p/N_s\Big) + \mathcal{N}\Big(0, (s-1)pb_s/N\Big) \\ &= 1 + \mathcal{N}(0,\alpha) + \mathcal{N}(0,\gamma b_s), \end{aligned} \tag{5.3}$$

where $\mathcal{N}(\mu, \sigma^2)$ is a normally-distributed random variable with mean μ and variance σ^2 (see Methods 5.4.3). In order to keep this neuron in the elevated state, the

total input must be greater than zero. Thus, if we can compute the probability that the last two terms are less than -1, we can have an idea of how likely this neuron will turn off – meaning that the memory is not stable.

The two normally-distributed terms represent two different sources of noise in the network connectivity that interfere with the stability of pattern ν of subnetwork k (often called "crosstalk" terms). The first is the noise from the other $p-1$ patterns of context k. This term thus scales with the number of patterns per context (p). The second term represents the noise coming from all of the $(s-1)p$ other patterns in the other contexts. Thus, this term scales with the total number of other patterns, but also with the relative amount of shared active neurons, i.e., the ratio of subnetwork size (b_s). Keeping the probability of inactivating this neuron low, we arrive at an estimate of the maximum capacity per context of

$$\alpha = \frac{\alpha_H}{1 + (s-1)b_s^2}, \tag{5.4}$$

where α_H is the maximum capacity of the standard Hopfield network ($\alpha_H \approx 0.138$).

In addition to the signal-to-noise analysis described here, we also studied the memory capacity of this network using mean field methods, which provide a more accurate estimation ([Amit et al., 1985, Hertz et al., 1991]; see methods for details). The results of this calculation for the capacity per context are plotted in Fig. 5.2A-C, and followed the intuition gained from the signal-to-noise analysis of Eqn. (5.4). The denominator in Eqn. (5.4) is always greater than or equal to 1, and so $\alpha \leq \alpha_H$. Thus, the subnetwork capacity is upper bounded by the standard Hopfield capacity (green line & arrow, Fig. 5.2A-C), and decreases as the number of subnetworks grows large or the subnetwork size gets closer to the full network size. This should make intuitive sense, as each subnetwork can be thought of as a standalone Hopfield network of size N_s with p patterns, but with more noise in the weights due to the influence of the other $(s-1)p$ memories. The second term in the denominator of Eqn. (5.4), $(s-1)b_s^2$, functions as a measure of the amount of overlap between subnetworks (see Fig. 5.11A,C,E). Areas of parameter space where capacity is high will coincide with low overlap between contexts.

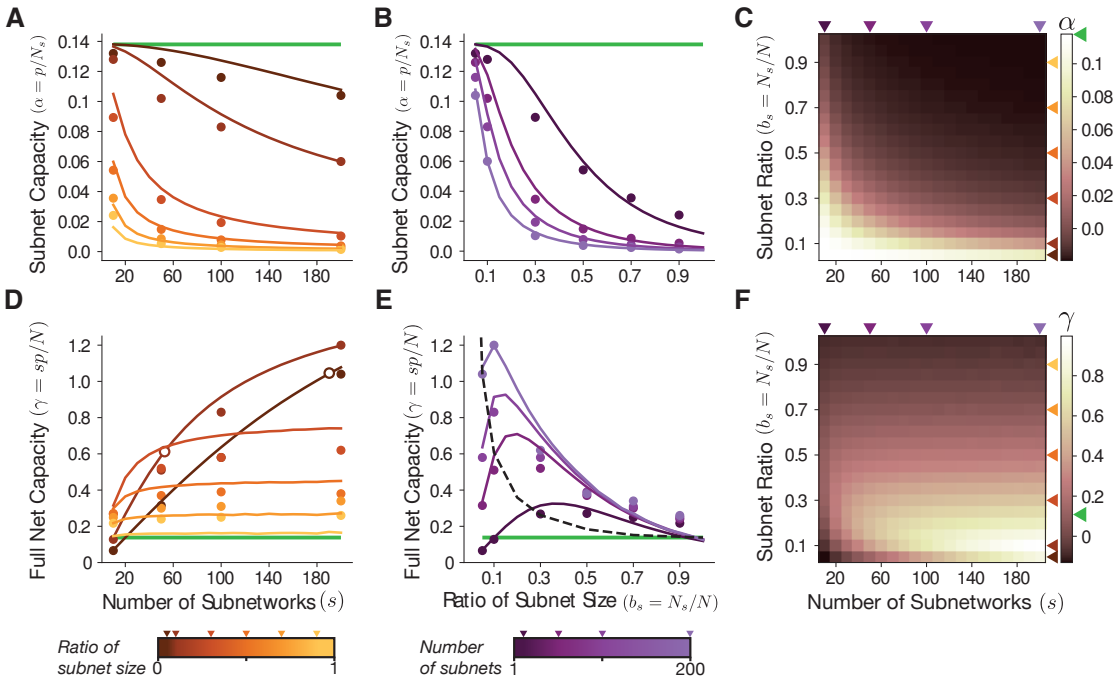


Figure 5.2: Memory capacity of the context-modular Hopfield network with subnetwork selection. **A-C**: Mean-field capacity estimation (solid lines) and numerical simulations (data points) for a single context ($\alpha = p/N_s$), plotted as a function of the number of contexts (s; **A**), the relative subnetwork size of each context ($b_s = N_s/N$; **B**), and over the full range of parameters (**C**). Lines plotted in **A,B** are slices through **C** indicated by the colored triangles. Comparison is made with the standard Hopfield network (green lines, **A,B**; green arrow, **C**). **D-F**: Same as **A-C** but for full network capacity ($\gamma = sp/N = \alpha s b_s$). Memory capacity of low-activity Hopfield networks are plotted as open circles overlayed on the line with same activity level $a = b_s/2$, and as the dotted black line in **E**, again as a function of activity level. $N = 10000$ for all simulations.

Although subnetwork capacity remains below the standard Hopfield model for most parameter ranges, the total network capacity γ, shown in Fig. 5.2D-F, grows well above α_H for many parameter values, to almost an order of magnitude higher when chosen optimally. We thus see that the network as a whole has substantially increased memory capacity, owing to the fact that there is less interference between memories found in different contexts. Importantly, this reduced interference depends upon the fact that the majority of memories are not retrievable in each context.

The theoretical capacity results were confirmed numerically by simulating context-modular Hopfield networks of size $N = 10000$ over different numbers of subnetworks (s) and relative subnetwork size (b_s) (see Methods for simulation

details). These numerical results showed a qualitatively good match (data points in Fig. 5.2A,B,D,E). Discrepancies between the numerics and theory arise due to finite-size effects, as well as approximations in the theory.

This architecture takes advantange of reduced interference by having *low-activity* memories, as the number of active neurons scales with b_s (activity level $a = \frac{1}{2}b_s$). It is known that memory capacity dramatically increases with low-activity patterns, and so we compared our results here with the low-activity version of the Hopfield network [Tsodyks and Feigelman, 1988]. As shown in Fig. 5.2E (dotted line), the capacity of the low-activity network is comparable to that of the context-modular network, and can even surpass it, but only for very low activity levels. When the activity level is made to match that of the context-modular case, it is outperformed when the number of subnetworks s is increased (Fig. 5.2D, open circles, shown for $b_s = 0.05, 0.1$).

Given that the subnetwork size puts a limit on the number of neurons that can participate in each memory, one concern that arises is the information content of the resulting memories [Amit et al., 1987a]. We thus performed a simple calculation of the information content of all memories, following Gardner [1988]. As shown in Fig. 5.3, the relative information content of the context-modular Hopfield network with subnetwork selection is similar to the standard Hopfield network (as well as that of the low-activity Hopfield network, shown in Fig. 5.10C). Though additional information is needed for the context-modular network to be in the correct contextual configuration, as long as the number of contexts (s) is much smaller than the network size, this will affect the information content minimally. Thus, this modular architecture based on context offers a robust increase in memory capacity with higher activity compared to the low-activity variant, and thus retains higher information content.

5.2.2 Modulation through dendritic input selection does not enhance capacity

We next studied the context-modular Hopfield network which has selective input modularity, but no subnetwork modularity (Fig. 5.1C, middle; $b_s = 1$, $b_d \leq 1$). In

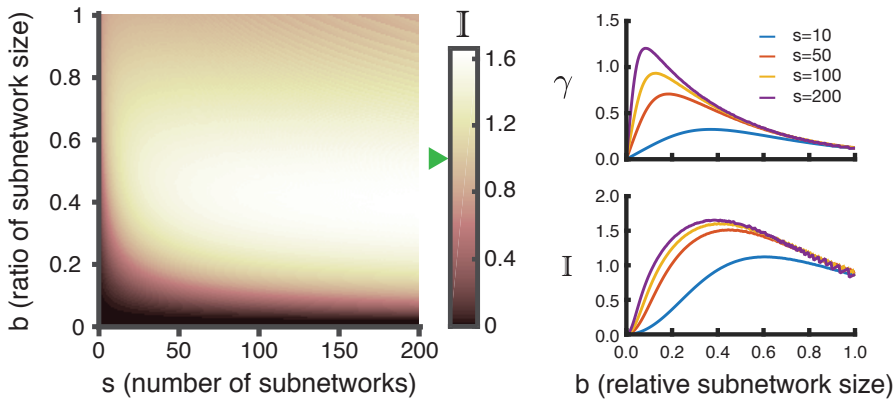


Figure 5.3: Relative information content of context-modular Hopfield network with subnetwork modulation as compared to the standard Hopfield network (see Methods). Vertical slices through the 2D plot are shown (lower right) and compared with full network memory capacity (upper right).

---

essence, this makes each contextual configuration a sparse (or diluted) Hopfield network with some shared weights with the other networks. The capacity measures remain the same as before from Eqn. (5.1) and Eqn. (5.2). As before, we analyze the capacity by considering the perspective of a single contextual state, and then extrapolate to the full network by assuming each contextual state is equivalent.

We again consider that the network is currently in one of the memory states – specifically, pattern ν of context k. In an analogous way to Eqn. (5.3), we approximate the total input coming into a particular neuron i, denoted $h_i^{k\nu}$, as [Hertz et al., 1991]

$$\begin{aligned} h_i^{k\nu} &\approx 1 + \mathcal{N}\Big(0, \tfrac{1}{b_d} p/N_s\Big) + \mathcal{N}\Big(0, (s-1)p/N\Big) \\ &= 1 + \mathcal{N}(0, \alpha/b_d) + \mathcal{N}(0, \gamma). \end{aligned} \tag{5.5}$$

The two normally-distributed terms again represent crosstalk noise coming from the influence of other patterns on the network connectivity. The first is the noise from the other $p-1$ patterns of subnetwork k, but now also scales with $1/b_d$ – counterintuitively, this term will get larger as the sparsity of the network connectivity increases, mainly due to the fact that each individual pattern becomes less stable with increased sparsity [Sompolinsky, 1987]. The second term again represents

the noise coming from all of the $(s-1)p$ patterns in the other contexts, but now it is also larger because all neurons participate in all contexts. We estimated the memory capacity as a function of the original Hopfield capacity α_H, and arrived at a maximum capacity per context of

$$\alpha = \frac{b_d \alpha_H}{1 + (s-1)b_d}. \tag{5.6}$$

The factor b_d in the numerator suggests that even without multiple contexts, memory capacity degrades roughly linearly with network dilution [Sompolinsky, 1987, Hertz et al., 1991]. The mean-field results for the capacity per context for dendritic input selection are shown in Fig. 5.4A. We see that the capacity for each contextual configuration diminishes rapidly as the number of contextual states (s) increases, or sparsity (b_d) increases. In comparison with subnetwork modularity, this reduction is much steeper (compare with Fig. 5.2C). This may not be expected, as capacity should scale approximately linearly with both network size in the case of subnetworks, and with dilution of synapses (as already mentioned above). However, the amount of overlap between synapses in different contexts is much higher in the case of dendritic input selection (Fig. 5.11). To see this, note that the second term of the denominator of Eqn. (5.6) now scales linearly with b_d, instead of quadratically with b_s, leading to much more overlap between different contexts.

Because of this, little benefit is gained from contextual configurations that only depend on dendritic input gating. We see that the overall capacity remains upper bounded by standard Hopfield (Fig. 5.4B). Furthermore, adding this on top of subnetwork modularity does not add any benefit, and only decreases the maximally-obtained capacity (Fig. 5.4C-D). Numerical simulations are not shown here because it is very difficult to get an accurate estimate of capacity for the majority of parameter values with such small values of α (finite-size effects are too large).

5.2.3 Targeted dendritic input gating on top of modular Hopfield further enhances capacity

We have shown above that context-dependence can be implemented in an associative memory network through subnetwork modularity and through dendritic

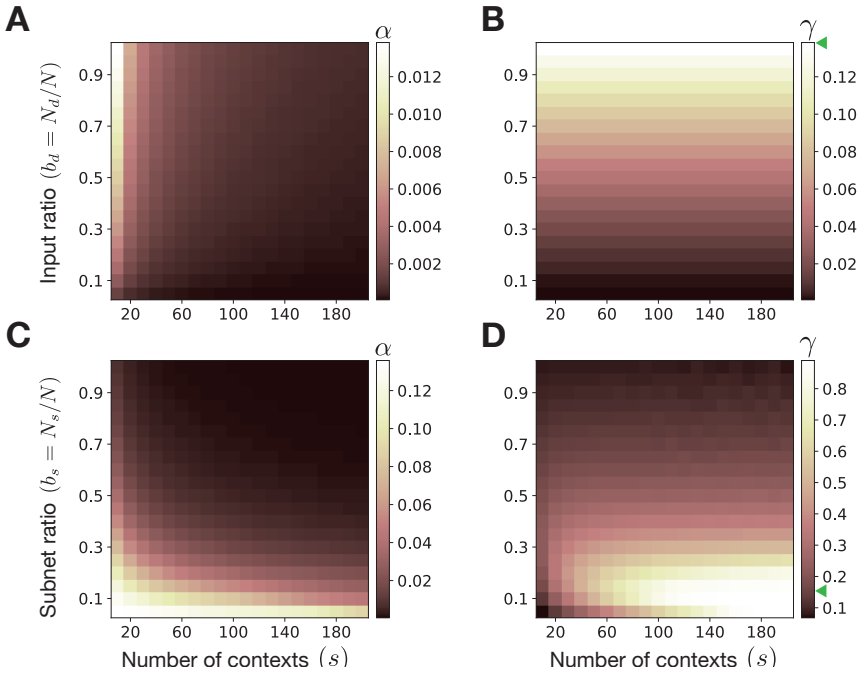


Figure 5.4: Memory capacity of the context-modular Hopfield network with random dendritic input selection (**A,B**) and with both subnetwork and input selection (**C,D**). **A,B**: Analytical capacity estimation for dendritic input selection for a single contextual configuration ($\alpha = p/N$; **A**) and for total network ($\gamma = sp/N = \alpha s$; **B**), plotted as a function of the number of subnetworks (s) and the relative input ratio ($b_d = N_d/N$). **C,D**: Analytical capacity estimation for combined subnetwork and dendritic input selection (shown for fixed $b_d = N_d/N_s = 0.5$) for a single contextual configuration ($\alpha = p/N_s$; **C**) and for total network ($\gamma = sp/N = \alpha s b_s$; **D**), plotted as a function of the number of subnetworks (s) and the ratio of subnetwork size ($b_s = N_s/N$).

input modularity. Subnetwork modularity acts to increase the memory capacity of the network, however, dendritic input modularity does not appear to have much benefit for storing memories. Thus, it does not seem prudent to combine the two types of modularity together in a random fashion (Fig. 5.4C,D).

We instead propose a more clever way of imposing dendritic input gating in concert with subnetwork selection that may have added benefits. To motivate this, we consider the context-modular Hopfield network with subnetwork selection, and compare the weight matrix considering a single subnetwork k with the final weight matrix across all contexts (Fig. 5.5, top and middle). As previously noted, the resulting weight matrix following subnetwork selection can be seen as a noisier version of the weight matrix for a single subnetwork. Looking at one weight, $w_{8,3}$,

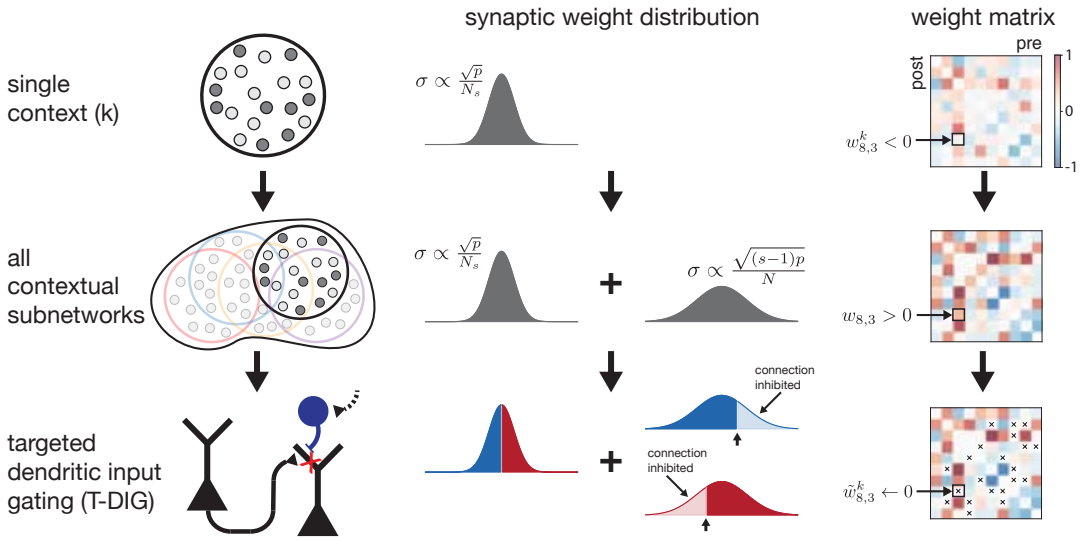


Figure 5.5: Schematic of targeted dendritic input gating (T-DIG). Comparison between standard Hopfield network (single context, k; top), multiple contextual subnetworks (middle), and targeted dendritic input gating (T-DIG; bottom). Memory patterns from the same context add noise to the weight matrix proportional to $\sqrt{p}/N$ (gray, middle left), and additional noise is added from other subnetworks proportional to $\sqrt{(s-1)p}/N$ (gray, middle right). Depending upon the sign of the resulting weight for one context (blue for positive, red for negative), the additional noise from other subnetworks may flip the sign of the weight. These weights that flip sign are inhibited by T-DIG. An example is shown on the right for a particular weight w^k_{83} which flips from negative to positive, and so is inhibited.

---

we see that it has a negative value for context k, indicating that on average, neurons 8 and 3 are anti-correlated for memory patterns in this one context. However, the final weight considering all contexts is positive, which means that averaging over all patterns of all contexts, neurons 8 and 3 are actually correlated. This means that this particular weight will have a net *harmful* effect on the majority of patterns in subnetwork k, and it would actually be better for recall performance if it were set to zero. Thus, we propose that dendritic input gating could be used very selectively to temporarily turn off harmful weights for particular contextual states (Fig. 5.5, bottom; see Methods for details). We refer to this as targeted dendritic input gating (T-DIG).

As shown in Fig. 5.6A,B, this method of temporarily gating inputs may result in a substantial amount of the connectivity being hidden in any given context, going up

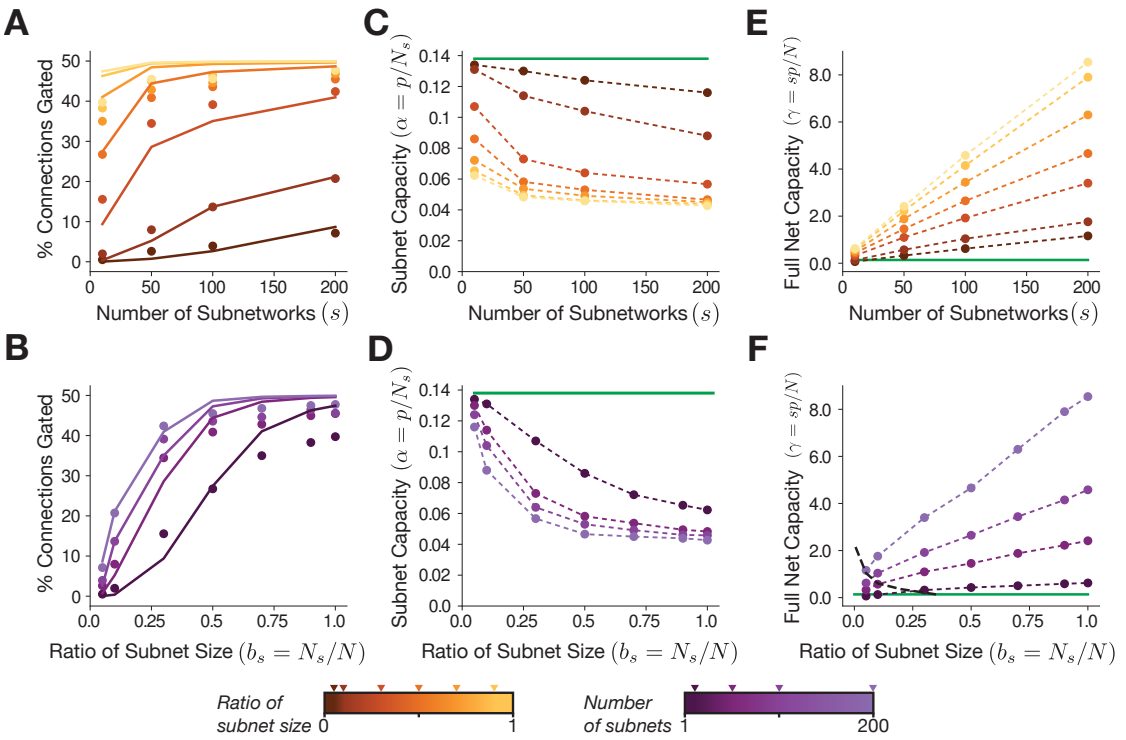


Figure 5.6: Targeted dendritic input gating (T-DIG) further enhances memory capacity. **A,B**: Percentage of connections (temporarily) inhibited due to T-DIG, plotted as a function of number of contexts over fixed subnetwork size (**A**), and as a function of subnetwork size over fixed number of contexts (**B**). **C-F**: Numerical capacity estimation for combined subnetwork and targeted input gating for a single contextual configuration ($\alpha = p/N_s$; **D,E**) and for total network ($\gamma = sp/N = \alpha s b_s$; **F,G**), plotted as a function of the number of subnetworks (s) and the ratio of subnetwork size ($b_s = N_s/N$). Comparison is made with standard Hopfield network (green), and low-activity Hopfield networks (dotted black, **F**).

to close to 50% sparsity in some cases. Counterintuitively, however, even though the network connectivity is getting more sparse, the capacity of each subnetwork remains high (Fig. 5.6C,D; compare with Fig. 5.2A-C), as the input gating is selectively removing noise from the connectivity matrix. The full network capacity is therefore also very high (Fig. 5.6E,F), now up to 60 times that of the standard Hopfield network, and much higher than the low-activity variant over similar activity levels (Fig. 5.6F, dotted line). In summary, with contextual state controlled by a detailed configuration of subnetworks and targeted dendritic inhibition, the network is able to store a large number of stable memory patterns.

5.2.4 Robustness to noise, overloading and spurious memories

We next examined several other characteristics of this context-modular Hopfield network, focusing on subnetwork selection with and without targeted dendritic input gating. We obtained an estimate of the basin of attraction sizes of the patterns in the network by looking at their corresponding *stability parameters* [Gardner, 1988, Abbott, 1990] over different parameter values for number of contexts (s) and relative subnetwork size (b_s). In general, a large positive average stability parameter value indicates that memories are stable and have large basins of attraction, and are thus robust to noise. As shown in Fig. 5.7A,B, the context-modular Hopfield network with subnetwork selection exhibits only a modest decrease in average basin of attraction size compared with the standard Hopfield model for most parameter ranges. Networks with T-DIG showed similar stability levels (not shown).

For low-activity Hopfield networks (dotted lines, Fig. 5.7A; black line, Fig. 5.7B), the average stability parameter decreases as the activity level decreases – patterns have fewer active neurons, and are thus more susceptible to noise. In comparison, we see that the average stability parameter for the context-modular network stays positive even for small b_s. Thus, context-modular Hopfield networks offer an increase in memory capacity with only minimal shrinking to basin of attraction size. We note that this result is based on the assumption other neurons in the network are being strongly suppressed, and thus are less susceptible to noise.

Due to the fact that contexts share neurons, it may be the case that some memories overlap enough to be attractors in other contextual configurations. In principle, the context-modular Hopfield network should have the same issues with spurious memories and mixture states as the standard Hopfield model, but the effects of overlapping subnetworks are less intuitive. To measure this, we calculated the average stability parameter across all memory patterns in the non-active context. In order for this architecture to work in the intended way, it is desirable that these other memories are not stable, and thus have low average stability parameters – ideally negative, or at least close to zero.

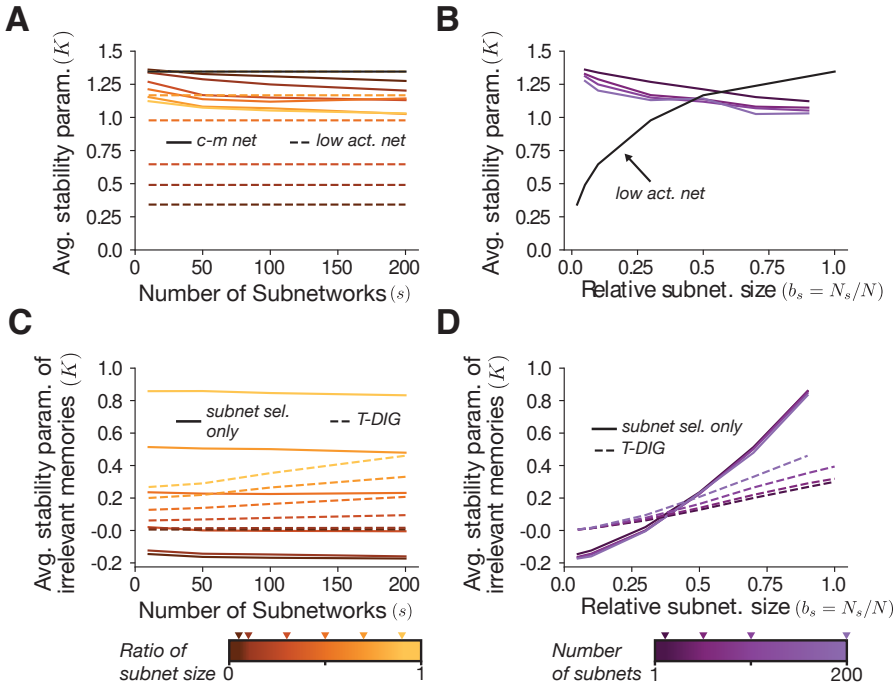


Figure 5.7: Stability of relevant and irrelevant memories. **A**: Robustness of memories as measured numerically by the average stability parameter (K; see methods) for context-modular Hopfield networks (c-m net) with different relative subnetwork size (b_s) as a function of number of subnetworks (s), compared with low-activity Hopfield networks (dotted lines, color indicates equivalent activity level $a = b_s/2$) **B**: Same as **A**, but plotted as a function of relative subnetwork size (b_s) for fixed number of subnetworks (s). **C,D**: Robustness of "irrelevant" memories (patterns from non-active contexts) as measured numerically by average stability parameter (K) for context-modular Hopfield networks with subnetwork selection only (full lines), and with subnetwork selection and T-DIG (dotted lines). Note, stability of relevant memories does not change with T-DIG, so are not shown in **A,B**.

As shown in Fig. 5.7C,D, the stability of "irrelevant" patterns took on a wide range of values depending upon the network parameters of number of contexts (s), relative subnetwork size (b_s), and whether or not targeted dendritic input gating was applied. Generally, irrelevant pattern stability depended heavily on the subnetwork size, growing to similar values as for the active memory patterns when relative subnetwork size is close to 1. Reassuringly, this stability dropped when targeted dendritic input selection was applied, but still remained well above zero for large b_s. Interestingly, for small relative subnetwork size (around 0.4 or less), subnetwork selection without targeted dendritic input selection suppresses irrelevant memories more effectively (Fig. 5.7D). We thus see that there is a tradeoff

between memory capacity and robustness, especially for T-DIG, as the area of parameter space with the largest memory capacity coincides with relatively high stability for memories that should be suppressed.

Next, we considered the robustness of the networks to memory overloading. It is a well-known fact that overloading traditional associative memory networks normally results in instabilities in the vast majority of memory patterns (so-called *blackout interference*; Amit [1992]). To explore the effects seen in context-modular Hopfield networks, we simulated various networks with different parameter ranges and different numbers of overloaded memories (p_x), from maximum capacity ($p_x/N = 0$), to increasing the number of memories by N ($p_x/N = 1$). Critically, we distributed the overloaded memories differently across contextual configurations, ranging from distributing them evenly in each context, to having only 10% of contexts store all of the extra memories. These results are shown in Fig. 5.8A,B for two example parameter sets ($b_s = 0.05, s = 50$ and $b_s = 0.3, s = 100$). When compared with the standard Hopfield and low-activity Hopfield, we see that the modular architecture of the context-modular network naturally makes it resistant to overloading in only some parts of the network. As can be seen in the two examples shown, the extent of this resistance to overloading depends upon the particular parameters chosen.

5.2.5 Dynamic gating of memory expression

Up to now, we have analyzed characteristics of context-dependent associative memory from the perspective of one contextual state. However, if such a system exists in the brain, then it must be able to switch between different contextual states when the need arises.

We thus propose that a separate network encodes context and interacts with the associative memory network (Fig. 5.9A). For the purposes of simplicity, we model this network with winner-take-all dynamics (only one unit can be active at a time), but it could in principle be implemented in another Hopfield-like associative memory network. Importantly, this network has reciprocal connections with the associative memory network – the active context unit provides inhibition to neurons

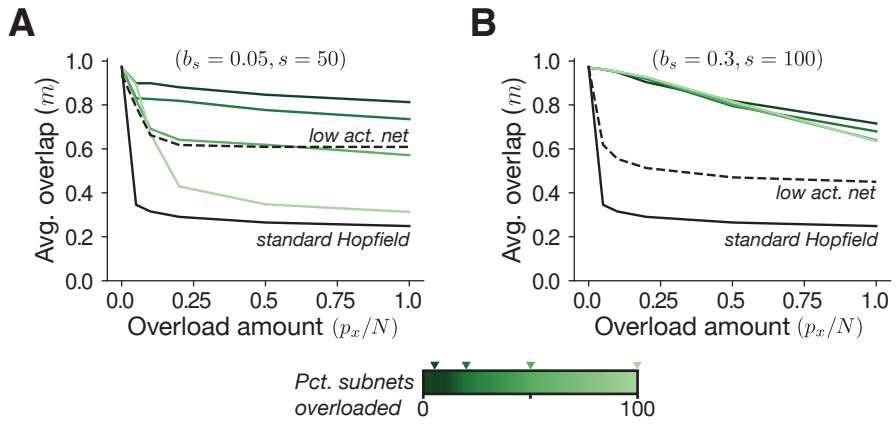


Figure 5.8: Robustness to memory overloading. Recall performance, as measured numerically by the average overlap across all patterns and contexts (m) as a function of overload amount (number of additional memories p_x after maximum capacity has been reached, divided by network size N. Overlap is measured by simulating the network dynamics using each memory pattern as the initial condition, and then measuring the relative number of neurons which remain in the original pattern state (see Methods 5.4.4 for more details). Overloaded memories are given to a particular subset of subnetworks (10%, 20%, 50%, and 100%; green colors). Shown for two particular parameter settings $b_s = 0.05, s = 50$ (**A**) and $b_s = 0.3, s = 100$ (**B**).

outside of its corresponding contextual subnetwork, as well as to dendritic inputs that should be inhibited as part of this contextual state (see methods for details). Equivalently, this unit could also provide a baseline excitation to units within its subnetwork. Correspondingly, this unit then receives excitation from units inside of the active subnetwork. This loop keeps the active context and memory stable, and prevents inputs to other areas of the network to be activated. However, given strong enough input, the context should change.

The functionality of this network is illustrated for an example simulation in Fig. 5.9B by looking at memory expression level for different contexts. Here, two memories from one context are activated, and readouts show the appropriate memory being activated. Then a memory from a different context is probed, but the context-encoding network suppresses the activity of this memory. Only when a stronger input is given does the context switch, and the corresponding memory is activated. Thus, this network architecture allows for dynamic memory gating, in which memory expression depends on the currently active memory. We show only a proof of concept here, and leave a more thorough analysis of dynamic memory for future work.

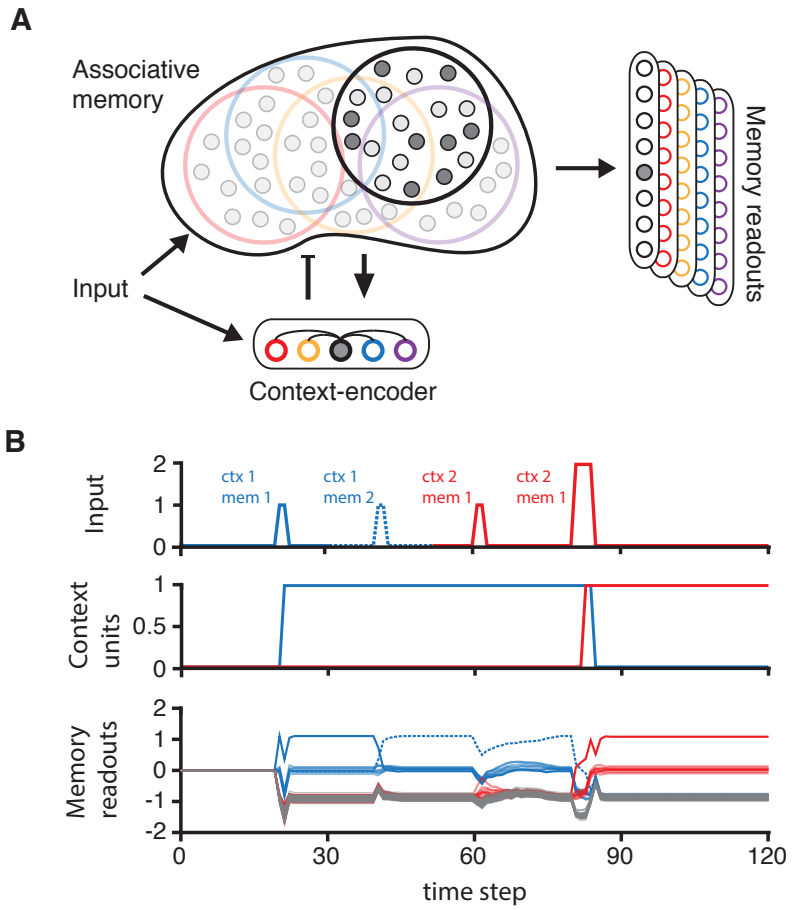


Figure 5.9: Dynamic gating of memory expression. **A**: Schematic of dynamic gating setup. A context-modular associative memory network is recurrently connected to a winner-take-all context-encoding network with one unit per context. Linear readouts measure memory activation, one per memory in each context. **B**: Example of dynamic memory gating. Top: four inputs are given to the network, probing two separate memories from context 1 (solid and dotted blue lines), followed by one memory from context 2, at two different amplitudes (red). Middle,bottom: Inputs from memories of context 1 (blue) cause the context to switch to 1, and the corresponding memory readout to activate (readouts for context 1 are blue). The input probe for the memory from context 2 causes no change at first, and requires stronger amplitude to cause the network to switch context and for the memory readout to change (*red*). Memory readouts for other contexts are shown in gray. Simulation parameters: $N = 1000$, $N_s = 200$, $s = 10$, $p = 10$.

5.3 Discussion

In this study, we presented a new model of context-dependent associative memory, in which contexts define particular network configurations of active neurons and dendritic inputs. Using analytical and numerical methods, we showed that these networks can have substantially improved memory capacity and robustness to

memory overloading, while retaining memory stability. Furthermore, we propose that this associative memory network can interact with a context-encoding network in order to dynamically select the active context, and switch between different contexts depending upon the situation.

This study relies on reduced interference between memories due to *sparsity* in activation and in shared representations between contexts. This phenomenon has been recognized in previous studies on associative memory [Tsodyks and Feigelman, 1988, Treves and Rolls, 1991, Okada, 1996], as well as in models of classical conditioning [Gershman et al., 2017]. Furthermore, it has also been used in machine learning as a potential mechanism for continual learning, in which training occurs in non-stationary environments [Pape et al., 2011, Masse et al., 2018].

Several previous studies have proposed modular architectures for associative memory similar to what we present here [Mari and Treves, 1998, Renart et al., 1999, Dubreuil and Brunel, 2016]. However, most of these studies use modularity to mimic the architecture of cortex – dense local connectivity with sparse long-ranged connectivity – and each module can have activity concurrently with others. The context-modular network presented here instead supposes that this modularity can exist within a single region, and that only one module is active at a time. In fact, the model is more similar to some previous proposals of hierarchical memory [Gutfreund, 1988, Perez-Vicente, 1989, Tsodyks, 1990], in which memories with common features are stored in a common representation. Here, context would represent the first step down the hierarchy, and the explicit memory is at the second level. In principle, more levels can be added to this hierarchy for more refined memory search, though we do not explore this here.

Subnetwork selection was chosen randomly in this study – each neuron participated in a given context with probability b_s. In contrast, dendritic input selection was shown to be substantially improved by targeting particular inputs that have a net harmful effect on memory recall. In a similar way, the subnetwork selection procedure could also be optimized for better performance. For example, selecting neurons based on how many other contexts they participate in should benefit performance.

Moreover, in reality, different contexts may have a different number of corresponding memories, which should affect the size of the available neurons coding for them.

Furthermore, while memories in this study had a one-to-one correspondence with context, we realize that memories are likely shared across contexts, and that it may be the case for several contexts to be active at one time. For analytical simplicity, we did not consider this here, as it is out of the scope of this study. However, it represents an interesting direction for future work.

We suggest that detailed top-down feedback and inhibitory control is responsible for creating the contextual network configurations required to group memories by context. This notion is in line with previous assertions that neurons are recruited to become part of cell assemblies based on relative excitability and local inhibitory microcircuits [Holtmaat and Caroni, 2016]. Recent experimental studies provide evidence for specific inhibitory cell motifs which could be responsible for such control [Letzkus et al., 2015, Kuchibhotla et al., 2017]. Additionally, contextual subnetwork selection may be implemented not only through inhibition, but also through *selective excitation* of a subnetwork of neurons. There is evidence that baseline shifts may dynamically control memory availability [Norman et al., 2017], which plays into this idea.

We present the context-modular Hopfield model as a generic associative memory network without explicit mention to brain areas and circuits. However, we speculate that this architecture may relate to two aspects of the brain. First of all, evidence suggests that contextual signals interact with cortical memories through the hippocampus and prefrontal cortex [Eichenbaum et al., 1999, Smith and Mizumori, 2006, Eichenbaum and Cohen, 2014, Eichenbaum, 2017]. It is thus possible that this model may inform our understanding of the circuit mechanisms behind these interactions. Second of all, the modularization presented here acts as a natural candidate mechanism for *pattern separation* – the phenomenon by which similar input patterns are stored in distinct neural representations – which is commonly attributed to the dentate gyrus in the hippocampus [Leutgeb et al., 2007]. Thus

we also suggest that this may relate to hippocampal mechanisms of separating memories in CA3 by context.

Finally, and perhaps most importantly, we do not propose any biologically-plausible learning mechanisms for which such an architecture can be constructed in the brain. However, the "Hebbian" learning rule used here suggests that such memories can be learned using local learning rules. As for the contextual grouping of different memories, we speculate that context changes slowly in an environment, and so the temporal proximity of memories can be used to group them into a common contextual architecture. Furthermore, top-down contextual signals through, e.g., neuromodulation, may act to control network state [Shulz et al., 2000]. This is also area for future work.

Overall, this work brings together known concepts about context-dependence of memory from psychology with a classic model of associative memory from neuroscience. It illustrates potential benefits of structuring memory around context, and has several exciting future directions.

5.4 Methods

5.4.1 Model formulation

The context-modular Hopfield network model is a fully connected recurrent network of N binary neurons denoted V_i for neuron i, and taking values of $\{0, 1\}$. The modular architecture defines s contextual states, each with a corresponding set of active neurons $N_s \leq N$, chosen unformly at random. We use $\mathcal{S}_k$ to denote the set of active neurons in contextual state k. We also define $b_s = N_s/N$ as the ratio of subnetwork to full network size, which is also the probability that each unit takes part in any given context.

Furthermore, each contextual state also determines a particular set of active inputs per neuron, chosen randomly with probability $b_d = N_d/N_s$, such that each neuron receives $N_d \leq N_s$ inputs on average. We define a symmetric matrix C^k with elements $c^k_{ij} = c^k_{ji} = 1$ if connections ij and ji are present in contextual state k, i.e.,

$\Pr(c_{ij}^k = 1) = b_d$ and $\Pr(c_{ij}^k = 0) = (1 - b_d)$. Given the symmetry of the standard Hopfield model, we only consider input configurations which are also symmetric.

Importantly, at any given time, only one contextual state is "active". Considering contextual state k is active, the dynamics of each unit in the network are defined as

$$V_i = H(h_i) = \begin{cases} H\left(\sum_j c_{ij}^k w_{ij} V_j - \theta_i^k\right) & \text{if } i \in \mathcal{S}_k, \\ 0 & \text{otherwise,} \end{cases} \tag{5.7}$$

where $H()$ is the heaviside step function, h_i is the total input to unit i, w_{ij} is the synaptic weight between neurons i and j (symmetric), and θ_i^k is the threshold for neuron i when contextual state k is active (defined below).

The network stores p memory patterns per contextual state, making $P = sp$ total memories. Memory patterns are denoted $\eta_i^{\sigma\mu}$ as the configuration of neuron i for memory μ of state σ, taking values $\{0, 1\}$ with equal probability provided unit i is in $\mathcal{S}_\sigma$, and 0 otherwise. The connectivity matrix is defined using a variant of a "Hebbian" learning rule [Hopfield, 1982, Tsodyks and Feigelman, 1988]

$$w_{ij} = \frac{8}{b_d N_s} \sum_{\sigma=1}^{s} \sum_{\mu=1}^{p} c_{ij}^\sigma \tilde{\eta}_i^{\sigma\mu} \tilde{\eta}_j^{\sigma\mu}, \tag{5.8}$$

where

$$\tilde{\eta}_i^{\sigma\mu} = \begin{cases} \eta_i^{\sigma\mu} - \frac{1}{2} & \text{if } i \in \mathcal{S}_\sigma, \\ 0 & \text{otherwise.} \end{cases} \tag{5.9}$$

Note that the factor $1/b_d$ ensures that the average total synaptic input to each neuron is around unity regardless of the choice of b_d [Sompolinsky, 1987]. The threshold is defined as

$$\theta_i^k = \frac{1}{2} \sum_{j \in \mathcal{S}_k} c_{ij}^k w_{ij}. \tag{5.10}$$

This choice of threshold, along with the factor 8 outside of the sum, ensures that this model is equivalent to the standard Hopfield model with $\{\pm 1\}$ units (see supplementary materials for further elaboration). Note that this threshold is different for each neuron and also changes with the contextual state of the network, whereas the connectivity matrix remains constant, with some weights being effectively set to zero through the c_{ij}^k terms in Eqn. (5.7).

The activity level, or coding level, of the patterns denotes the fraction of active neurons for any given pattern. We define two measures of activity level – per individual context, and for the network as a whole. The activity level per individual context, denoted a_s, was set to $\frac{1}{2}$, since each pattern unit was chosen to be 0 or 1 with uniform probability. The activity level as defined from the perspective of the entire network was $a = a_s b_s$, where b_s is the relative subnetwork size. We note that for the sake of analysis, we impose that all subnetworks contain the same number of active neurons and store the same number of memory patterns, though this need not be true in practice.

5.4.2 Comparison with standard Hopfield network and low-activity network

We compare the results obtained in this study with that of the standard Hopfield model [Hopfield, 1982, Amit et al., 1985], as well as the low-activity model [Tsodyks and Feigelman, 1988]. For these models, we consider a fully connected network of N neurons which store a set of P memory patterns. Again, units in these networks are binary and take on values $\{0, 1\}$ (see Supplementary Methods for a comparison of 01 and ± 1 formulations). Patterns are denoted $\eta_i^\mu \in \{0, 1\}$ with $\Pr(\eta_i^\mu = 1) = a$ and $\Pr(\eta_i^\mu = 0) = (1 - a)$, where a is the coding level.

Dynamics of this model are as follows

$$V_i = H\Big(\sum_{j=1}^{N} w_{ij} V_j - \theta_i - \theta_0\Big), \tag{5.11}$$

where w_{ij} is the connectivity matrix defined using the "Hebbian" rule [Tsodyks and Feigelman, 1988]

$$w_{ij} = \frac{1}{N} \sum_{\mu=1}^{N} (\eta_i^\mu - a)(\eta_j^\mu - a), \tag{5.12}$$

θ_i is a neuron-specific threshold, defined as

$$\theta_i = a \sum_j w_{ij}, \tag{5.13}$$

and θ_0 is a constant threshold defined as

$$\theta_0 = a(1-a)^2 - a^2(1-a). \tag{5.14}$$

The standard Hopfield model is obtained by setting $a = \frac{1}{2}$, which then makes $\theta_0 = 0$ (see supplementary methods for more details on the equivalence with standard ± 1 network).

Several previous works have studied the theoretical memory capacity limits of these networks [Amit et al., 1985, Tsodyks and Feigelman, 1988]. For the standard Hopfield model with a Hebbian learning rule, this is approximately $\alpha_H = P/N \approx 0.138$. For the low-activity model, capacity scales with the activity level as:

$$\alpha_{TF}(a) = \frac{1}{2a|\ln a|}, \tag{5.15}$$

for $a \ll 1$. The comparison with the context-modular Hopfield network is only relevant at intermediate activity levels, where this estimate does not hold. We thus use numerical simulations to obtain a more accurate estimate of memory capacity (see Fig. 5.10).

5.4.3 Analytical capacity estimation

In the following, we adapt a signal-to-noise analysis and a heuristic mean-field theory of memory capacity for the standard Hopfield network [Peretto, 1988, Hertz et al., 1991, Amit, 1992] to the case of the context-modular Hopfield network.

Wald's equation

We will make use of a result in statistics known as Wald's equation [Ross, 2014], which we summarize here. Consider a Binomial random variable $K \sim \mathcal{B}(N_K, p_K)$. Let Y be the sum of a sequence of independent identically distributed random variables X_i of length K, i.e., $Y = \sum_{i=1}^{K} X_i$. Assuming that each X_i is independent of K, then the mean and variance of Y can then be written as

$$\mathbb{E}[Y] = \mathbb{E}[K]\mathbb{E}[X] \tag{5.16}$$

and

$$\mathrm{Var}[X] = \mathbb{E}[K]\mathrm{Var}[X] + (\mathbb{E}[X])^2\mathrm{Var}[K]. \tag{5.17}$$

Signal-to-noise analysis

We are interested in calculating the stability of an arbitrary neuron i with respect to a particular pattern ν of context k, $\eta_i^{k\nu}$, which generalizes to $H(h_i^{k\nu}) = \eta_i^{k\nu}$, where $h_i^{k\nu}$ is the total input to neuron i when the network's state is exactly at pattern ν of context k. Plugging in Eqn. (5.8) into Eqn. (5.7), we obtain

$$h_i^{k\nu} = \sum_{j=1}^{N} c_{ij}^k w_{ij} V_j - \theta_i^k = \frac{8}{b_d N_s} \sum_{j=1}^{N} c_{ij}^k \Big[\sum_{\sigma} \sum_{\mu} c_{ij}^{\sigma} \tilde{\eta}_i^{\sigma\mu} \tilde{\eta}_j^{\sigma\mu} \Big] \tilde{\eta}_j^{k\nu} \tag{5.18}$$

$$= 2\tilde{\eta}_i^{k\nu} + \frac{8}{b_d N_s} \sum_{j=1}^{N} \sum_{\mu \neq \nu} c_{ij}^k \tilde{\eta}_i^{k\mu} \tilde{\eta}_j^{k\mu} \tilde{\eta}_j^{k\nu} + \frac{8}{b_d N_s} \sum_{j=1}^{N} \sum_{\sigma \neq k} \sum_{\mu=1}^{N} c_{ij}^k c_{ij}^{\sigma} \tilde{\eta}_i^{\sigma\mu} \tilde{\eta}_j^{\sigma\mu} \tilde{\eta}_j^{k\nu} \tag{5.19}$$

The right two terms in Eqn. (5.19) are two different "crosstalk" terms that may disrupt the stability of memories. The first one is analogous to the standard Hopfield crosstalk term [Hertz et al., 1991], and accounts for other patterns stored in the *same* context. The second term accounts for patterns stored in *different* contexts. We expect patterns stored in the same versus different subnetworks to affect stability differently, which will become apparent in the next section. As shown in Eqn. (5.3) and Eqn. (5.5), if the desired state of neuron i is 1 ($2\tilde{\eta}_i^{k\nu} = 1$), then we see that this state will be stable providing that the sum of the two crosstalk terms are greater than -1.

The input described in Eqn. (5.19) depends upon the actual value of $\tilde{\eta}_i^{k\nu}$, which complicates the analysis. In order to generalize to the case in which the desired state of the neuron can be either 0 or 1, we multiply Eqn. (5.19) by $-2\tilde{\eta}_i^{k\nu}$, turning the first term into -1. We now study the effect of the two crosstalk terms by considering the quantity

$$C_i^{k\nu} \triangleq -\frac{16\tilde{\eta}_i^{k\nu}}{b_d N_s} \sum_{j=1}^{N} \sum_{\mu \neq \nu} c_{ij}^k \tilde{\eta}_i^{k\mu} \tilde{\eta}_j^{k\mu} \tilde{\eta}_j^{k\nu} - \frac{16\tilde{\eta}_i^{k\nu}}{b_d N_s} \sum_{j=1}^{N} \sum_{\sigma \neq k} \sum_{\mu=1}^{N} c_{ij}^k c_{ij}^{\sigma} \tilde{\eta}_i^{\sigma\mu} \tilde{\eta}_j^{\sigma\mu} \tilde{\eta}_j^{k\nu} \tag{5.20}$$

which is the two crosstalk terms from Eqn. (5.19) multiplied by $-2\tilde{\eta}_i^{k\nu}$. We can now define the probability that any given bit will flip as

$$P_{\text{err}} = \Pr(C_i^{k\nu} > 1). \tag{5.21}$$

The aim is now to approximate the two terms in Eqn. (5.20) with Gaussian random variables.

Let's start with the first term. We first notice that $c_{ij}^k \in \{0,1\}$ will set some terms in the sum to zero. The four pattern variables, $\tilde{\eta}_i^{k\nu}\tilde{\eta}_i^{k\mu}\tilde{\eta}_j^{k\mu}\tilde{\eta}_j^{k\nu}$, combined with the total factor of 16 outside of the sum, act as a shifted and scaled Bernoulli random variable that takes on values ± 1 with equal probability. This sum can be interpreted as a random sum of random variables of the form $Y = \frac{1}{b_d N_s}\sum_{i=1}^{K}(2X_i - 1)$, with $K \sim \mathcal{B}(N_s(p-1), b_d)$ and $X_i \sim \text{Bern}(\frac{1}{2})$. Since this is a sum of Bernoulli random variables, we know that it will take the form of a Binomial distribution. We can then use Wald's equation to determine the statistics of this distribution. Given the symmetry of the Bernoulli random variables (± 1), it is easy to see that $\mathbb{E}[X] = 0$ and thus $\mathbb{E}[Y] = 0$. We use Eqn. (5.17) to estimate the variance of Y, first ignoring the factor $\frac{1}{b_d N_s}$, to obtain $\text{Var}[Y] = \mathbb{E}[K]\text{Var}[(2X-1)] = N_s(p-1)b_d$. Dividing by $b_d N_s$, we obtain $\text{Var}[Y] = \frac{p}{b_d N_s} = \frac{\alpha}{b_d}$.

The second term can now be approximated analogously to the first. It again takes the form $Z = \frac{1}{b_d N_s}\sum_{i=1}^{K}(2X_i - 1)$, but now with $K \sim \mathcal{B}(N_s(s-1)p, b_d^2 b_s^2)$ and $X_i \sim \text{Bern}(\frac{1}{2})$. The reason for the form of K in this term is that now we have both c_{ij}^k and c_{ij}^σ, independent and each with non-zero probability b_d, as well as $\tilde{\eta}_i^{\sigma\mu}$ and $\tilde{\eta}_i^{\sigma\mu}$, each with non-zero probability b_s (see Eqn. (5.9)). The mean of the resulting distribution is again zero, but now the variance is $\text{Var}[Z] = \frac{(s-1)p}{N} b_s = \gamma b_s$.

Since the two Binomial distributions Y and Z are symmetric and feature a large number of trials, they are well approximated by Gaussian distributions. Plugging these two approximations into Eqn. (5.20) gives us

$$
\begin{aligned}
C_i^{k\nu} &\approx \mathcal{N}\left(0, \frac{p}{b_d N_s}\right) + \mathcal{N}\left(0, \frac{(s-1)p}{N} b_s\right) \\
&= \mathcal{N}\left(0, \frac{\alpha}{b_d}\right) + \mathcal{N}\left(0, \gamma b_s\right) \qquad (5.22) \\
&= \mathcal{N}\left(0, \frac{\alpha}{b_d} + \gamma b_s\right) \qquad (5.23)
\end{aligned}
$$

We can now obtain a rough estimate of the memory capacity by ensuring that P_{err} remains low [Hertz et al., 1991]. Alternatively, we can also fomulate the memory capacity in terms of the standard Hopfield model, in which the crosstalk takes

the form of a single Gaussian centered around zero with variance $\alpha_H \approx 0.138$ (see Appendix). Setting the variance of Eqn. (5.23) equal to that of the standard Hopfield model, we arrive at the relationship

$$\alpha = \frac{\alpha_H}{\frac{1}{b_d} + (s-1)b_s^2}. \tag{5.24}$$

This equation applies generally for cases of arbitrary numbers of contexts s, relative subnetwork size b_s and relative input size b_d. To obtain the expression for context modulation by subnetwork selection only, we set $b_d = 1$, and for context modulation by dendritic-input selection only, we set $b_s = 1$. Finally, to obtain an expression for the full network capacity γ, we need to multiply Eqn. (5.24) by sb_s.

Information content

The information content of the standard Hopfield model can be calculated as the total entropy (average Shannon information) across all patterns in the following way [Gardner, 1988]. Considering that each neuron of each pattern is randomly chosen to be 0 or 1 with probability $\frac{1}{2}$, the entropy of each bit of each pattern is equal to the binary entropy function evaluated at $\frac{1}{2}$, which we will denote $H_b(\frac{1}{2})$. Multiplying this by the number of neurons in each pattern, and the number of patterns, we arrive at the total entropy across all patterns

$$NpH_b(\tfrac{1}{2}) = N^2\alpha_H. \tag{5.25}$$

This can be extended to the low-activity variant of the Hopfield network simply by replacing α_H in the equation above with the corresponding low-activity capacity for a particular activity level a: $\alpha_{TF}(a)$ (see section 5.4.2), and evaluating the binary entropy function at the activity level a. Together, this means that the information content of the low-activity Hopfield network for activity level a is

$$NpH_b(a) = N^2\alpha_{TF}(a)H_b(a). \tag{5.26}$$

For the context-modular network with subnetwork selection, we consider the following. For each subnetwork with N_s neurons, there are 2^{N_s} possible patterns, each equally likely, leading to an information content of

$$-\log_2(1/2^{N_s}) = N_s \log_2(2) \tag{5.27}$$

for each pattern. Multiplying this by the number of patterns, we obtain

$$spN_s \log_2(2) = spN_sN^2/N^2 = N^2\gamma b_s. \tag{5.28}$$

We can write this as a ratio with the information content of the standard Hopfield network to obtain

$$\frac{N^2\gamma b_s}{N^2\alpha_H} = \frac{\gamma b_s}{\alpha_H}. \tag{5.29}$$

Note that this equation for information content also holds for targeted dendritic input gating (T-DIG), with the only difference being that the memory capacity γ is different with and without T-DIG.

Mean-field theory

The memory capacity of the context-modular Hopfield network was calculated using established mean-field methods [Amit et al., 1985, Peretto, 1988, Geszti, 1990, Hertz et al., 1991]. See supplementary methods for details.

Targeted dendritic inhibition

Targeted dendritic inhibitory gating (T-DIG) was implemented algorithmically in the following way. Given a context-modular Hopfield network with parameters N, s, b_s, and p, we define the resulting weight matrix $\mathbf{W}$ from Eqn. (5.8). We also define the isolated weight matrix considering only patterns in a single subnetwork as $\tilde{\mathbf{W}}^k$ for each context k. The resulting weight matrix for each context after applying T-DIG is

$$\mathbf{W}^k = \mathbf{D}^k \odot \mathbf{W}, \tag{5.30}$$

where $\mathbf{D}^k$ is a binary matrix of zeros and ones (with elements d_{ij}^k), defined as

$$d_{ij}^k = \begin{cases} 0 & \text{if } (w_{ij}\tilde{w}_{ij}^k) < 0 \\ 1 & \text{otherwise} \end{cases} \tag{5.31}$$

and $\odot$ is the Hadamard product (element-wise product). This algorithm ensures that for each context k, the sign of each weight w_{ij}^k will reflect the correlation between neurons i and j over patterns in context k, but not over all other patterns.

This selective input gating substantially complicates the analytical methods for memory capacity estimation. Considering Eqn. (5.20), it is now the case that the two crosstalk terms are not independent anymore. We speculate that the analyses can be reformulated as multiplicative noise on the weights of a standard Hopfield network, but thus far have not been able to describe the capacity.

The expected amount of sparsity for each contextual configuration can be estimated assuming that the weight distributions for a single context and across all contexts are Gaussian with zero mean and variances proportional to the amount of crosstalk that they contribute. Given this, the probability that a particular weight will be removed can be approximated as:

$$\Pr(w_{ij} \leftarrow 0) = \int_0^\infty \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{x^2}{2\sigma_1^2}) \int_x^\infty \frac{1}{\sqrt{2\pi\sigma_A^2}} \exp(-\frac{y^2}{2\sigma_A^2}) dy dx \tag{5.32}$$

$$= \frac{1}{2\pi} \arctan(\sigma_A/\sigma_1), \tag{5.33}$$

where σ_1^2 is the variance of the weight distribution for a single context, and σ_A^2 is the variance of the weight distribution across all contexts. Based on the mean field results described below (see Eqn. (5.91)), we can approximate σ_1^2 and σ_A^2 as αr and $\frac{1}{2}\gamma r_n(b_s + \gamma + b_s^2)$, respectively. We compare this estimate with numerical simulations in Fig. 5.5B,C.

5.4.4 Numerical simulations

We briefly describe the details of all numerical simulations here. More information can be found in the supplementary methods. Code was written in C++ with the help of *Armadillo*, a linear algebra library [Sanderson and Curtin, 2016].

Memory capacity was estimated numerically by building finite-sized networks ($N = 10000$) initialized with a set of random patterns, connectivity and thresholds as defined above in section 5.4.1. Dynamics were run *synchronously* according to Eqn. (5.7), i.e., all units were updated simultaneously. Synchronous dynamics

were chosen for efficiency reasons despite potential convergence issues. To test stability of the patterns, we initialized the network in each memory state and simulated the dynamics until they either reached a steady-state or they reached the maximum number of allowed time steps (100). We then calculated the overlap of the network state with the original pattern state as:

$$m^{\mu} = \frac{4}{N_s} \sum_i \tilde{\eta}_i^{\sigma\mu} (V_i - \tfrac{1}{2}), \tag{5.34}$$

where $m \in [-1, 1]$ ($m = 1$ when the state is sitting exactly at the pattern, and $m = 0$ when it is uncorrelated with the pattern). This was done for all patterns to obtain an average overlap $\bar{m} = N_s^{-1} \sum_\mu m^\mu$. A small amount of noise was allowed in the overlap, such that successful memory retrieval was said to occur as long as $\bar{m} \geq 0.97$ [Amit et al., 1987b].

Binomial test of proportions

Due to the finite network size, random choices of patterns may have non-zero correlations, which affects recall performance (weight correlations scale as $1/\sqrt{N}$). This may lead to networks which are able to store a larger or smaller number of patterns (relative to network size) stably compared to a network storing patterns with absolutely zero correlations. We thus ran several trials for each configuration and treated the average overlap from each trial as an estimate of the parameter p_m from a Binomial distribution. The average overlap was corrected to fall between 0 and 1, such that the estimate of p_m was formulated as: $\hat{p}_m = N_{tr}^{-1} \sum_n \frac{1}{2}(\bar{m}_n + 1)$, where N_{tr} is the total number of trials. To determine the maximum number of patterns, we used a Binomial test of proportions with test statistic

$$T = \frac{p_0 - \hat{p}_m}{\sqrt{p_0(1 - p_0)/N_{tr}}}, \tag{5.35}$$

where $p_0 = 0.97$ is the cutoff value. The p-value for this test was set to 10%, such that the overlap was considered too low if $T > 1.281$. This allowed us to obtain a more accurate estimate of the average overlap with fewer trials. For all simulations shown here, $N_{tr} = 10$.

Estimating capacity from a single subnetwork

To further speed up numerical simulations, we also took advantage of the fact that all contextual configurations in the context-modular Hopfield network should be identical in memory storage properties. We therefore constructed a single contextual configuration and then mimicked the effect of the other contexts by adding noise to the weight matrix in the following way

$$w_{ij} = \frac{1}{N_s} \sum_{\mu=1}^{N_s} c_{ij} \tilde{\eta}_i^\mu \tilde{\eta}_j^\mu + \tfrac{1}{N_s}(\delta - \tfrac{p}{2}), \tag{5.36}$$

with

$$\delta = \begin{cases} \mathcal{B}(p, 0.5) & \text{with } \Pr = b_d \\ 0 & \text{otherwise,} \end{cases} \tag{5.37}$$

where $\mathcal{B}(n_b, p_b)$ is a binomial distribution with parameters n_b and p_b.

We confirmed the equivalence of this method with the simulation of all contextual configurations, and the two methods closely correspond (Fig. 5.14). The simulation of a single configuration with noise shows a small improvement over the simulation of all configurations due to the fact that it ignores small correlations that appear in the effective noise added to each synaptic weight. This difference has a negligible effect on the results shown in this chapter.

5.4.5 Robustness to noise

Robustness to noise was measured using so-called *stability parameters* [Gardner, 1988, Abbott and Kepler, 1990]. We adapt this notion here, and write the stability parameter of a particular pattern ν of a context k, for neuron i as

$$K_i^{k\nu} = \frac{2h_i^{k\nu}\tilde{\eta}_i^{k\nu}}{|W|_i}, \tag{5.38}$$

where

$$|W|_i = \left(\sum_{j=1}^{N} W_{ij}^2 \right)^{1/2} \tag{5.39}$$

We average this across all neurons and patterns to obtain the average stability parameter K.

5.4.6 Dynamic gating model

The dynamic memory gating model (Fig. 5.9) featured an associative memory network with $N = 1000$ units (again denoted V_i). The network was composed of $s = 10$ contexts, each defined by a subnetwork of $N_s = 200$ neurons, storing $p = 10$ random patterns each. Note that we do not consider sparse connectivity (and so $b_d = 1$), and targeted dendritic input gating eas not applied to the network.

The memory network is connected to a second network of s context-encoding units, denoted c_k for unit k, corresponding to context k. Units in both networks are binary, taking on values of 0 and 1. Finally, we found it necessary to add a third component, which was a global inhibitory neuron, denoted by y, which provides inhibition proportional to the population activity in the memory network (similar to Brunel [2016]). We also defined a linear readout for each memory of each context, denoted $z^{k\nu}$ for the readout for pattern ν of context k.

The connectivity matrix for the memory network, denoted w_{ij}^{MM} (for units i and j; M for memory), was defined according to the Hebbian rule in Eqn. (5.8).

Connectivity from the memory network to the context-encoding network was defined as:

$$w_{kj}^{CM} = \begin{cases} 1.5/N_s & \text{if } j \in \mathcal{S}_k, \\ 0 & \text{otherwise,} \end{cases} \tag{5.40}$$

which means that memory units that belong to context k will excite context unit k. The connectivity from the context-encoding network back to the memory network was defined as

$$w_{ik}^{MC} = \begin{cases} 4\max\limits_{\nu}\left(\sum_j w_{ij}^{MM} \eta_j^{k\nu} \right) & \text{if } i \notin \mathcal{S}_k, \\ 0 & \text{otherwise.} \end{cases} \tag{5.41}$$

Context units will inhibit memory neurons outside of their corresponding subnetwork proportional to the maximum activation that these memory units receive across all patterns in that context (the factor of 4 was added for stability purposes). This ensures that these neurons remain in the suppressed state.

The recurrent connectivity of the context-encoding network was implemented as

$$w_{kl}^{CC} = \begin{cases} 0.4 & \text{if } k = l, \\ -\max_{\nu} \sum_{i=1}^{N} w_{ki}^{CM} \eta_i^{l\nu} & \text{otherwise,} \end{cases} \tag{5.42}$$

which means that a particular context unit l will inhibit other context units proportional to the maximum input they receive across all patterns in context l. This ensures that the dynamics in the context-encoding network are approximately winner-take-all. The global inhibitory neuron, y, has a synaptic weight of $w_i^{MG} = 0.05$ for all memory units i.

Finally, output connectivity was defined as

$$w_{k\nu i}^{OM} = \eta_i^{k\nu} \tag{5.43}$$

Dynamics for the memory network were defined as:

$$V_i(t+1) = H\Big(\sum_{j=1}^{N} w_{ij}^{MM} V_j(t) - \sum_{j=1}^{s} w_{ij}^{MC} c_j(t) - w_{ij}^{MG} y(t) + I_i(t)\Big), \tag{5.44}$$

where $H(\cdot)$ denotes the heaviside step function, $I_i(t)$ is the time-dependent input to unit i, and t denotes the time step. For the context-encoding network, dynamics were implemented as:

$$c_i(t+1) = H\Big(\sum_{j=1}^{s} w_{ij}^{CC} c_j(t) + \sum_{j=1}^{N} w_{ij}^{CM} V_j(t) - \theta_c\Big), \tag{5.45}$$

where θ_c is a threshold set to 0.5. Finally, the global inhibitory neuron, y, was implemented as a linear neuron, defined as:

$$y(t+1) = \frac{1}{N_s} \sum_{j=1}^{N} V_j(t). \tag{5.46}$$

5.A Supplementary Methods

5.A.1 Equivalence of {-1,+1} and {0,1} formulations

The $\{0, 1\}$ formulation for the networks used in this study were designed such that they are equivalent to the standard $\{\pm 1\}$ formulation of the original Hopfield network [Hopfield, 1982]. To see this, it is sufficient to show that the dynamics

in either case are equivalent. We consider the context-modular Hopfield network as defined in section 5.4.1, but with units $S_i \in \{\pm 1\}$, and patterns $\xi_i^{\sigma\mu} \in \{\pm 1\}$ within the active context ($S_i = 0$ and $\xi_i^{\sigma\mu} = 0$ for any unit i not contained within the subnetwork). The synaptic weights are defined as

$$J_{ij} = \frac{1}{b_d N_s} \sum_{\sigma} \sum_{\mu} c_{ij}^k \xi_i^{\sigma\mu} \xi_j^{\sigma\mu} \tag{5.47}$$

and the dynamics of this network are described by the following equation (assuming context k is active)

$$S_i = \begin{cases} \text{sgn}(\sum_j c_{ij}^k J_{ij} S_j) & \text{if } i \in \mathcal{S}_k, \\ 0 & \text{otherwise.} \end{cases} \tag{5.48}$$

Now, starting from the dynamics of the $\{0, 1\}$ network, it is simple to show the equivalence to Eqn. (5.48) using the relationships between the two formulations: $2V_i - 1 = S_i$, $\tilde{\eta}^\mu = \frac{1}{2}\xi_i^\mu$, and $2H(x) - 1 = \text{sgn}(x)$ for all x. Assuming that context k is active, the dynamics of the $\{0, 1\}$ fomulation for a particular unit i within subnetwork k follow

$$\begin{aligned}
V_i &= H(\sum_j c_{ij}^k w_{ij} V_j - \theta_i^k), \\
2V_i - 1 &= 2H(\sum_j c_{ij}^k w_{ij} V_j - \tfrac{1}{2} \sum_j c_{ij}^k w_{ij}) - 1, \\
S_i &= \text{sgn}(\sum_j c_{ij}^k w_{ij} (V_j - \tfrac{1}{2})), \\
S_i &= \text{sgn}(\frac{8}{b_d N_s} \sum_j c_{ij}^k \sum_{\sigma} \sum_{\mu} \tilde{\eta}_i^{\sigma\mu} \tilde{\eta}_j^{\sigma\mu} (V_j - \tfrac{1}{2})), \\
S_i &= \text{sgn}(\frac{1}{b_d N_s} \sum_j c_{ij}^k \sum_{\sigma} \sum_{\mu} \xi_i^{\sigma\mu} \xi_j^{\sigma\mu} S_i), \\
S_i &= \text{sgn}(\sum_j c_{ij}^k J_{ij} S_j).
\end{aligned}$$

Thus, the deterministic dynamics of the two formulations are exactly the same. We note that this holds also for the stochastic version of the dynamics, in which $V_i = \sigma(h_i)$ and $S_i = \tanh(h_i)$, due to the identity $\tanh(x) = 2\sigma(x) - 1$. This is used in the following section.

5.A.2 Heuristic mean field analysis

In this section, we follow the heuristic mean-field analysis as described in [Peretto, 1988, Geszti, 1990, Hertz et al., 1991] and apply it to the context-modular Hopfield network. We derive this here for the case of the standard ± 1 Hopfield model, as defined in 5.A.1. Due to the equivalence with the $0, 1$ formulation, this result holds for both cases.

Incorporating context-dependence into the mean-field theory

In order to consider context-dependence through subnetwork selection and dendritic input selection, we must appropriately incorporate them into the mean-field theory. For subnetwork selection, as described in the main text, we can take the perspective of a single subnetwork k, and assume that the effect of the other contexts is just to add noise to the connectivity matrix relative to the subnetwork size $b_s = N_s/N$ and the number of contexts s. Thus, we do not need to do anything in addition to using the full weight matrix formulation as described in equation Eqn. (5.47), which we separate into two terms here since we are taking the perspective of context k:

$$J_{ij} = \frac{1}{b_d N_s} \sum_{\mu} c_{ij}^{k} \xi_i^{k\mu} \xi_j^{k\mu} + \frac{1}{b_d N_s} \sum_{\sigma \neq k} \sum_{\mu} c_{ij}^{\sigma} \xi_i^{k\mu} \xi_j^{k\mu} \tag{5.49}$$

As for dendritic input gating, this is a bit trickier. Essentially, this creates a so-called diluted Hopfield network, with connection sparsity controlled by the parameter $b_d = N_d/N$, the average number of inputs per neuron. It turns out that for a standard Hopfield network, symmetric dilution of synapses can be approximated by a fully-connected weight matrix with independent Gaussian noise added to each element with variance $N^{-1}\alpha(1-c)/c$, where N is the network size, $\alpha = p/N$ is the memory capacity, and c is the connection probability [Sompolinsky, 1987]. In this case, we can approximate the dilution in the first term in this way, to obtain

$$J_{ij} = \frac{1}{N_s} \sum_{\mu} \xi_i^{k\mu} \xi_j^{k\mu} + \frac{1}{b_d N_s} \sum_{\sigma \neq k} \sum_{\mu} c_{ij}^{\sigma} \xi_i^{k\mu} \xi_j^{k\mu} + \delta_{ij} \tag{5.50}$$

where δ_{ij} is the independent symmetric Gaussian noise with zero mean and variance $N_s^{-1}\alpha(1-b_d)/b_d$. The dilution in the second term should be treated differently. Since the second term only concerns patterns that we are not interested in recalling, the dilution does not add extra noise to the weights, but instead reduces the crosstalk effects. Thus we can treat this term as undiluted, but scaled by the amount of dilution b_d, since on average $b_d N$ connections will be present. Thus we are left with:

$$J_{ij} = \frac{1}{N_s}\sum_{\mu}\xi_i^{k\mu}\xi_j^{k\mu} + \frac{b_d}{N_s}\sum_{\sigma\neq k}\sum_{\mu}\xi_i^{k\mu}\xi_j^{k\mu} + \delta_{ij}, \tag{5.51}$$

Setup

We consider a stochastic version of the Hopfield network, with updates to the units S_i taking the form:

$$\Pr(S_i = \pm 1) = \sigma_\beta(\pm h_i) = \frac{1}{1+\exp(\mp 2\beta h_i)}, \tag{5.52}$$

where h_i is the input to neuron i, β is the inverse temperature parameter, which effectively controls the amount of noise in the updates, and σ_β is a variant of a sigmoid function, parameterized by β.

In mean field theory, we replace the true fluctuating input h_i by its average value $\langle h_i\rangle = \sum_j w_{ij}\langle S_j\rangle$, which allows us to compute the average activation of each neuron as

$$\langle S_i\rangle = \sum_{S_i=\{\pm 1\}} S_i \Pr(S_i) = \sigma(-2\beta h_i) - \sigma(2\beta h_i) = \tanh(\beta h_i) \tag{5.53}$$

We can then combine these two equations to get an expression for the average activation of each neuron as a function of the average activity of the rest of the network:

$$\langle S_i\rangle = \tanh(\beta\langle h_i\rangle) = \tanh\left(\beta\sum_j J_{ij}\langle S_j\rangle\right). \tag{5.54}$$

This allows us to describe the dynamics of the system using so-called order parameters. First of all, we use the two measures of capacity as defined in the main text, which are repeated here for convenience:

$$\begin{aligned} \alpha &= \frac{p}{N_s} && \text{(single context capacity)} \\ \gamma &= \frac{sp}{N} = \alpha s b_s && \text{(overall capacity)} \end{aligned} \tag{5.55}$$

where p is the number of patterns stored per context, N is the total network size, N_s is the number of active neurons per context, s is the number of contexts, and $b_s = N_s/N$ is the relative subnetwork size. Throughout the analysis, it is important to determine the nearness of the current network state to the different memory patterns, both in the current context, as well as across contexts. Consider that the currently active context is k. We use m_ν^k and n_μ^σ to denote the *overlap* between the average network state $\langle S_i \rangle$ and a particular pattern ν of context k ($\xi_i^{k\nu}$), or a pattern μ of context σ ($\xi_i^{\sigma\mu}$), respectively, when k is active:

$$m_\nu^k = \frac{1}{N_s} \sum_{i \in \mathcal{S}_k} \xi_i^{k\nu} \langle S_i \rangle \tag{5.56}$$

$$n_\mu^\sigma = \frac{1}{N_s} \sum_{i \in \mathcal{S}_k} \xi_i^{\sigma\mu} \langle S_i \rangle \tag{5.57}$$

where $\mathcal{S}_k$ is the set of active neurons in context k (as defined in 5.4.1). Note that both m_ν^k and n_μ^σ are sums of N_s terms and are normalized by $1/N_s$. However, the $\xi_i^{\sigma\mu}$ terms of Eqn. (5.57) may be zero depending upon the subnetwork size, so we expect Eqn. (5.56) and Eqn. (5.57) to have different variances. Importantly, we suppose that the network is close to one of the patterns, say pattern 1 of context k. Thus m_1^k will be of order unity, and the rest of m_ν^k's for $\nu \neq 1$ are small, of order $1/\sqrt{N_s}$. To see this, consider the fact that m_ν^k is a sum of N_s terms, each of which will be ± 1 with equal probability. This can be approximated by a zero-mean Gaussian with variance N_s. The normalizer $1/N_s$ makes the variance $1/N_s$, and thus the standard deviation $1/\sqrt{N_s}$. As for n_μ^σ, it is a sum of N_s terms, each of which will be non-zero with probability b_s, due to $\xi_i^{\sigma\mu}$. The non-zero terms will then be ± 1 with equal probability. This can be approximated by a Gaussian with variance N_s^2/N, which, divided by N_s, yields $1/N$. Thus the standard deviation is $1/\sqrt{N}$.

We next introduce r and r_n to denote the *mean square overlap* of the system configuration with the nonretrieved patterns in subnetwork k, and all other

subnetworks, respectively:

$$r = \frac{1}{\alpha} \sum_{\nu \neq 1} (m_\nu^k)^2 \tag{5.58}$$

$$r_n = \frac{1}{\gamma} \sum_{\sigma \neq k} \sum_{\mu} (n_\mu^\sigma)^2. \tag{5.59}$$

Both of these quantities should be approximately of order unity. For the case of r, we have a sum of p terms, each with variance $1/N_s$, and normalized by $1/\alpha = N_s/p$. For r_n, we have a sum of $(s-1)p$ terms, each with variance of approximately $1/N$, and normalized by $1/\gamma = N/sp$. *Our task is now to get a self-consistent calculation of* r, r_n, *and* m_1.

Derivation Part A: finding an expression for r

We begin by plugging the mean field equations for $\langle S_i \rangle$ from Eqn. (5.54) into equation Eqn. (5.56), and expanding the weight matrix using the expression in Eqn. (5.51):

$$\begin{aligned} m_\nu^k = & \frac{1}{N_s} \sum_i \xi_i^{k\nu} \tanh \left(\frac{\beta}{N_s} \sum_{j \in T_k} \sum_\mu \xi_i^{k\mu} \xi_j^{k\mu} \langle S_j \rangle \right. \\ & \left. + \frac{\beta b_d}{N_s} \sum_{j \in T_k} \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} \xi_j^{\sigma\mu} \langle S_j \rangle + \beta \sum_{j \in T_k} \delta_{ij} \langle S_j \rangle \right) \end{aligned} \tag{5.60}$$

We then rearrange terms and substitute the other m_μ^k and n_μ^σ terms into the equation:

$$\begin{aligned} m_\nu^k = & \frac{1}{N_s} \sum_i \xi_i^{k\nu} \tanh \left(\beta \sum_\mu \xi_i^{k\mu} \frac{1}{N_s} \sum_{j \in T_k} \xi_j^{k\mu} \langle S_j \rangle \right. \\ & \left. + \beta b_d \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} \frac{1}{N_s} \sum_{j \in T_k} \xi_j^{\sigma\mu} \langle S_j \rangle + \beta \sum_{j \in T_k} \delta_{ij} \langle S_j \rangle \right) \end{aligned} \tag{5.61}$$

$$= \frac{1}{N_s} \sum_i \xi_i^{k\nu} \tanh \left(\beta \sum_\mu \xi_i^{k\mu} m_\mu^k + \beta b_d \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} n_\mu^\sigma + \beta \sum_{j \in T_k} \delta_{ij} \langle S_j \rangle \right) \tag{5.62}$$

Next, we pull out the terms with m_1^k and m_ν^k, and then use a trick to multiply the equation by $\xi_i^{k1} \xi_i^{k1} = 1$. We can put one of these terms inside of the tanh

due to the fact that $\tanh(-x) = -\tanh(x)$:

$$m_\nu^k = \frac{1}{N_s}\sum_i \xi_i^{k\nu}\tanh\left(\beta\Big(\xi_i^{k1}m_1^k + \xi_i^{k\nu}m_\nu^k + \sum_{\mu\neq 1,\nu}\xi_i^{k\mu}m_\mu^k + b_d\sum_{\sigma\neq k}\sum_\mu \xi_i^{\sigma\mu}n_\mu^\sigma + \sum_{j\in T_k}\delta_{ij}\langle S_j\rangle\Big)\right) \tag{5.63}$$

$$= \frac{1}{N_s}\sum_i \xi_i^{k\nu}\xi_i^{k1}\tanh\left(\beta\Big(m_1^k + \xi_i^{k\nu}\xi_i^{k1}m_\nu^k + \sum_{\mu\neq 1,\nu}\xi_i^{k\mu}\xi_i^{k1}m_\mu^k + b_d\sum_{\sigma\neq k}\sum_\mu \xi_i^{\sigma\mu}\xi_i^{k1}n_\mu^\sigma + \sum_{j\in T_k}\xi_i^{k1}\delta_{ij}\langle S_j\rangle\Big)\right) \tag{5.64}$$

We then use another trick, and replace the right-hand side of Eqn. (5.64) with the first two terms of the Taylor expansion with respect to m_ν^k, i.e., $f(m_\nu^k) = f(a) + f'(a)(m_\nu^k - a)$, where $f(m_\nu^k)$ is the right-hand side of Eqn. (5.64). We assume that m_ν^k is small, of order $1/\sqrt{N}$, and so we take $a = 0$. This expansion yields:

$$m_\nu^k \approx \frac{1}{N_s}\sum_i \xi_i^{k\nu}\xi_i^{k1}\tanh\Big(\beta(m_1^k + c_i)\Big) + \beta m_\nu^k + \beta d m_\nu^k \tag{5.65}$$

where we use c_i to denote the crosstalk terms:

$$c_i = \sum_{\mu\neq 1}\xi_i^{k\mu}\xi_i^{k1}m_\mu^k + b_d\sum_{\sigma\neq k}\sum_\mu \xi_i^{\sigma\mu}\xi_i^{k1}n_\mu^\sigma + \sum_{j\in T_k}\xi_i^{k1}\delta_{ij}\langle S_j\rangle \tag{5.66}$$

and

$$d = \frac{1}{N_s}\sum_i \tanh^2\Big(\beta(m_1^k + c_i)\Big) \tag{5.67}$$

Note that we write the crosstalk terms c_i to include all patterns except for pattern 1 of context k ($k1$). Based on the expression in Eqn. (5.65), c_i should also exclude the term $\xi_i^{k\nu}\xi_i^{k1}m_\nu^k$, but this will be a negligible addition assuming large N and p. This term is left in for generality purposes, as the expression c_i will reappear later in the derivation. We now approximate d as an average of a function $f(z)$ of a Gaussian random variable z. Since N is large (and therefore N_s too), we can replace the average $\frac{1}{N_s}\sum_i \tanh^2$ with the integral over the distribution of z, which we will call q:

$$d \approx q = \int \frac{dz}{\sqrt{2\pi}\sigma}\exp\left(\frac{-(z-\bar{z})^2}{2\sigma^2}\right)f(z) \tag{5.68}$$

The mean and variance of this distribution will depend upon the architecture of the system, i.e., the choice of b_s, b_d, s, etc. There may also be several ways of

devising appropriate approximations for the terms inside. We thus remain agnostic to this for the moment, and write the mean and variance as $m_1^k + \mu_*$ and σ_*^2, respectively. We are thus left with:

$$m_\nu^k = \frac{1}{N_s} \sum_i \xi_i^{k\nu} \xi_i^{k1} \tanh\Big(\beta(m_1^k + c_i)\Big) + \beta m_\nu^k + \beta q m_\nu^k \tag{5.69}$$

where

$$q = \int \frac{dz}{\sqrt{2\pi}} \exp(-\frac{1}{2}z^2) \tanh^2(\beta(m_1 + \mu_* + \sigma_* z)). \tag{5.70}$$

Moving all terms with m_ν^k to the left, this can be rewritten as:

$$m_\nu^k = \frac{N_s^{-1} \sum_i \xi_i^{k\nu} \xi_i^{k1} \tanh\Big(\beta(m_1^k + c_i)\Big)}{1 - \beta(1-q)} \tag{5.71}$$

Finally, we can obtain an expression for r by squaring and averaging Eqn. (5.71) (since this is the definition of r, see Eqn. (5.58)). First, squaring Eqn. (5.71) yields

$$\begin{aligned}(m_\nu^k)^2 =& \Big(\frac{1}{1-\beta(1-q)}\Big)^2 \frac{1}{N_s^2} \sum_{ij} \xi_i^{k\nu} \xi_i^{k1} \xi_j^{k\nu} \xi_j^{k1} \tanh\Big(\beta(m_1^k + c_i)\Big) \\ &\times \tanh\Big(\beta(m_1^k + c_j)\Big)\end{aligned} \tag{5.72}$$

Then averaging, we get

$$\begin{aligned}\langle (m_\nu^k)^2 \rangle =& \Big(\frac{1}{1-\beta(1-q)}\Big)^2 \frac{1}{N_s^2} \sum_{ij} \xi_i^{k1} \xi_j^{k1} \langle \xi_i^{k\nu} \xi_j^{k\nu} \rangle \tanh\Big(\beta(m_1^k + c_i)\Big) \\ &\times \tanh\Big(\beta(m_1^k + c_j)\Big)\end{aligned} \tag{5.73}$$

We see that the only terms that are affected by the averaging are $\xi_i^{k\nu}$ and $\xi_j^{k\nu}$. It is easy to see that if $i \neq j$, these two terms are independent and so the expected value will be zero, which removes most terms from the sum. For $i = j$, both $\xi_i^{k\nu}\xi_j^{k\nu}$ and $\xi_i^{k1}\xi_j^{k1}$ will equal 1. This leaves us with

$$\langle (m_\nu^k)^2 \rangle = \Big(\frac{1}{1-\beta(1-q)}\Big)^2 \frac{1}{N_s} \sum_i \tanh^2\Big(\beta(m_1^k + c_i)\Big). \tag{5.74}$$

Lastly, we can replace the sum of $\tanh^2$ functions with the same q variable defined in Eqn. (5.70), to get an expression for r:

$$r = \langle (m_\nu^k)^2 \rangle = \frac{q}{[1-\beta(1-q)]^2}. \tag{5.75}$$

Note that this expression is equivalent to the expression for r in the standard Hopfield model, except that q has changed.

Derivation Part B: finding an expression for r_n

We again start by using the mean field equations Eqn. (5.54), but now plug them in to Eqn. (5.57) for a particular n_ν^ϕ. We again rearrange terms so as to plug m_μ^k and n_μ^σ into the equation:

$$\begin{aligned} n_\nu^\phi =& \frac{1}{N_s}\sum_i \xi_i^{\phi\nu} \tanh\left(\frac{\beta}{N_s}\sum_{j=1}^{N}\sum_\mu \xi_i^{k\mu}\xi_j^{k\mu}\langle S_j\rangle \right. \\ &\left. + \frac{\beta b_d}{N_s}\sum_{j\in T_k}\sum_{\sigma\neq k}\sum_\mu \xi_i^{\sigma\mu}\xi_j^{\sigma\mu}\langle S_j\rangle + \beta\sum_{j\in T_k}\delta_{ij}\langle S_j\rangle\right) \qquad (5.76) \\ =& \frac{1}{N_s}\sum_i \xi_i^{\phi\nu} \tanh\left(\beta\sum_\mu \xi_i^{k\mu}\frac{1}{N_s}\sum_{j\in T_k}\xi_j^{k\mu}\langle S_j\rangle \right. \\ &\left. + \beta b_d\sum_{\sigma\neq k}\sum_\mu \xi_i^{\sigma\mu}\frac{1}{N_s}\sum_{j\in T_k}\xi_j^{\sigma\mu}\langle S_j\rangle + \beta\sum_{j\in T_k}\delta_{ij}\langle S_j\rangle\right) \qquad (5.77) \\ =& \frac{1}{N_s}\sum_i \xi_i^{\phi\nu} \tanh\left(\beta\sum_\mu \xi_i^{k\mu}m_\mu^k + \beta b_d\sum_{\sigma\neq k}\sum_\mu \xi_i^{\sigma\mu}n_\mu^\sigma + \beta\sum_{j\in T_k}\delta_{ij}\langle S_j\rangle\right) \qquad (5.78) \end{aligned}$$

We then follow the same steps as the previous section, by pulling out the terms with m_1^k and n_ν^ϕ, multiplying the equation by $\xi_i^{k1}\xi_i^{k1} = 1$, and finally taking the Taylor expansion with respect to n_ν^ϕ around zero, to get:

$$n_\nu^\phi \approx \frac{1}{N_s}\sum_i \xi_i^{\phi\nu}\xi_i^{k1}\tanh\Big(\beta(m_1^k + c_i)\Big) + \beta n_\nu^\phi + \beta d n_\nu^\phi, \qquad (5.79)$$

where c_i and d are as defined above in Eqn. (5.66) and Eqn. (5.67), respectively. The reason for the approximate equality is because c_i in this case does not include pattern $\phi\nu$. Again, because we are considering the case of large N and p, this change is negligible. We then approximate d using the very same method as before in Eqn. (5.68) and Eqn. (5.69). We can thus approximate Eqn. (5.79) with the following:

$$\begin{aligned} n_\nu^\phi \approx& \frac{1}{N_s}\sum_i \xi_i^{\phi\nu}\xi_i^{k1}\tanh\Big(\beta(m_1^k + c_i)\Big) + \beta n_\nu^\phi + \beta q n_\nu^\phi \qquad (5.80) \\ =& \frac{N^{-1}\sum_i \xi_i^{\phi\nu}\xi_i^{k1}\tanh\Big(\beta(m_1^k + c_i)\Big)}{1-\beta(1-q)} \qquad (5.81) \end{aligned}$$

where q is defined as in Eqn. (5.70). Finally, following Eqn. (5.72) and Eqn. (5.73), we can obtain an expression for r_n by squaring and averaging Eqn. (5.81). This yields

$$\begin{aligned}\langle (n_\nu^\phi)^2 \rangle =& \Big(\frac{1}{1-\beta(1-q)}\Big)^2 \frac{1}{N_s^2} \sum_{ij} \xi_i^{k1}\xi_j^{k1}\langle \xi_i^{\phi\nu}\xi_j^{\phi\nu}\rangle \tanh\Big(\beta(m_1^k + c_i)\Big) \\ & \times \tanh\Big(\beta(m_1^k + c_j)\Big) \end{aligned} \tag{5.82}$$

Again, only the terms with $i = j$ will be non-zero (see description following Eqn. (5.73)). However, in this case the terms $\langle \xi_i^{\phi\nu}\xi_j^{\phi\nu}\rangle$ for $i = j$ do not equal 1, but equal $b_s = N_s/N$. This then leaves us with

$$\langle (n_\nu^\phi)^2 \rangle = \frac{b_s}{[1-\beta(1-q)]^2} \frac{1}{N_s} \sum_i \tanh^2\Big(\beta(m_1^k + c_i)\Big) \tag{5.83}$$

Lastly, we can replace the sum of $\tanh^2$ functions with the same q variable defined in Eqn. (5.72), to get an expression for r_n:

$$r_n = \langle (n_\nu^\phi)^2 \rangle = \frac{qb}{[1-\beta(1-q)]^2}. \tag{5.84}$$

Derivation Part C: finding an expression for m_1^k

We assume the network configuration is close to pattern 1 of subnetwork k, and so m_1^k should be much larger than the other overlaps. We follow the steps from Part A up to Eqn. (5.64), but now using m_1^k instead of an arbitrary m_ν^k to get:

$$\begin{aligned} m_1^k =& \frac{1}{N_s}\sum_i \tanh\Bigg(\beta\Big(m_1^k + \sum_{\mu\neq 1} \xi_i^{k\mu}\xi_i^{k1} m_\mu^k \\ & + b_d \sum_{\sigma\neq k}\sum_\mu \xi_i^{\sigma\mu}\xi_i^{k1} n_\mu^\sigma + \sum_{j\in T_k} \xi_i^{k1}\delta_{ij}\langle S_j\rangle\Big)\Bigg) \end{aligned} \tag{5.85}$$

$$= \frac{1}{N_s}\sum_i \tanh\Big(\beta(m_1^k + c_i)\Big) \tag{5.86}$$

Finally, we use the same trick as in Eqn. (5.68), and treat m_1^k as an average of a function $f(z)$ over the Gaussian random variable z:

$$m_1^k \approx \int \frac{dz}{\sqrt{2\pi}} \exp(-\frac{1}{2}z^2) \tanh[\beta(m_1^k + \mu_* + \sigma_* z)], \tag{5.87}$$

again remaining agnostic as to the exact form of the mean and variance of the distribution, which will be addressed in the next section.

Derivation Part D: approximating the distributions of q and m_1^k

We now focus on determining an appropriate approximating distribution for q and m_1^k. To do this, we will take a more in-depth look at the expression for c_i in Eqn. (5.66), which we repeat here for convenience:

$$c_i = \sum_{\mu \neq 1} \xi_i^{k\mu} \xi_i^{k1} m_\mu^k + b_d \sum_{\sigma \neq k} \sum_{\mu} \xi_i^{\sigma\mu} \xi_i^{k1} n_\mu^\sigma + \sum_{j \in T_k} \xi_i^{k1} \delta_{ij} \langle S_j \rangle \tag{5.88}$$

The first term of Eqn. (5.88) is a sum over the product of three independent random variables $\xi_i^{k\mu}$, ξ_i^{k1} and m_μ^k. Based on Eqn. (5.58), we know that m_μ^k should have zero mean and variance $\alpha r/p$. Due to the symmetry of m_ν^k around zero, the other two variables $\xi_i^{k\mu}\xi_i^{k1} = \pm 1$ will not have any effect. We thus have the sum of $p-1$ random variables, each with variance $\alpha r/p$, which leads to a single random variable with zero mean and variance αr.

Now we move on to the second term of Eqn. (5.88), which is again a sum over the product of three random variables. From Eqn. (5.59), we know that n_μ^σ should be zero mean with variance $\gamma r_n/(s-1)p$. The term ξ_i^{k1} again does not have an effect due to the symmetry of n_μ^σ. However, the term $\xi_i^{\sigma\mu}$ will be different – it will be nonzero with probability b_s, and ± 1 otherwise. If we assume that $\xi_i^{\sigma\mu}$ and n_μ^σ are independent, then we get a random sum of random variables of the form $b_d \sum_{i=1}^K X_i$, with $K \sim \mathcal{B}((s-1)p, b_s)$ and $X_i \sim \mathcal{N}(0, \gamma r_n/(s-1)p)$. Using Wald's equation (see section X), we arrive at a Gaussian distribution with zero mean and variance $\gamma r_n b_s b_d^2$.

However, it turns out that the assumption that $\xi_i^{\sigma\mu}$ and n_μ^σ are independent is not true. They are slightly correlated, which shifts the mean to γ (shown in Fig. 5.12A). Thus, as the total capacity increases, the mean-field approximation gets worse. In fact, we even face this issue with the first term, as m_μ^k and $\xi_i^{k\mu}$ are also correlated, which causes the mean to scale with α. However, since α never goes above 0.138, this fact can be safely ignored. For the second term, we can solve this by setting $\mu_* = \gamma$.

Adding a term to the mean turns out to complicate the solution quite a bit, so we propose an alternative method as well. This is based on two tricks. First of all, we can calculate the variance assuming zero mean by computing $\mathbb{E}[X^2] = \mathrm{Var}[X] + \mathbb{E}[X]^2$. Second of all, since each neuron has equal probability of being 1 and -1, we can

split up the contribution to the variance from the positive and negative components of the distribution, each having an effect half of the time. It turns out that we get the following expression for the variance:

$$\begin{aligned}\text{Var} &= \tfrac{1}{2}\mathbb{E}[X_+^2] + \tfrac{1}{2}\mathbb{E}[X_-^2] && (5.89)\\ &= \tfrac{1}{2}(\gamma b_s + \gamma^2) b_d^2 r_n + \tfrac{1}{2}\gamma b_s^2 b_d^2 r_n && (5.90)\\ &= \tfrac{1}{2}\gamma b_d^2 r_n (b_s + \gamma + b_s^2) && (5.91)\end{aligned}$$

This was confirmed empirically by simulating several networks over different realizations of b_s, b_d and s (Fig. 5.12B).

Finally, for the last term of Eqn. (5.88), it can be shown that this will inject Gaussian noise with variance $\Delta^2 q$, where $\Delta^2 = \alpha(1 - b_d)/b_d$ [Sompolinsky, 1987]. To see this, we first note that all three terms in this sum are independent random variables with zero mean, which means that the sum should also have zero mean. The term ξ_i^{k1}, being ± 1, will also have no effect due to the symmetry of the sum. Now, we can estimate the variance of this term, which we will call Z:

$$\begin{aligned}\text{Var}[Z] &= \mathbb{E}\left[(\sum_{j \in T_k} \delta_{ij} \langle S_j \rangle)^2 \right] && (5.92)\\ &= \mathbb{E}\left[\sum_{j,l \in T_k} \delta_{ij}\delta_{il} \langle S_j \rangle \langle S_l \rangle \right] && (5.93)\end{aligned}$$

Note that for $j \neq l$, the expected value of this expression is zero because each item is idependent and has zero mean. We are thus only left with terms for $j = l$:

$$\begin{aligned}\text{Var}[Z] &= \mathbb{E}\left[\sum_{j \in T_k} \delta_{ij}^2 \langle S_j \rangle^2 \right] && (5.94)\\ &= \mathbb{E}\left[\delta^2\right] \mathbb{E}\left[\sum_{j \in T_k} \langle S_j \rangle^2 \right] && (5.95)\\ &= \frac{\alpha(1 - b_d)}{b_d} \mathbb{E}\left[\frac{1}{N_s} \sum_{j \in T_k} \langle S_j \rangle^2 \right] && (5.96)\\ &= \Delta^2 q. && (5.97)\end{aligned}$$

As a whole we thus have the following:

$$\begin{aligned}\mu_* &= 0 && (5.98)\\ \sigma_*^2 &= \alpha r + \tfrac{1}{2}\gamma b_d^2 r_n (b_s + \gamma + b_s^2) + \Delta^2 q && (5.99)\end{aligned}$$

Derivation Part E: solving for α

We can now solve simultaneously for q, r, r_n, and m_1^k (from here on, we will refer to m_1^k as m). We list the four equations of interest [Eqn. (5.70), Eqn. (5.75), Eqn. (5.84), Eqn. (5.87)] here again for convenience:

$$q = \int \frac{dz}{\sqrt{2\pi}} \exp(-\frac{1}{2}z^2) \tanh^2(\beta(m + \sigma_* z)) \tag{5.100}$$

$$r = \frac{q}{[1 - \beta(1 - q)]^2} \tag{5.101}$$

$$r_n = \frac{qb}{[1 - \beta(1 - q)]^2} \tag{5.102}$$

$$m = \int \frac{dz}{\sqrt{2\pi}} \exp(-\frac{1}{2}z^2) \tanh[\beta(m + \sigma_* z)], \tag{5.103}$$

where $\sigma_*^2 = \alpha r + \frac{1}{2}\gamma b_d^2 r_n(b_s + \gamma + b_s^2) + \Delta^2 q$. These equations resemble the resulting equations of the standard Hopfield network very closely. The only difference is that the variance of z has changed in Eqn. (5.100) and Eqn. (5.101) and there is an extra equation for r_n in Eqn. (5.102). We will take the same approach as for the standard Hopfield network, following [Hertz et al., 1991], and solve the equations in the limit of $\beta \to \infty$. Given this limit, we can make use of the following two integral identities:

$$\int \frac{dz}{\sqrt{2\pi}} \exp(-z^2/2)(1 - \tanh^2[\beta(az + b)]) \approx \sqrt{\frac{2}{\pi}} \frac{1}{a\beta} \exp(-b^2/2a^2) \tag{5.104}$$

$$\int \frac{dz}{\sqrt{2\pi}} \exp(-z^2/2) \tanh[\beta(az + b)] \approx \text{erf}(\frac{b}{\sqrt{2}a}) \tag{5.105}$$

This enables us to write the four equations as:

$$C \triangleq \beta(1 - q) = \sqrt{\frac{2}{\pi\sigma_*^2}} \exp\Big(-\frac{m^2}{2\sigma_*^2}\Big) \tag{5.106}$$

$$r = \frac{q}{(1 - C)^2} \tag{5.107}$$

$$r_a = \frac{qb}{(1 - C)^2} \tag{5.108}$$

$$m = \text{erf}\left(\frac{m}{\sqrt{2\sigma_*^2}}\right) \tag{5.109}$$

We see that $q \to 1$ as $\beta \to \infty$, and the expression $C \triangleq \beta(1 - q)$ is undetermined for $\beta \to \infty$. We can, however, express its limit with respect to m and r. Given

that $q \to 1$, we can rewrite equations Eqn. (5.101) and Eqn. (5.102) as:

$$r = \frac{1}{(1-C)^2} \tag{5.110}$$

$$r_n = \frac{b}{(1-C)^2} \tag{5.111}$$

which can be rearranged to get $C = 1 - \frac{1}{\sqrt{r}}$, and therefore $r_n = rb$. We can thus write the solution as two equations which can be simultaneously solved for m and r, given a particular capacity α as well as the other parameters of σ_*^2:

$$1 - \frac{1}{\sqrt{r}} = \sqrt{\frac{2}{\pi\sigma_*^2}} \exp\Big(- \frac{m^2}{2\sigma_*^2}\Big) \tag{5.112}$$

$$m = \mathrm{erf}\left(\frac{m}{\sqrt{2\sigma_*^2}}\right) \tag{5.113}$$

5.B Supplementary Figures

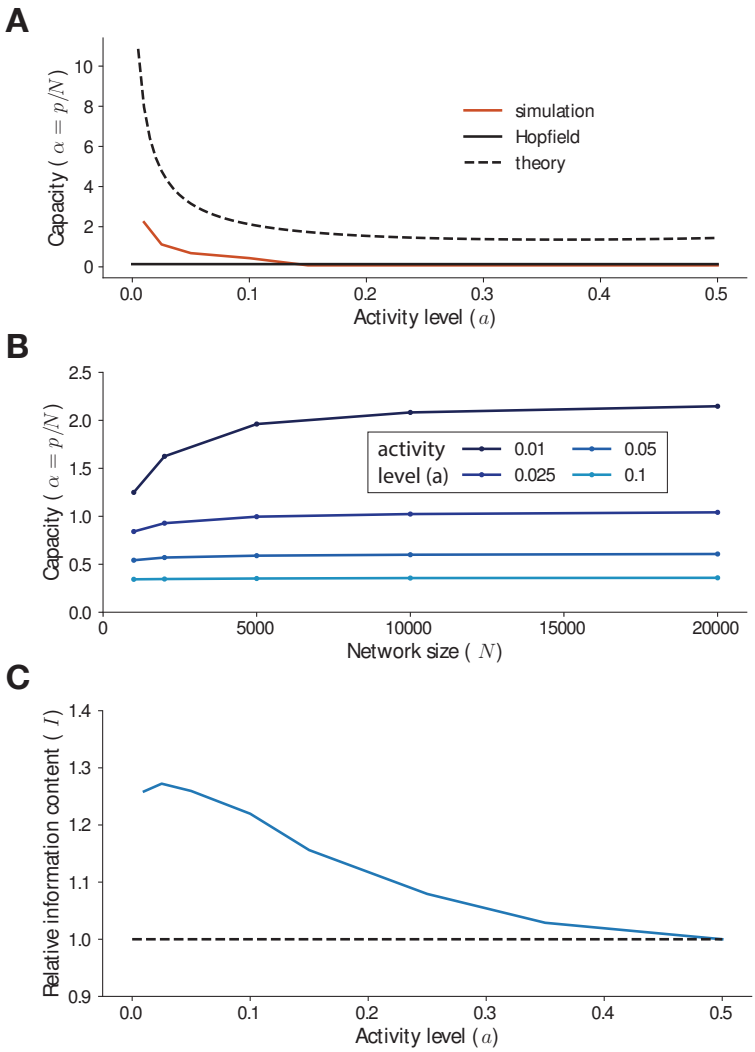


Figure 5.10: Memory capacity of low-activity Hopfield network. **A**: Comparison of theoretical (dotted black) and numerical (red) estimations of memory capacity for the low-activity Hopfield network with $\{0, 1\}$ units as in Tsodyks and Feigelman [1988] as a function of activity level a. Standard Hopfield capacity (0.138) is plotted as solid black line for reference. **B**: Numerical estimation of memory capacity as a function of network size, for different activity levels a. Plot in **A** used $N = 20000$. **C**: Information content (relative to the standard Hopfield network) as determined from numerical simulations (red curve in **A**). See methods section 5.4.3 for details.

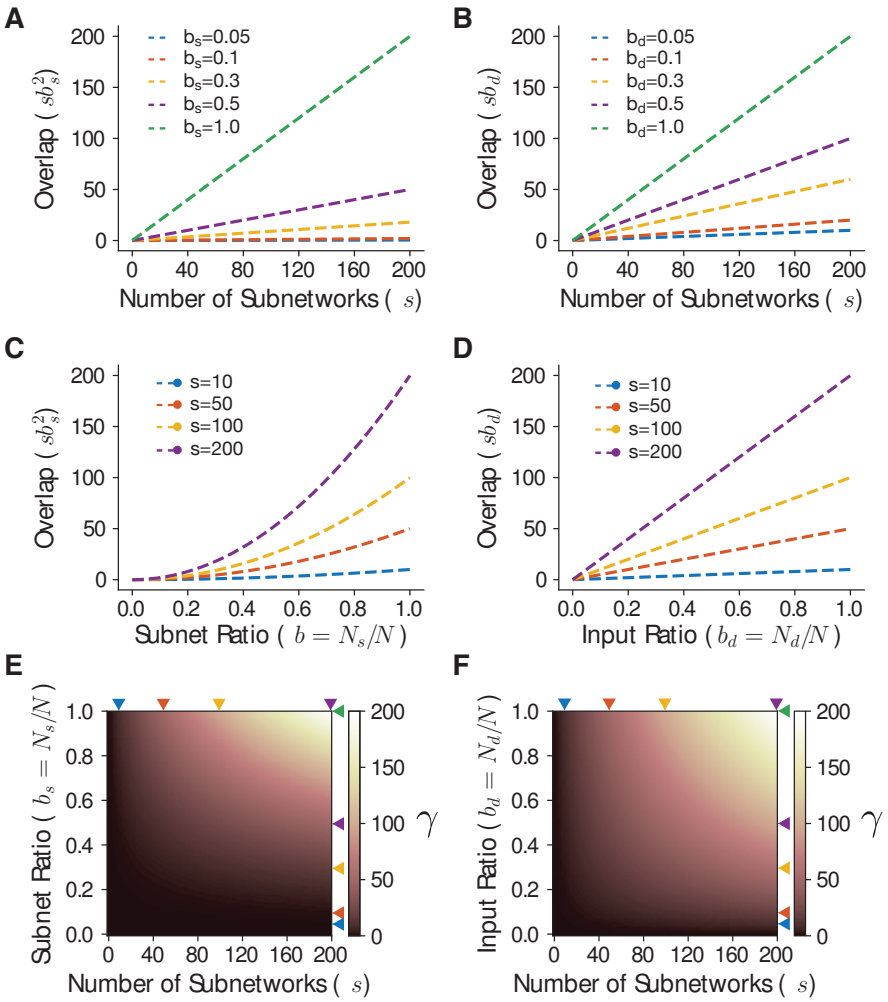


Figure 5.11: Comparison of overlap of the context-modular Hopfield network with subnetwork selection (**A,C,E**) vs. dendritic input selection **B,D,F**. Overlap is defined as sb_s^2 for subnetwork selection and sb_d for dendritic input selection.

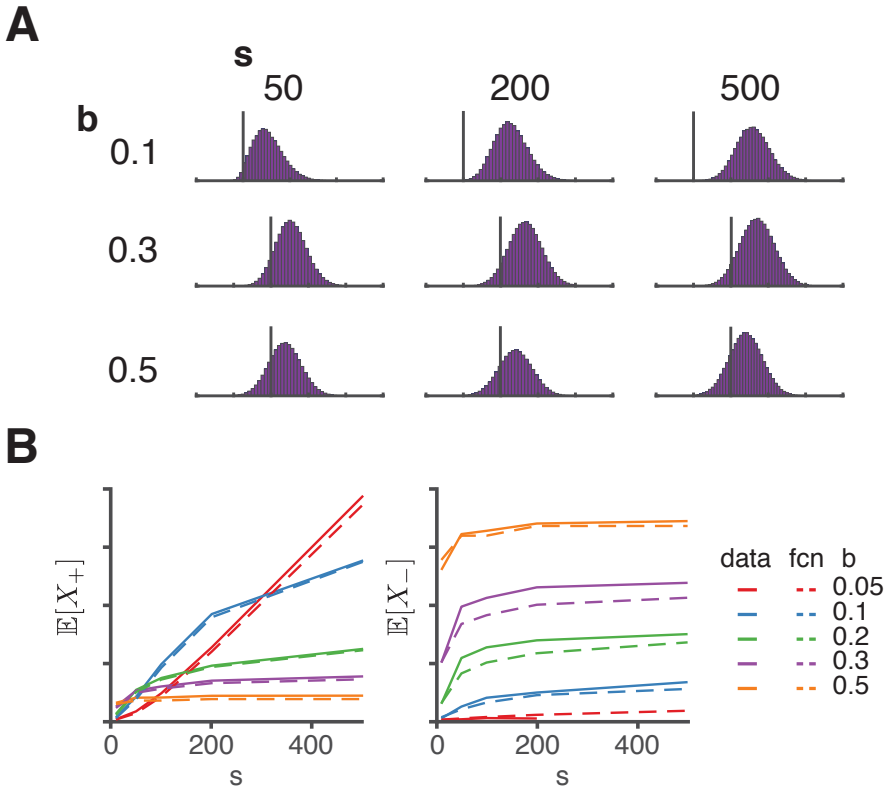


Figure 5.12: Noise distribution and variance estimation for mean-field method. **A**: Distributions of crosstalk noise for "irrelevant" memories over different numbers of contexts (s) and relative subnetwork sizes (b_s). **B**: Due to non-zero mean, variance of the distribution was measured using the second moment of all positive values of the distribution ($\mathbb{E}[X_+]$) and the second moment of all negative values of the distribution ($\mathbb{E}[X_-]$). See Eqn. (5.91)

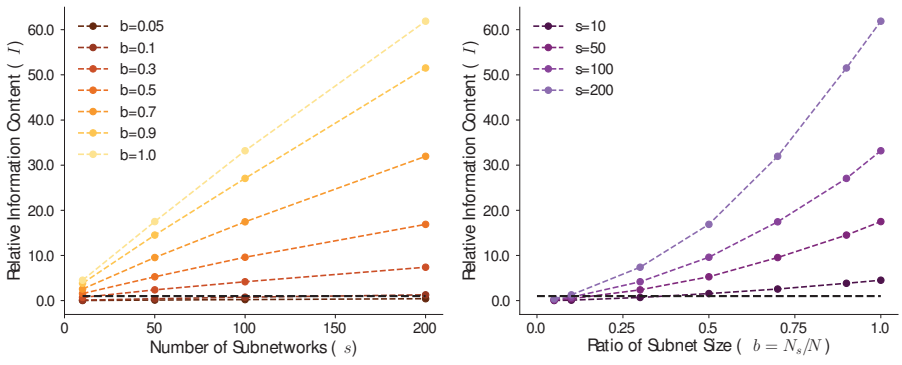


Figure 5.13: Relative information content of context-modular Hopfield network with targeted dendritic input gating (T-DIG). Shown as a function of the number of subnetworks (s) for fixed subnet size ratios (b_s) on the left, and vice versa on the right.

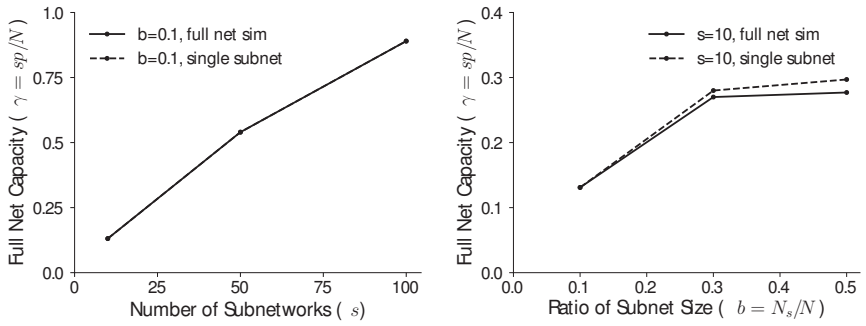


Figure 5.14: Evaluation of simulation aproximation procedure. Single subnetworks are simulated with the effects of other subnetworks being approximated as noise, and compared with full network simulations in which all subnetworks are explicitly modelled. Shown for $b = 0.1$ as a function of s (left), and for $s = 10$ as a function of b (right).

6
Conclusion

The work presented in this thesis spans several levels of description of brain activity–from the molecular underpinnings of single-neuron physiology, to the abstract dynamics of memories in neural circuits. As stated earlier, I believe that the two components of the thesis highlight two current directions of research in neuroscience – the development of resources for data and models, i.e., neuroinformatics, as well as the push to understand circuits at the network level. Here, I briefly describe the results obtained in each of these parts, as well as some future directions to go in.

Ion channels and biophysical neuron models

In part I of the thesis, I presented the state of biophysical modelling as an active community with a large number of models and data, but with a lack of standardization and deep understanding behind many of the models and how they compare to one another. To address this concern, I presented two studies which aim to address this by building tools to help towards better understanding and utilization of models, specifically applied to neuronal ion channels.

In Chapter 2, I presented a resource called ICGenealogy, which (now) hosts over 3000 ion channel models, providing detailed metadata information and a quantitative characterization of ion channel behavior for each one. This resource

was created with the motivation to facilitate better experimentally-constrained modelling, as well as provide some insights into the variability and validity of ion channel models used in detailed neuronal simulations. This work was published in eLife last year [Podlaski et al., 2017].

The ICGenealogy resource is based upon a detailed curation and characterization of ion channel models coded in the NEURON language, as available on ModelDB. Metadata information was collected for each model, which included important biological details, such as neuron type, brain area and animal species, as well as modelling details, such as temperature dependence and model authors. Critically, this metadata was collected manually, and is based upon information taken from the associated journal articles which published the models, as well as any extra information found in the model code itself. While this may lead to some issues regarding fidelity and model coverage, it remains the only feasible method for aggregating this type of data.

The second major component of the ICG resource was the quantitative characterization of the models in the database through a series of five voltage-clamp protocols. This allowed us to compare the behavioral responses of the models without concern for the underlying implementations, which may be varied and idiosyncratic. We were able to cluster model responses into different groups and to compare the behavior of the models with the metadata information. This enabled us to have an internal check on the validity of the metadata information, which for the most part followed the quantitative characterization. Together, the metadata collection and voltage-clamp protocol characterization make up the core of the database. In order to provide the infrastructure to keep the database up to date, as well as to interact more closely with experiments, we also added a functionality to the database which allows users to upload new models and experimental data, and compare immediately with the database of models.

The intentions behind this work were twofold: (i) to provide a resource for the community for better experimentally-constrained modelling, and (ii) to explore the space of ion channel models in order to gain a better understanding of ion

channel dynamics and how they relate to particular neuron types, species, and other biological data. Based on data collected about the number and type of visits to the ICGenealogy website, we can safely say that the ICG resource is currently being used at least by a subset of the biophysical modelling community to discover and compare currently available ion channel models, as well as to choose models for new simulations. Thus, some progress towards these goals has been made.

However, there are several limitations to the work. First of all, the fact that we are strictly comparing models, instead of focusing on the true biological systems, suggests that there is a degree of biological realism that is missing in the ICG resource. This means that questions arising about the validity of particular models over others, or the amount of variability seen in models of any given type, may not be answerable at the current moment. Unfortunately, these questions are also difficult to address experimentally. This is not only due to a similar lack of standardization in experimental research, but also due to the fact that it is difficult to isolate individual ion channel currents, which would be necessary to fully characterize each one. Furthermore, while we are observing some regular online visits to the ICG website, there are very few users who seem to be interested in uploading data.

In any case, the limitations of this work, as well as feedback from the community, actually provided a motivation for Chapter 3 of the thesis, which focused on fitting a standardized ion channel model to all existing models on ICGenealogy. Although this effort has similar limitations in terms of biological fidelity, we were excited by the prospect of obtaining a more interpretable state space of models – parameter space.

In this follow-up work to ICG, we addressed some of the limitations of the ICGenealogy resource by taking a different approach towards understanding ion channel behavior. We revived the idea of a standardized model of ion channel dynamics, and applied this model formalism to the ICGenealogy database. The novelty in this study was not in the formulation, but rather the fact that we now had a large database of ion channel models with which to test and fit this standardized model. While this is work remains in progress, we hoped to illustrate how a relatively simple ion channel formulation can provide significant benefits to the field.

More specifically, in this work we proposed a thermodynamic formalism for ion channel models, comprising a modified sigmoid function for activation and inactivation steady states, and a second-order thermodynamic formalism for time constants. We were able to successfully fit the majority of ion channel models to the formalism. The main focus of the chapter was on the fitting procedure, as well as the validation of this fitting procedure, both by directly comparing the fitted curves, as well as comparing the corresponding current responses to the ICGenealogy voltage-clamp protocols. Finally, we explored the effects of this standardized model formulation in a full single-neuron simulation of a layer 2/3 pyramidal neuron with six or eight models replaced. Interestingly, despite very good fits with low error, we observed several deviations from the results seen with the original models.

The fact that it was so difficult to compare the standardized models with the originals in full neuronal simulations highlights an important limitation of the study, as well as a general point about biophysical modelling. Our method of directly comparing results between the original and modified models may not be the right criterion by which to validate the models. However, due to the very nature of multicompartmental models as being high-dimensional and flexible, but also very sensitive, means that there may not be a good way of comparing models. One potential remedy to this would be to devise a "sensitivity" or "robustness" measure for each model based on the size of the parameter space for which the correct outcome is observed. We could then limit our analysis to models which pass a certain robustness threshold, thereby ensuring that a standardized model with a close enough fit should be able to achieve the correct output.

Overall, it is my hope that the efforts made in the first part of this thesis will provide some benefit to the community of computational neuroscientists. However, these are just the first steps in the general effort to develop neuroinformatics tools for computational neuroscience, and there is more work to be done. We hope to continue working on the standardized model formalism in order to push towards more standardized and reusable biophysical models, such as those espoused by Open Source Brain. In fact, one long-term goal of this project is to build a

framework within which any ion channel model can be automatically converted to a standardized model description in NeuroML.

Furthermore, one of the main goals of both projects was to enable easier comparison of models – whether it be in current response space or parameter space. While we were able to explore this briefly here, it remains a large task for future work, in which more sophisticated methods in machine learning may be helpful to understand the space of models.

Finally, moving on to other aspects of biophysical modelling, the sort of cataloguing and analysis done here could be applied to groups of ion channels together, as well as to morphologies. Admittedly, these would be substantially more difficult to analyze compared to single ion channel models, we must work towards understanding them if we would like to have a complete understanding of neuronal dynamics.

The Hopfield network and associative memory

In the second part of this thesis, the theme shifted to abstract network models of neuronal function, with a special focus on memory. Memory research has had a rich and long history, but arguably is still a young field with room for many new ideas. However, while new models of memory function continue to be published, old models such as the Hopfield network continue to act as the primary influences on many views of memory function. Despite now being over 30 years old, the Hopfield model is still used and adapted in current models alongside the integration of new experimental findings.

The focus of the model that I presented in Chapter 5 was on context-dependent memory. The motivation for the model comes out of a large body of research which explores the organization of memory, and how it is influenced by the environment and the current behavioral state of an organism. Research in psychology has explored this and related ideas for many decades, but arguably many of the network models have not incorporated it.

We thus proposed a new model of context-dependent associative memory based upon the Hopfield network formalism. The network groups memories into several contexts, with different underlying network states used to store memories from each group. The network states are defined by different sets of active neurons, or subnetworks, as well as sets of active inputs per neuron. We show that this modular architecture has interesting effects on the capacity of the network to store memories, as well as the expression of memories – depending upon the current context, some memories will be able to be recalled, while others remain dormant.

Excitingly, we think that this model may tie in with research on the interaction of different memory systems in the brain, especially relating to the hippocampus and prefrontal cortex. These areas have been known to participate in memory function, but the exact representations, coding schemes, and interactions are not yet clear. We propose that a hierarchical context-dependent architecture may explain how the hippocampus interacts with the cortex – the hippocampus acts as a context-encoding network, which controls memory expression throughout the cortex. Or alternatively, this model may explain some sub-circuitry within the hippocampus – context-dependent control of network state provides an interesting mechanism for pattern separation, which is known to occur in the hippocampus.

The model presented relies on an extremely simple and abstract formalism – the Hopfield model – and thus has several limitations. Previous studies have attempted to adapt the Hopfield model into a more realistic model of memory formation only to realize that it becomes much more difficult when memories must be stored slowly over time, and realistic spiking neurons must be used [Fusi and Abbott, 2007, Zenke et al., 2015]. Furthermore, while a simple Hebbian learning rule can be used to store memory patterns, there does not appear to be a simple way of learning the contextual network states, especially dendritic gating states. The complexities of learning appropriate contextual states may turn out to outweigh the potential benefits of the architecture. However, in any case, I would propose that this study mainly serves as a conceptual model of how context might interact with long-term

memory, and the potential benefits that can come with it. Therefore, it can serve as an idealistic scenario that can be made more biologically plausible in the future.

Future work will hopefully be able to determine if this type of model can be implemented in a more realistic setting. At the circuit level, this model can be made more realistic by using more detailed rate or spiking neuron models, as well as incorporating explicit inhibitory cell types. In doing so, we would then be able to directly compare the model with experimental data. Furthermore, an important direction to explore will be to propose learning rules both for the memory network itself, as well as for the contextual network configurations. Finally, another interesting direction would be to incorporate this type of model into a setting with a functional task – this would enable a test to see if dynamic control of memory expression has some benefits, or if it is too costly to control memory expression and it is better to allow all memories to be available at any given time.

Bibliography

Abbott L., and Kepler T.B. (1990). Model neurons: From hodgkin-huxley to hopfield. In Statistical mechanics of neural networks (Springer), pp. 5–18.

Abbott L.F. (1990). Learning in neural network memories. Network: Computation in neural systems *1*, 105–122.

Abbott L.F. (2008). Theoretical neuroscience rising. Neuron *60*, 489–495.

Abbott L.F., DePasquale B., and Memmesheimer R.M. (2016). Building functional networks of spiking model neurons. Nature neuroscience *19*, 350.

Abbott L.F., and Nelson S.B. (2000). Synaptic plasticity: taming the beast. Nature neuroscience *3*, 1178.

Abraham W.C., and Robins A. (2005). Memory retention–the synaptic stability versus plasticity dilemma. Trends in neurosciences *28*, 73–78.

Aertsen A., and Johannesma P. (1981). The spectro-temporal receptive field. Biological cybernetics *42*, 133–143.

Aiken S.P., Lampe B.J., and Brown B.S. (1995). Reduction of spike frequency adaptation and blockade of m-current in rat ca1 pyramidal neurones by linopirdine (dup 996), a neurotransmitter release enhancer. British journal of pharmacology *115*, 1163–1168.

Akil H., Martone M.E., and Van Essen D.C. (2011). Challenges and opportunities in mining neuroscience data. Science (New York .

Alberini C.M. (2005). Mechanisms of memory stabilization: are consolidation and reconsolidation similar or distinct processes? Trends in neurosciences *28*, 51–56.

Alcock J., and Rubenstein D.R. (1989). Animal behavior (Sinauer Associates Sunderland, MA, USA).

Aldrich R., Corey D., and Stevens C. (1983). A reinterpretation of mammalian sodium channel gating based on single channel recording. Nature *306*, 436–441.

Allen-Institute (2015). Allen cell types database. [Online; accessed 21-Sept-2018: http://help.brain-map.org/display/celltypes/Documentation].

Amit D.J. (1992). Modeling brain function: The world of attractor neural networks (Cambridge university press).

Amit D.J., Gutfreund H., and Sompolinsky H. (1985). Storing infinite numbers of patterns in a spin-glass model of neural networks. Physical Review Letters *55*, 1530.

Amit D.J., Gutfreund H., and Sompolinsky H. (1987a). Information storage in neural networks with low levels of activity. Physical Review A *35*, 2293.

Amit D.J., Gutfreund H., and Sompolinsky H. (1987b). Statistical mechanics of neural networks near saturation. Annals of physics *173*, 30–67.

Amit D.J., and Tsodyks M. (1991). Quantitative study of attractor neural network retrieving at low spike rates: I. substrate—spikes, rates and neuronal gain. Network: Computation in neural systems *2*, 259–273.

Armstrong C.M. (1981). Sodium channels and gating currents. Physiological reviews *61*, 644–683.

Ascoli G.A., Donohue D.E., and Halavi M. (2007). Neuromorpho. org: a central resource for neuronal morphologies. Journal of Neuroscience *27*, 9247–9251.

Ashburner M., Ball C.A., Blake J.A., Botstein D., and Butler H. (2000). Gene Ontology: tool for the unification of biology. Nature Genetics *20*, 25–29.

Baddeley A.D., and Hitch G. (1974). Working memory. In Psychology of learning and motivation (Elsevier), vol. 8, pp. 47–89.

Barak O. (2017). Recurrent neural networks as versatile tools of neuroscience research. Current opinion in neurobiology *46*, 1–6.

Barron H., Vogels T., Emir U., Makin T., O'shea J., Clare S., Jbabdi S., Dolan R.J., and Behrens T. (2016). Unmasking latent inhibitory connections in human cortex to reveal dormant cortical memories. Neuron *90*, 191–203.

Bartlett F.C. (1995). Remembering: A study in experimental and social psychology, vol. 14 (Cambridge University Press).

Bastian M., Heymann S., and Jacomy M. (2009). Gephi: An open source software for exploring and manipulating networks. International AAAI Conference on Weblogs and Social Media .

Bazhenov M., Stopfer M., Rabinovich M., Huerta R., Abarbanel H.D., Sejnowski T.J., and Laurent G. (2001). Model of transient oscillatory synchronization in the locust antennal lobe. Neuron *30*, 553–567.

Bear M.F., and Kirkwood A. (1993). Neocortical long-term potentiation. Current opinion in neurobiology *3*, 197–202.

Beaulieu C. (1993). Numerical data on neocortical neurons in adult rat, with special reference to the gaba population. Brain research *609*, 284–292.

Ben-Yishai R., Bar-Or R.L., and Sompolinsky H. (1995). Theory of orientation tuning in visual cortex. Proceedings of the National Academy of Sciences *92*, 3844–3848.

Bi G.q., and Poo M.m. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. Journal of neuroscience *18*, 10464–10472.

Bliss T.V., and Lømo T. (1973). Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path. The Journal of physiology *232*, 331–356.

Borg-Graham L.J. (1991). Modeling the non-linear conductances of excitable membranes. Cellular Neurobiology: A Practical Approach *13*, 247–275.

Bostock M., Ogievetsky V., and Heer J. (2011). D3 data-driven documents. IEEE Transactions on Visualization and Computer Graphics *17*, 2301–2309.

Bouton M.E. (1993). Context, time, and memory retrieval in the interference paradigms of pavlovian learning. Psychological bulletin *114*, 80.

Bouton M.E. (2007). Learning and behavior: A contemporary synthesis. (Sinauer Associates).

Brette R. (2015a). Philosophy of the spike: rate-based vs. spike-based theories of the brain. Frontiers in systems neuroscience *9*, 151.

Brette R. (2015b). What is the most realistic single-compartment model of spike initiation? PLoS computational biology *11*, e1004114.

Brette R., and Gerstner W. (2005). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. Journal of neurophysiology *94*, 3637–3642.

Brette R., Rudolph M., Carnevale T., Hines M., and et al. (2007). Simulation of networks of spiking neurons: a review of tools and strategies. Journal of Computational Neuroscience *23*, 349–398.

Britten K.H., Shadlen M.N., Newsome W.T., and Movshon J.A. (1992). The analysis of visual motion: a comparison of neuronal and psychophysical performance. Journal of Neuroscience *12*, 4745–4765.

Brunel N. (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. Journal of computational neuroscience *8*, 183–208.

Brunel N. (2016). Is cortical connectivity optimized for storing information? Nature neuroscience *19*, 749.

Burak Y., and Fiete I.R. (2012). Fundamental limits on persistent activity in networks of noisy neurons. Proceedings of the National Academy of Sciences .

Cai X., Liang C.W., Muralidharan S., Kao J., and Tang C.M. (2004). Unique roles of SK and Kv4. 2 potassium channels in dendritic integration. Neuron *44*, 351–364.

Caliński T., and Harabasz J. (1974). A dendrite method for cluster analysis. Communications in Statistics *3*, 1–27.

Campiglio M., and Flucher B.E. (2015). The Role of Auxiliary Subunits for the Functional Diversity of Voltage-Gated Calcium Channels.

Cannon R.C., Gleeson P., Crook S., Ganapathy G., Marin B., Piasini E., and Silver R.A. (2014). LEMS: a language for expressing complex biological models in concise and hierarchical form and its use in underpinning NeuroML 2. Frontiers in Neuroinformatics *8*.

Carandini M., Demb J.B., Mante V., Tolhurst D.J., Dan Y., Olshausen B.A., Gallant J.L., and Rust N.C. (2005). Do we know what the early visual system does? Journal of Neuroscience *25*, 10577–10597.

Carandini M., and Heeger D.J. (2012). Normalization as a canonical neural computation. Nature Reviews Neuroscience *13*, 51.

Carbonell-Pascual B., Godoy E., Ferrer A., Romero L., and Ferrero J.M. (2016). Comparison between hodgkin–huxley and markov formulations of cardiac ion channels. Journal of theoretical biology *399*, 92–102.

Carnevale N., and Hines M. (2014). NEURON for empirically-based simulations of neurons and networks of neurons. [Online; accessed 26-May-2015: http://www.neuron.yale.edu/neuron].

Carnevale N.T., and Hines M.L. (2006). The NEURON book (Cambridge University Press).

Carrillo-Reid L., Yang W., Bando Y., Peterka D.S., and Yuste R. (2016). Imprinting and recalling cortical ensembles. Science *353*, 691–694.

Catterall W.A. (1984). The Molecular Basis of Neuronal Excitability. Science *223*, 653–661.

Catterall W.a. (1995). Structure and function of voltage-gated ion channels. Annual review of biochemistry *64*, 493–531.

Catterall W.A., Dib-Hajj S., and Meisler M.H. (2008). Inherited neuronal ion channelopathies: new windows on complex neurological diseases. The Journal of Neuroscience *28*, 11768–11777.

Chandy K.G. (1991). Simplified gene nomenclature. Nature *352*, 26.

Chapeton J., Fares T., LaSota D., and Stepanyants A. (2012). Efficient associative memory storage in cortical circuits of inhibitory and excitatory neurons. Proceedings of the National Academy of Sciences *109*, E3614–E3622.

Chaudhuri R., and Fiete I. (2016). Computational principles of memory. Nature neuroscience *19*, 394.

Cherry J.L., and Adler F.R. (2000). How to make a biological switch. Journal of theoretical biology *203*, 117–133.

Chevenet F., Brun C., Bañuls A.L., Jacq B., and Christen R. (2006). TreeDyn: towards dynamic graphics and annotations for analyses of trees. BMC Bioinformatics *7*, 439.

Churchland M.M., Cunningham J.P., Kaufman M.T., Foster J.D., Nuyujukian P., Ryu S.I., and Shenoy K.V. (2012). Neural population dynamics during reaching. Nature *487*, 51.

Cohen M.A., and Grossberg S. (1983). Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. IEEE transactions on systems, man, and cybernetics pp. 815–826.

Cohen N.J., and Squire L.R. (1980). Preserved learning and retention of pattern-analyzing skill in amnesia: Dissociation of knowing how and knowing that. Science *210*, 207–210.

Cole K.S., and Curtis H.J. (1939). Electric impedance of the squid giant axon during activity. The Journal of general physiology *22*, 649–670.

Colquhoun D., and Hawkes A. (1981). On the stochastic properties of single ion channels. Proc. R. Soc. Lond. B *211*, 205–235.

Constantinescu A.O., O'Reilly J.X., and Behrens T.E. (2016). Organizing conceptual knowledge in humans with a gridlike code. Science *352*, 1464–1468.

Costa R., Assael I.A., Shillingford B., de Freitas N., and Vogels T. (2017). Cortical microcircuits as gated-recurrent neural networks. In Advances in Neural Information Processing Systems. pp. 272–283.

Cowansage K.K., Shuman T., Dillingham B.C., Chang A., Golshani P., and Mayford M. (2014). Direct reactivation of a coherent neocortical memory of context. Neuron *84*, 432–441.

Cragg B.G., and Temperley H.N.V. (1955). Memory: the analogy with ferromagnetic hysteresis. Brain *78*, 304–316.

Crill W.E. (1996). Persistent sodium current in mammalian central neurons. Annual review of physiology *58*, 349–362.

Curti E., Mongillo G., Camera G.L., and Amit D.J. (2004). Mean field and capacity in realistic networks of spiking neurons storing sparsely coded random memories. Neural computation *16*, 2597–2637.

Davies D.L., and Bouldin D.W. (1979). A Cluster Separation Measure. IEEE Trans. Pattern Anal. Mach. Intell. *1*, 224–227.

Dayan P., and Abbott L.F. (2001). Theoretical neuroscience, vol. 806 (Cambridge, MA: MIT Press).

Dayan P., and Berridge K.C. (2014). Model-based and model-free pavlovian reward learning: revaluation, revision, and revelation. Cognitive, Affective, & Behavioral Neuroscience *14*, 473–492.

Dayan P., and Daw N.D. (2008). Decision theory, reinforcement learning, and the brain. Cognitive, Affective, & Behavioral Neuroscience *8*, 429–453.

De Carlos J.A., and Borrell J. (2007). A historical reflection of the contributions of cajal and golgi to the foundations of neuroscience. Brain research reviews *55*, 8–16.

Debanne D., Daoudal G., and Sourdet V. (2003). Brain plasticity and ion channels. Journal of Physiology-Paris *27*, 403–414.

DePasquale B., Cueva C.J., Rajan K., Abbott L., et al. (2018). full-force: A target-based method for training recurrent networks. PloS one *13*, e0191527.

Derrida B., Gardner E., and Zippelius A. (1987). An exactly solvable asymmetric neural network model. EPL (Europhysics Letters) *4*, 167.

Destexhe A., and Huguenard J.R. (2000a). 5. Which formalism to use for modeling voltage-dependent conductances? Computational Neuroscience: Realistic Modeling for Experimentalists pp. 129–157.

Destexhe A., and Huguenard J.R. (2000b). Nonlinear thermodynamic models of voltage-dependent currents. Journal of computational neuroscience *9*, 259–270.

Destexhe A., and Huguenard J.R. (2000c). Which formalism to use for modeling voltage-dependent conductances. Computational neuroscience: realistic modeling for experimentalists pp. 129–157.

Destexhe A., and Paré D. (1999). Impact of network activity on the integrative properties of neocortical pyramidal neurons in vivo. Journal of neurophysiology *81*, 1531–1547.

Destexhe A., Rudolph M., and Paré D. (2003). The high-conductance state of neocortical neurons in vivo. Nature reviews neuroscience *4*, 739.

DiCarlo J.J., Zoccolan D., and Rust N.C. (2012). How does the brain solve visual object recognition? Neuron *73*, 415–434.

DiFrancesco D., and Tortora P. (1991). Direct activation of cardiac pacemaker channels by intracellular cyclic amp. Nature *351*, 145.

Doboli S., Minai A.A., and Best P.J. (2000). Latent attractors: a model for context-dependent place representations in the hippocampus. Neural Computation *12*, 1009–1043.

Doll B.B., Simon D.A., and Daw N.D. (2012). The ubiquity of model-based reinforcement learning. Current opinion in neurobiology *22*, 1075–1081.

Douglas R.J., Koch C., Mahowald M., Martin K., and Suarez H.H. (1995). Recurrent excitation in neocortical circuits. Science *269*, 981–985.

Douglas R.J., and Martin K.A. (2007). Recurrent neuronal circuits in the neocortex. Current biology *17*, R496–R500.

Druckmann S., Berger T.K., Schürmann F., Hill S., Markram H., and Segev I. (2011). Effective stimuli for constructing reliable neuron models. PLoS Computational Biology *7*.

Dubreuil A.M., and Brunel N. (2016). Storing structured sparse memories in a multi-modular cortical network model. Journal of computational neuroscience *40*, 157–175.

Dudek S.M., and Bear M.F. (1992). Homosynaptic long-term depression in area ca1 of hippocampus and effects of n-methyl-d-aspartate receptor blockade. Proceedings of the National Academy of Sciences *89*, 4363–4367.

Dunn J.C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. Journal of Cybernetics *3*, 32–57.

Ebbinghaus H. (2013). Memory: A contribution to experimental psychology. Annals of neurosciences *20*, 155.

Edin F., Klingberg T., Johansson P., McNab F., Tegnér J., and Compte A. (2009). Mechanism for top-down control of working memory capacity. Proceedings of the National Academy of Sciences *106*, 6802–6807.

Eichenbaum H. (2017). Prefrontal–hippocampal interactions in episodic memory. Nature Reviews Neuroscience *18*, 547.

Eichenbaum H., and Cohen N.J. (2004). From conditioning to conscious recollection: Memory systems of the brain. 35 (Oxford University Press on Demand).

Eichenbaum H., and Cohen N.J. (2014). Can we reconcile the declarative memory and spatial navigation views on hippocampal function? Neuron *83*, 764–770.

Eichenbaum H., Dudchenko P., Wood E., Shapiro M., and Tanila H. (1999). The hippocampus, memory, and place cells: is it spatial memory or a memory space? Neuron *23*, 209–226.

Ertel E.A., Campbell K.P., Harpold M.M., and Hofmann F. (2000). Nomenclature of voltage-gated calcium channels. Neuron *25*, 533–535.

Evan D., Seth W., Devika D., Jing J., and Mehrdad M. (2018). A dynamical systems perspective on flexible motor timing. Trends in Cognitive Sciences .

Fairhall A. (2014). The receptive field is dead. long live the receptive field? Current opinion in neurobiology *25*, ix–xii.

Feldmeyer D., Qi G., Emmenegger V., and Staiger J.F. (2018). Inhibitory interneurons and their circuit motifs in the many layers of the barrel cortex. Neuroscience *368*, 132–151.

Felleman D.J., and Van Essen D.C. (1991). Distributed hierarchical processing in the primate cerebral cortex. Cerebral cortex (New York, NY: 1991) *1*, 1–47.

Ferguson A.R., Nielson J.L., Cragin M.H., Bandrowski A.E., and Martone M.E. (2014). Big data from small data: data-sharing in the 'long tail' of neuroscience. Nature Neuroscience *17*, 1442–1447.

Fineberg J.D., Ritter D.M., and Covarrubias M. (2012). Modeling-independent elucidation of inactivation pathways in recombinant and native A-type Kv channels. The Journal of General Physiology *140*, 513–527.

Fink M., and Noble D. (2009). Markov models for ion channels: versatility versus identifiability and speed. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences *367*, 2161–2179.

Fourcaud-Trocmé N., Hansel D., Van Vreeswijk C., and Brunel N. (2003). How spike generation mechanisms determine the neuronal response to fluctuating inputs. Journal of Neuroscience *23*, 11628–11640.

Fusi S., and Abbott L. (2007). Limits on the memory storage capacity of bounded synapses. Nature neuroscience *10*, 485.

Gabbott P.L., and Bacon S.J. (1996). Local circuit neurons in the medial prefrontal cortex (areas 24a, b, c, 25 and 32) in the monkey: Ii. quantitative areal and laminar distributions. Journal of Comparative Neurology *364*, 609–636.

Gabbott P.L., Dickie B.G., Vaid R.R., Headlam A.J., and Bacon S.J. (1997). Local-circuit neurones in the medial prefrontal cortex (areas 25, 32 and 24b) in the rat: morphology and quantitative distribution. Journal of Comparative Neurology *377*, 465–499.

Gao P., Trautmann E., Byron M.Y., Santhanam G., Ryu S., Shenoy K., and Ganguli S. (2017). A theory of multineuronal dimensionality, dynamics and measurement. bioRxiv p. 214262.

Gardner E. (1988). The space of interactions in neural network models. Journal of physics A: Mathematical and general *21*, 257.

Gasque G., Labarca P., Reynaud E., and Darszon A. (2005). Shal and shaker differential contribution to the K+ currents in the Drosophila mushroom body neurons. The Journal of neuroscience : the official journal of the Society for Neuroscience *25*, 2348–2358.

Gershman S.J., Monfils M.H., Norman K.A., and Niv Y. (2017). The computational nature of memory modification. Elife *6*, e23763.

Gerstner W., Kistler W.M., Naud R., and Paninski L. (2014a). Neuronal Dynamics. From Single Neurons to Networks and Models of Cognition (Cambridge University Press).

Gerstner W., Kistler W.M., Naud R., and Paninski L. (2014b). Neuronal dynamics: From single neurons to networks and models of cognition (Cambridge University Press).

Gerstner W., and Naud R. (2009). How good are neuron models? Science *326*, 379–380.

Geschwind N. (1970). The organization of language and the brain. Science *170*, 940–944.

Geszti T. (1990). Physical models of neural networks (World Scientific).

Gleeson D.P., Silver P.A., and Cantarelli M. (2013). Open source brain. In Encyclopedia of Computational Neuroscience, D. Jaeger, and R. Jung, eds. (Springer New York), pp. 1–3.

Gleeson P., Cantarelli M., Marin B., Quintana A., Earnshaw M., Piasini E., Birgiolas J., Cannon R.C., Cayco-Gajic N.A., Crook S., et al. (2018). Open source brain: a collaborative resource for visualizing, analyzing, simulating and developing standardized models of neurons and circuits. bioRxiv p. 229484.

Gleeson P., Crook S., Cannon R.C., Hines M.L., Billings G.O., Farinella M., Morse T.M., Davison A.P., Ray S., Bhalla U.S., et al. (2010). Neuroml: a language for describing data driven models of neurons and networks with a high degree of biological detail. PLoS computational biology *6*, e1000815.

Gluck M.A., Mercado E., and Myers C.E. (2008). Learning and memory (Worth Publ.).

Goldberg E.M., Clark B.D., Zagha E., Nahmani M., and Erisir A. (2008). K+ channels at the axon initial segment dampen near-threshold excitability of neocortical fast-spiking GABAergic interneurons. Neuron *58*, 387–400.

Goldin A.L., Barchi R.L., Caldwell J.H., and Hofmann F. (2000). Nomenclature of voltage-gated sodium channels. Neuron *28*, 365–8.

Goldman M.S. (2009). Memory without feedback in a neural network. Neuron *61*, 621–634.

Goldman M.S., Golowasch J., Marder E., and Abbott L.F. (2001). Global structure, robustness, and modulation of neuronal models. The Journal of neuroscience : the official journal of the Society for Neuroscience *21*, 5229–38.

Golowasch J., Goldman M.S., Abbott L.F., and Marder E. (2002). Failure of averaging in the construction of a conductance-based neuron model. Journal of neurophysiology *87*, 1129–1131.

Grillner S. (2014). Megascience efforts and the brain. Neuron *82*, 1209–1211.

Grossberg S. (1968). Some nonlinear networks capable of learning a spatial pattern of arbitrary complexity. Proceedings of the National Academy of Sciences *59*, 368–372.

Gu Q. (2002). Neuromodulatory transmitter systems in the cortex and their role in cortical plasticity. Neuroscience *111*, 815–835.

Guerguiev J., Lillicrap T.P., and Richards B.A. (2017). Towards deep learning with segregated dendrites. ELife *6*, e22901.

Guskjolen A., Kenney J.W., de la Parra J., Yeung B.r.A., Josselyn S.A., and Frankland P.W. (2018). Recovery of "lost" infant memories in mice. Current Biology .

Gutfreund H. (1988). Neural networks with hierarchically correlated patterns. Physical Review A *37*, 570.

Haider B., and McCormick D.A. (2009). Rapid neocortical dynamics: cellular and network mechanisms. Neuron *62*, 171–189.

Halkidi M., Batistakis Y., and Vazirgiannis M. (2001). On clustering validation techniques. Journal of Intelligent Information Systems *17*, 107–145.

Harris K.D. (2005). Neural signatures of cell assembly organization. Nature Reviews Neuroscience *6*, 399.

Harris K.D. (2013). Top-down control of cortical state. Neuron *79*, 408–410.

Harris K.D., and Shepherd G.M. (2015). The neocortical circuit: themes and variations. Nature neuroscience *18*, 170.

Hay E., Hill S., Schürmann F., Markram H., and Segev I. (2011). Models of neocortical layer 5b pyramidal cells capturing a wide range of dendritic and perisomatic active properties. PLoS computational biology *7*, e1002107.

Hebb D.O. (1949). The organization of behavior: A neurophysiological approach.

Hennequin G., Agnes E.J., and Vogels T.P. (2017). Inhibitory plasticity: balance, control, and codependence. Annual review of neuroscience *40*, 557–579.

Hennequin G., Ahmadian Y., Rubin D.B., Lengyel M., and Miller K.D. (2018). The dynamical regime of sensory cortex: Stable dynamics around a single stimulus-tuned attractor account for patterns of noise variability. Neuron *98*, 846–860.

Hennequin G., Vogels T.P., and Gerstner W. (2014). Optimal control of transient dynamics in balanced networks supports generation of complex movements. Neuron *82*, 1394–1406.

Herculano-Houzel S. (2012). The remarkable, yet not extraordinary, human brain as a scaled-up primate brain and its associated cost. Proceedings of the National Academy of Sciences *109*, 10661–10668.

Herculano-Houzel S., Mota B., and Lent R. (2006). Cellular scaling rules for rodent brains. Proceedings of the National Academy of Sciences *103*, 12138–12143.

Hertz J., Krogh A., and Palmer R.G. (1991). Introduction to the theory of neural computation. (Addison-Wesley/Addison Wesley Longman).

Herz A.V., Gollisch T., Machens C.K., and Jaeger D. (2006). Modeling single-neuron dynamics and computations: a balance of detail and abstraction. science *314*, 80–85.

Hibino H., Inanobe A., Furutani K., Murakami S., Findlay I., and Kurachi Y. (2010). Inwardly rectifying potassium channels: their structure, function, and physiological roles. Physiological reviews *90*, 291–366.

Hill T.L., and Chen Y.D. (1972). On the theory of ion transport across the nerve membrane. Biophysical journal *12*, 960–976.

Hillar C.J., and Tran N.M. (2018). Robust exponential memory in hopfield networks. The Journal of Mathematical Neuroscience *8*, 1.

Hille B., et al. (2001). Ion channels of excitable membranes, vol. 507 (Sinauer Sunderland, MA).

Hines M., and Carnevale N.T. (2001). Neuron: a tool for neuroscientists. The Neuroscientist *7*, 123–135.

Hines M.L., and Carnevale N.T. (2000). Expanding NEURON's repertoire of mechanisms with NMODL. Neural Computation *12*, 995–1007.

Hines M.L., Morse T., Migliore M., Carnevale N.T., and Shepherd G.M. (2004). ModelDB: A Database to Support Computational Neuroscience. Journal of Computational Neuroscience *17*, 7–11. PMID: 15218350 PMCID: PMC3732827.

Hodgkin A.L., and Huxley A.F. (1952a). A quantitative description of membrane current and its application to conduction and excitation in nerve. The Journal of Physiology *117*, 500–544. PMID: 12991237 PMCID: PMC1392413.

Hodgkin A.L., and Huxley A.F. (1952b). The dual effect of membrane potential on sodium conductance in the giant axon of Loligo. The Journal of Physiology *116*, 497–506.

Hodgkin A.L., and Katz B. (1949). The effect of sodium ions on the electrical activity of the giant axon of the squid. The Journal of physiology *108*, 37–77.

Hoffman D., Magee J., Colbert C., and D J. (1997). K+ channel regulation of signal propagation in dendrites of hippocampal pyramidal neurons. Nature *387*, 869–875.

Holtmaat A., and Caroni P. (2016). Functional and structural underpinnings of neuronal assembly formation in learning. Nature neuroscience *19*, 1553.

Hopfield J.J. (1982). Neural networks and physical systems with emergent collective computational abilities. Proceedings of the national academy of sciences *79*, 2554–2558.

Hopfield J.J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. Proceedings of the national academy of sciences *81*, 3088–3092.

Howard M.W., and Kahana M.J. (2002). A distributed representation of temporal context. Journal of Mathematical Psychology *46*, 269–299.

Hsu C.L., Cembrowski M.S., Mensh B.D., and Spruston N. (2015). Dendritic sodium spikes are required for long-term potentiation at distal synapses on hippocampal pyramidal neurons. eLife *10.7554*, eLife.06414.

Hubel D.H., and Wiesel T.N. (1959). Receptive fields of single neurones in the cat's striate cortex. The Journal of physiology *148*, 574–591.

Hubel D.H., and Wiesel T.N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of physiology *160*, 106–154.

Hutcheon B., and Yarom Y. (2000). Resonance, oscillation and the intrinsic frequency preferences of neurons. Trends in neurosciences *23*, 216–222.

ICGenealogy (2017). Icgenealogy website. [Online; accessed 1-Oct-2017: http://icg.neurotheory.ox.ac.uk].

Isaacson J.S., and Scanziani M. (2011). How inhibition shapes cortical activity. Neuron *72*, 231–243.

Ito M. (1972). Neural design of the cerebellar motor control system. Brain research *40*, 81–84.

Jahnsen H., and Llinas R. (1984). Ionic basis for the electro-responsiveness and oscillatory properties of guinea-pig thalamic neurones in vitro. The Journal of physiology *349*, 227–247.

Josselyn S.A., Köhler S., and Frankland P.W. (2015). Finding the engram. Nature Reviews Neuroscience *16*, 521.

Jun J.J., Steinmetz N.A., Siegle J.H., Denman D.J., Bauza M., Barbarits B., Lee A.K., Anastassiou C.A., Andrei A., Aydın Ç., et al. (2017). Fully integrated silicon probes for high-density recording of neural activity. Nature *551*, 232.

Kandel E.R., Schwartz J.H., and Jessell T.M. (2000). Principles of Neural Science (McGraw-Hill Medical), 4th edn.

Kase D., and Imoto K. (2012). The Role of HCN Channels on Membrane Excitability in the Nervous System. Journal of signal transduction *2012*, 619747.

Khan A.G., and Hofer S.B. (2018). Contextual signals in visual cortex. Current opinion in neurobiology *52*, 131–138.

Kobak D., Brendel W., Constantinidis C., Feierstein C.E., Kepecs A., Mainen Z.F., Qi X.L., Romo R., Uchida N., and Machens C.K. (2016). Demixed principal component analysis of neural population data. Elife *5*, e10989.

Koch C. (2004). Biophysics of Computation. Information Processing in Single Neurons (Oxford University Press).

Koch C., and Segev I. (1998). Methods in neuronal modeling: from ions to networks (MIT press).

Koch C., and Segev I. (2000). The role of single neurons in information processing. Nature Neuroscience *3*, 1171–1177.

Krzywinski M., Schein J., Birol I., Connors J., Gascoyne R., Horsman D., Jones S.J., and Marra M.A. (2009). Circos: An information aesthetic for comparative genomics. Genome Research *19*, 1639–1645.

Kuchibhotla K.V., Gill J.V., Lindsay G.W., Papadoyannis E.S., Field R.E., Sten T.A.H., Miller K.D., and Froemke R.C. (2017). Parallel processing by cortical inhibition enables context-dependent behavior. Nature neuroscience *20*, 62.

Lai H.C., and Jan L.Y. (2006). The distribution and targeting of neuronal voltage-gated ion channels. Nature Reviews Neuroscience *7*, 548–562.

Laje R., and Buonomano D.V. (2013). Robust timing and motor patterns by taming chaos in recurrent neural networks. Nature neuroscience *16*, 925.

Lancaster B., and Adams P. (1986). Calcium-dependent current generating the afterhyperpolarization of hippocampal neurons. Journal of Neurophysiology *55*, 1268–1282.

Lapique L. (1907). Recherches quantitatives sur l'excitation electrique des nerfs traitee comme une polarization. Journal of Physiology and Pathololgy *9*, 620–635.

Lehmann-Horn F., and Jurkat-Rott K. (1999). Voltage-gated ion channels and hereditary disease. Physiological reviews *75*, 1317–1372.

Lengyel M., and Dayan P. (2008). Hippocampal contributions to control: the third way. In Advances in neural information processing systems. pp. 889–896.

Letzkus J.J., Wolff S.B., and Lüthi A. (2015). Disinhibition, a circuit mechanism for associative learning and memory. Neuron *88*, 264–276.

Leutgeb J.K., Leutgeb S., Moser M.B., and Moser E.I. (2007). Pattern separation in the dentate gyrus and ca3 of the hippocampus. science *315*, 961–966.

Levitan I.B. (1994). Modulation of ion channels by protein phosphorylation and dephosphorylation. Annual review of physiology *56*, 193–212.

Lewicki M.S. (1998). A review of methods for spike sorting: the detection and classification of neural action potentials. Network: Computation in Neural Systems *9*, R53–R78.

Li Q., Lee J.A., and Black D.L. (2007). Neuronal regulation of alternative pre-mRNA splicing. Nature reviews. Neuroscience *8*, 819–831.

Lim S., and Goldman M.S. (2012). Noise tolerance of attractor and feedforward memory models. Neural computation *24*, 332–390.

Lim S., and Goldman M.S. (2013). Balanced cortical microcircuitry for maintaining information in working memory. Nature neuroscience *16*, 1306.

Lipscombe D., Helton T.D., and Xu W. (2004). L-type calcium channels: the low down. Journal of neurophysiology *92*, 2633–2641.

Little W.A. (1974). The existence of persistent states in the brain. In From High-Temperature Superconductivity to Microminiature Refrigeration (Springer), pp. 145–164.

Lueckmann J.M., Goncalves P.J., Bassetto G., Öcal K., Nonnenmacher M., and Macke J.H. (2017). Flexible statistical inference for mechanistic models of neural dynamics. In Advances in Neural Information Processing Systems. pp. 1289–1299.

Lynch G.S., Dunwiddie T., and Gribkoff V. (1977). Heterosynaptic depression: a postsynaptic correlate of long-term potentiation. Nature *266*, 737.

Maass W., Natschläger T., and Markram H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. Neural computation *14*, 2531–2560.

Mainen Z.F., and Sejnowski T.J. (1995). Reliability of Spike Timing in Neocortical Neurons. Science *268*, 1503–1506.

Mante V., Sussillo D., Shenoy K.V., and Newsome W.T. (2013). Context-dependent computation by recurrent dynamics in prefrontal cortex. nature *503*, 78.

Marder E., and Goaillard J.M. (2006). Variability, compensation and homeostasis in neuron and network function. Nature reviews. Neuroscience *7*, 563–74.

Mari C.F., and Treves A. (1998). Modeling neocortical areas with a modular neural network. Biosystems *48*, 47–55.

Markram H., Lübke J., Frotscher M., and Sakmann B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps. Science *275*, 213–215.

Markram H., Muller E., Ramaswamy S., Reimann M.W., Abdellah M., Sanchez C.A., Ailamaki A., Alonso-Nanclares L., Antille N., Arsever S., et al. (2015). Reconstruction and simulation of neocortical microcircuitry. Cell *163*, 456–492.

Marques T., Nguyen J., Fioreze G., and Petreanu L. (2018). The functional organization of cortical feedback inputs to primary visual cortex. Nature neuroscience *21*, 757.

Masse N.Y., Grant G.D., and Freedman D.J. (2018). Alleviating catastrophic forgetting using context-dependent gating and synaptic stabilization. arXiv preprint arXiv:1802.01569 .

McClelland J.L., McNaughton B.L., and O'reilly R.C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. Psychological review *102*, 419.

McCormick D.A., and Pape H.C. (1990). Properties of a hyperpolarization-activated cation current and its role in rhythmic oscillation in thalamic relay neurones. The Journal of physiology *431*, 291–318.

McCulloch W.S., and Pitts W. (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics *5*, 115–133.

McDougal R.A., Bulanova A.S., and Lytton W.W. (2016). Reproducibility in computational neuroscience models and simulations. IEEE Transactions on Biomedical Engineering *63*, 2021–2035.

Meshulam L., Gauthier J.L., Brody C.D., Tank D.W., and Bialek W. (2017). Collective behavior of place and non-place neurons in the hippocampal network. Neuron *96*, 1178–1191.

Migliore M., Hoffman D.A., and Magee J.C. (1999). Role of an A-type K+ conductance in the back-propagation of action potentials in the dendrites of hippocampal pyramidal neurons. Journal of Computational Neuroscience *7*, 5–15.

Miller G.A. (1956). The magic number seven plus or minus two: Some limits on our capacity for processing information. Psychological review *63*, 91–97.

Milligan G.W., and Cooper M.C. (1985). An examination of procedures for determining the number of clusters in a data set. Psychometrika *50*, 159–179.

Milner P.M. (1996). Neural representations: some old problems revisited. Journal of Cognitive Neuroscience *8*, 69–77.

Misonou H., Mohapatra D.P., Park E.W., Leung V., Zhen D., Misonou K., Anderson A.E., and Trimmer J.S. (2004). Regulation of ion channel localization and phosphorylation by neuronal activity. Nature neuroscience *7*, 711–718.

Moore C.I., and Nelson S.B. (1998). Spatio-temporal subthreshold receptive fields in the vibrissa representation of rat primary somatosensory cortex. Journal of Neurophysiology *80*, 2882–2892.

Mountcastle V.B. (1957). Modality and topographic properties of single neurons of cat's somatic sensory cortex. Journal of neurophysiology *20*, 408–434.

Murphy B.K., and Miller K.D. (2009). Balanced amplification: a new mechanism of selective amplification of neural activity patterns. Neuron *61*, 635–648.

Napper R., and Harvey R. (1988). Number of parallel fiber synapses on an individual purkinje cell in the cerebellum of the rat. Journal of Comparative Neurology *274*, 168–177.

Neher E., and Sakaba T. (2008). Multiple roles of calcium ions in the regulation of neurotransmitter release. Neuron *59*, 861–72.

Neher E., and Sakmann B. (1976). Single-channel currents recorded from membrane of denervated frog muscle fibres. Nature *260*, 799.

Nichols C.G., and Lopatin A.N. (1997). Inward rectifier potassium channels. Annual review of physiology *59*, 171–91.

Norman Y., Yeagle E.M., Harel M., Mehta A.D., and Malach R. (2017). Neuronal baseline shifts underlying boundary setting during free recall. Nature Communications *8*, 1301.

O'Doherty J., Dayan P., Schultz J., Deichmann R., Friston K., and Dolan R.J. (2004). Dissociable roles of ventral and dorsal striatum in instrumental conditioning. science *304*, 452–454.

Okada M. (1996). Notions of associative memory and sparse coding. Neural Networks *9*, 1429–1458.

O'Keefe J., and Dostrovsky J. (1971). The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. Brain research .

O'Leary T., Williams A.H., Franci A., and Marder E. (2014). Cell Types, Network Homeostasis, and Pathological Compensation from a Biologically Plausible Ion Channel Expression Model. Neuron *82*, 809–821.

Ozeki H., Finn I.M., Schaffer E.S., Miller K.D., and Ferster D. (2009). Inhibitory stabilization of the cortical network underlies visual surround suppression. Neuron *62*, 578–592.

Pandarinath C., O'Shea D.J., Collins J., Jozefowicz R., Stavisky S.D., Kao J.C., Trautmann E.M., Kaufman M.T., Ryu S.I., Hochberg L.R., et al. (2018). Inferring single-trial neural population dynamics using sequential auto-encoders. Nature methods p. 1.

Pape L., Gomez F., Ring M., and Schmidhuber J. (2011). Modular deep belief networks that do not forget. In Neural Networks (IJCNN), The 2011 International Joint Conference on (IEEE), pp. 1191–1198.

Pavlov P.I. (2010). Conditioned reflexes: an investigation of the physiological activity of the cerebral cortex. Annals of neurosciences *17*, 136.

Pawson A.J., Sharman J.L., and Benson H.E. (2014). The IUPHAR/BPS Guide to PHARMACOLOGY: an expert-driven knowledgebase of drug targets and their ligands. Nucl. Acids Res. *42*, D1098–106.

Peretto P. (1988). On learning rules and memory storage abilities of asymmetrical neural networks. Journal de Physique *49*, 711–726.

Perez-Vicente C. (1989). Hierarchical neural network with high storage capacity. Physical Review A *40*, 5356.

Perumal S., and Minai A.A. (2009). Stable-yet-switchable (sys) attractor networks. In Neural Networks, 2009. IJCNN 2009. International Joint Conference on (IEEE), pp. 2509–2516.

Pfeiffer B.E., and Foster D.J. (2015). Autoassociative dynamics in the generation of sequences of hippocampal place cells. Science *349*, 180–183.

Plebe A. (2018). The search of "canonical" explanations for the cerebral cortex. History and philosophy of the life sciences *40*, 40.

Podlaski W.F., Seeholzer A., Groschner L.N., Miesenböck G., Ranjan R., and Vogels T.P. (2017). Mapping the function of neuronal ion channels in model and experiment. Elife *6*, e22152.

Poirazi P., Brannon T., and Mel B.W. (2003). Pyramidal neuron as two-layer neural network. Neuron *37*, 989–999.

Poo M.m., Pignatelli M., Ryan T.J., Tonegawa S., Bonhoeffer T., Martin K.C., Rudenko A., Tsai L.H., Tsien R.W., Fishell G., et al. (2016). What is memory? the present state of the engram. BMC biology *14*, 40.

Press W.H., Teukolsky S.A., Vetterling W.T., and Flannery B.P. (2007). Numerical recipes 3rd edition: The art of scientific computing (Cambridge university press).

Prinz A.A., Billimoria C.P., and Marder E. (2003). Alternative to Hand-Tuning Conductance-Based Models: Construction and Analysis of Databases of Model Neurons. Journal of Neurophysiology *90*, 3998–4015.

Prinz A.A., Bucher D., and Marder E. (2004). Similar network activity from disparate circuit parameters. Nature neuroscience *7*, 1345–52.

Purves D., Augustine G.J., Fitzpatrick D., Hall W., LaMantia A., McNamara J., and White L. (2014). Neuroscience, 2008. De Boeck, Sinauer, Sunderland, Mass .

Quirk G.J., and Mueller D. (2008). Neural mechanisms of extinction learning and retrieval. Neuropsychopharmacology *33*, 56.

Rabinovich M.I., Varona P., Selverston A.I., and Abarbanel H.D. (2006). Dynamical principles in neuroscience. Reviews of modern physics *78*, 1213.

Rajan K., Abbott L., and Sompolinsky H. (2010). Stimulus-dependent suppression of chaos in recurrent neural networks. Physical Review E *82*, 011903.

Ramsay J., and Silverman B. (2005). Functional data analysis (Springer, New York), 2 edn.

Ranjan R., Khazen G., Gambazzi L., Ramaswamy S., Hill S.L., Schürmann F., and Markram H. (2011). Channelpedia: an integrative and interactive database for ion channels. Frontiers in Neuroinformatics *5*, 36.

Renart A., Parga N., and Rolls E.T. (1999). Associative memory properties of multiple cortical modules. Network: Computation in Neural Systems *10*, 237–255.

Rescorla R.A., Wagner A.R., et al. (1972). A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. Classical conditioning II: Current research and theory *2*, 64–99.

Reynolds J.H., and Chelazzi L. (2004). Attentional modulation of visual processing. Annu. Rev. Neurosci. *27*, 611–647.

Riesenhuber M., and Poggio T. (1999). Hierarchical models of object recognition in cortex. Nature neuroscience *2*, 1019.

Rigotti M., Barak O., Warden M.R., Wang X.J., Daw N.D., Miller E.K., and Fusi S. (2013). The importance of mixed selectivity in complex cognitive tasks. Nature *497*, 585.

Robertson B., Eacott M., and Easton A. (2015). Putting memory in context: Dissociating memories by distinguishing the nature of context. Behavioural brain research *285*, 99–104.

Romani S., Pinkoviezky I., Rubin A., and Tsodyks M. (2013). Scaling laws of associative memory retrieval. Neural computation *25*, 2523–2544.

Rosenblatt F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review *65*, 386.

Ross S.M. (2014). Introduction to probability models (Academic press).

Rousseeuw P.J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics *20*, 53–65.

Rubin R., Abbott L., and Sompolinsky H. (2017). Balanced excitation and inhibition are required for high-capacity, noise-robust neuronal selectivity. Proceedings of the National Academy of Sciences p. 201705841.

Rumelhart D.E., and McClelland J.L. (1986). Parallel distributed processing: explorations in the microstructure of cognition. volume 1. foundations (MIT Press, Cambridge, Ma).

Sacramento J., Costa R.P., Bengio Y., and Senn W. (2017). Dendritic error backpropagation in deep cortical microcircuits. arXiv preprint arXiv:1801.00062 .

Saleem A., Diamanti E., Fournier J., Harris K., and Carandini M. (2018). Coherent encoding of subjective spatial position in visual cortex and hippocampus. Nature .

Sanderson C., and Curtin R. (2016). Armadillo: a template-based c++ library for linear algebra. Journal of Open Source Software .

Sankey H. (1898). Introductory note on the thermal efficiency of steam-engines. In Minutes of Proceedings of The Institution of Civil Engineers. pp. 278–283.

Schmidt M. (2008). The sankey diagram in energy and material flow management. Journal of Industrial Ecology *12*, 82–94.

Schoenbaum G., Roesch M.R., Stalnaker T.A., and Takahashi Y.K. (2009). A new perspective on the role of the orbitofrontal cortex in adaptive behaviour. Nature Reviews Neuroscience *10*, 885.

Schultz W., Dayan P., and Montague P.R. (1997). A neural substrate of prediction and reward. Science *275*, 1593–1599.

Schulz D.J., Temporal S., Barry D.M., and Garcia M.L. (2008). Mechanisms of voltage-gated ion channel regulation: from gene expression to localization. Cellular and molecular life sciences : CMLS *65*, 2215–31.

Scoville W.B., and Milner B. (1957). Loss of recent memory after bilateral hippocampal lesions. Journal of neurology, neurosurgery, and psychiatry *20*, 11.

Sederberg P.B., Howard M.W., and Kahana M.J. (2008). A context-based theory of recency and contiguity in free recall. Psychological review *115*, 893.

Seung H.S. (1996). How the brain keeps the eyes still. Proceedings of the National Academy of Sciences *93*, 13339–13344.

Shadlen M.N., and Newsome W.T. (1994). Noise, neural codes and cortical organization. Current opinion in neurobiology *4*, 569–579.

Shah M.M., Hammond R.S., and Hoffman D.A. (2010). Dendritic ion channel trafficking and plasticity. Trends in neurosciences *33*, 307–316.

Shapley R., Hawken M., and Xing D. (2007). The dynamics of visual responses in the primary visual cortex. Progress in brain research *165*, 21–32.

Shatz C.J. (1992). The developing brain. Scientific American *267*, 60–67.

Shenoy K.V., Sahani M., and Churchland M.M. (2013). Cortical control of arm movements: a dynamical systems perspective. Annual review of neuroscience *36*.

Shepherd G.M. (2015). Foundations of the neuron doctrine (Oxford University Press).

Shin L.M., and Liberzon I. (2010). The neurocircuitry of fear, stress, and anxiety disorders. Neuropsychopharmacology *35*, 169.

Shu Y., Hasenstaub A., and McCormick D.A. (2003). Turning on and off recurrent balanced cortical activity. Nature *423*, 288.

Shulz D.E., Sosnik R., Ego V., Haidarliu S., and Ahissar E. (2000). A neuronal analogue of state-dependent learning. Nature *403*, 549.

Sigg D. (2014). Modeling ion channels: Past, present, and future. The Journal of general physiology *144*, 7–26.

Sjöström P.J., and Nelson S.B. (2002). Spike timing, calcium signals and synaptic plasticity. Current opinion in neurobiology *12*, 305–313.

Smith D.M., and Mizumori S.J. (2006). Hippocampal place cells, context, and episodic memory. Hippocampus *16*, 716–729.

Smith S.M. (1979). Remembering in and out of context. Journal of Experimental Psychology: Human Learning and Memory *5*, 460.

Sompolinsky H. (1987). The theory of neural networks: The hebb rule and beyond. In Heidelberg colloquium on glassy dynamics (Springer), pp. 485–527.

Stefanini F., Kheirbek M., Kushnir L., Jennings J.H., Stuber G., Hen R., and Fusi S. (2018). A distributed neural code in ensembles of dentate gyrus granule cells. bioRxiv p. 292953.

Steven H.S. (1994). Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering.

Stevens C.F. (1978). Interactions between intrinsic membrane protein and electric field. an approach to studying nerve excitability. Biophysical journal *22*, 295–306.

Stuart G., and Spruston N. (1998). Determinants of voltage attenuation in neocortical pyramidal neuron dendrites. The Journal of neuroscience *18*, 3501–3510.

Sussillo D. (2014). Neural circuits as computational dynamical systems. Current opinion in neurobiology *25*, 156–163.

Sussillo D., and Abbott L.F. (2009). Generating coherent patterns of activity from chaotic neural networks. Neuron *63*, 544–557.

Thompson R.F., Berger T.W., and Madden IV J. (1983). Cellular processes of learning and memory in the mammalian cns. Annual Review of Neuroscience *6*, 447–491.

Thompson R.F., and Kim J.J. (1996). Memory systems in the brain and localization of a memory. Proceedings of the national academy of sciences *93*, 13438–13444.

Thorpe S., Delorme A., and Van Rullen R. (2001). Spike-based strategies for rapid processing. Neural networks *14*, 715–725.

Titley H.K., Brunel N., and Hansel C. (2017). Toward a neurocentric view of learning. Neuron *95*, 19–32.

Tolman E.C. (1951). Purposive behavior in animals and men (Univ of California Press).

Traub R.D., Buhl E.H., Gloveli T., and Whittington M.a. (2003). Fast rhythmic bursting can be induced in layer 2/3 cortical neurons by enhancing persistent Na+ conductance or by blocking BK channels. Journal of neurophysiology *89*, 909–921.

Tremblay R., Lee S., and Rudy B. (2016). Gabaergic interneurons in the neocortex: from cellular properties to circuits. Neuron *91*, 260–292.

Treves A., and Rolls E.T. (1991). What determines the capacity of autoassociative memories in the brain? Network: Computation in Neural Systems *2*, 371–397.

Trimmer J.S., and Rhodes K.J. (2004). Localization of voltage-gated ion channels in mammalian brain. Annu Rev Physiol *66*, 477–519.

Tripathy S.J., Savitskaya J., Burton S.D., Urban N.N., and Gerkin R.C. (2014). NeuroElectro: a window to the world's neuron electrophysiology data. Frontiers in Neuroinformatics *8*.

Tsien R., and Noble D. (1969). A transition state theory approach to the kinetics of conductance changes in excitable membranes. The Journal of membrane biology *1*, 248–273.

Tsodyks M. (1990). Hierarchical associative memory in neural networks with low activity level. Modern Physics Letters B *4*, 259–265.

Tsodyks M., and Feigelman M. (1988). The enhanced storage capacity in neural networks with low activity level. EPL (Europhysics Letters) *6*, 101.

Tulving E. (2002). Episodic memory: From mind to brain. Annual review of psychology *53*, 1–25.

Tulving E., et al. (1972). Episodic and semantic memory. Organization of memory *1*, 381–403.

Turrigiano G. (2012). Homeostatic synaptic plasticity: local and global mechanisms for stabilizing neuronal function. Cold Spring Harbor perspectives in biology *4*, a005736.

Vacher H., Mohapatra D.P., and Trimmer J.S. (2008). Localization and targeting of voltage-dependent ion channels in mammalian central neurons. Physiological reviews *88*, 1407–1447.

Van Vreeswijk C., and Sompolinsky H. (1996). Chaos in neuronal networks with balanced excitatory and inhibitory activity. Science *274*, 1724–1726.

Vandenberg C., and Bezanilla F. (1991). A sodium channel gating model based on single channel, macroscopic ionic, and gating currents in the squid giant axon. Biophysical journal *60*, 1511–1533.

Vogels T.P., and Abbott L.F. (2005). Signal propagation and logic gating in networks of integrate-and-fire neurons. Journal of neuroscience *25*, 10786–10795.

Wallace D.J., and Kerr J.N. (2010). Chasing the cell assembly. Current opinion in neurobiology *20*, 296–305.

Wang X.J. (2002). Probabilistic decision making by slow reverberation in cortical circuits. Neuron *36*, 955–968.

Ward J.H. (1963). Hierarchical grouping to optimize an obective function. Journal of the American statistical association *58*, 236–244.

Wehr M., and Zador A.M. (2003). Balanced inhibition underlies tuning and sharpens spike timing in auditory cortex. Nature *426*, 442.

Willms A.R., Baro D.J., Harris-Warrick R.M., and Guckenheimer J. (1999). An improved parameter estimation method for Hodgkin-Huxley models. Journal of Computational Neuroscience *6*, 145–168.

Willshaw D.J., Buneman O.P., and Longuet-Higgins H.C. (1969). Non-holographic associative memory. Nature .

Wilson H.R., and Cowan J.D. (1973). A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. Kybernetik *13*, 55–80.

Wimmer K., Nykamp D.Q., Constantinidis C., and Compte A. (2014). Bump attractor dynamics in prefrontal cortex explains behavioral precision in spatial working memory. Nature neuroscience *17*, 431.

Wood K.C., Blackwell J.M., and Geffen M.N. (2017). Cortical inhibitory interneurons control sensory processing. Current opinion in neurobiology *46*, 200–207.

Xu C., Krabbe S., Gründemann J., Botta P., Fadok J.P., Osakada F., Saur D., Grewe B.F., Schnitzer M.J., Callaway E.M., et al. (2016). Distinct hippocampal pathways mediate dissociable roles of context in memory retrieval. Cell *167*, 961–972.

Yamins D.L., and DiCarlo J.J. (2016). Using goal-driven deep learning models to understand sensory cortex. Nature neuroscience *19*, 356.

Yang G.R., Murray J.D., and Wang X.J. (2016). A dendritic disinhibitory circuit mechanism for pathway-specific gating. Nature communications *7*, 12815.

Yu F., and Catterall W. (2004). The VGL chanome: a protein superfamily specialized for electrical signaling and ionic homeostasis. Science Signaling *2004*, re15.

Yuste R. (2015). From the neuron doctrine to neural networks. Nature reviews neuroscience *16*, 487.

Zenke F., Agnes E.J., and Gerstner W. (2015). Diverse synaptic plasticity mechanisms orchestrated to form and retrieve memories in spiking neural networks. Nature communications *6*, 6922.

Zola-Morgan S. (1995). Localization of brain function: The legacy of franz joseph gall (1758-1828). Annual review of neuroscience *18*, 359–383.

Zylberberg J. (2017). Untuned but not irrelevant: A role for untuned neurons in sensory information coding. bioRxiv p. 134379.

Zylberberg J., and Strowbridge B.W. (2017). Mechanisms of persistent activity in cortical circuits: possible neural substrates for working memory. Annual review of neuroscience *40*, 603–627.