Examining neuronal connectivity and its role in learning and memory

by Rohan Gala

B. Tech. and M. Tech. in Engineering Physics, Indian Institute of Technology Bombay

A dissertation submitted to

The Faculty of
the College of Science of
Northeastern University
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

August 3, 2017

Dissertation directed by
Armen Stepanyants
Associate Professor of Physics
Acknowledgments

I feel privileged to have spent much of my adult life learning and thinking about the mysteries of the world. I am deeply indebted to my advisor Armen Stepanyants for sharing his knowledge and nurturing my naïve curiosity with immense patience and humanity; past mentors Punit Parmananda, Pradeep Sarin and Praveen Tyagi for encouraging and guiding me at important junctures in my career thus far.

I would like to thank lab members Julio Chapeton, Seyed Mostafa, Shih-Luen Wang, Chi Zhang and Danke Zhang for fostering a relaxed but stimulating work environment; collaborators Anthony Holtmaat and Daniel Lebrecht for hosting me in Geneva and for giving me the opportunity to work with experimental data; staff members at the Physics department and College of Science for simplifying real-world concerns.

My time in graduate school was enriched by some very special people. I would like to thank all physics graduate students, especially Hasnain Hafiz, Alex Hyde, Nabuan Naufer, and Younggil Song for camaraderie at the department; Palvannan Bharathwaj, Pranav Joshi, Deepak Kurian, Shreyas Potnis, Mayank Sanganeria, and Tanuj Saxena for reality checks, philosophical discussions, road trips and mahasammelans; Vinesh Mandot, Ravi Thakkar, and Upasana Shah for putting up with me over nearly two decades now.

Boston feels like home now because of Leah Marie Harrigan, who has indulged me with love and unconditional support, and because of the warmth extended to me by the Harrigan family.

Finally, I would like to thank my parents, Ranjan and Dinesh, and brother Hitesh for shaping much of what I am today; for affording me trust, freedom, and inspiration to define my own purpose in life.
Abstract of Dissertation

Learning and long-term memory formation are accompanied with changes in the patterns and weights of synaptic connections in the underlying neuronal network. However, the fundamental rules that drive connectivity changes, and the precise structure-function relationships within neuronal networks remain elusive. Technological improvements over the last few decades have enabled the observation of large but specific subsets of neurons and their connections in unprecedented detail. Devising robust and automated computational methods is critical to distill information from ever-increasing volumes of raw experimental data. Moreover, statistical models and theoretical frameworks are required to interpret the data and assemble evidence into understanding of brain function. In this thesis, I first describe computational methods to reconstruct connectivity based on light microscopy imaging experiments. Next, I use these methods to quantify structural changes in connectivity based on \textit{in vivo} time-lapse imaging experiments. Finally, I present a theoretical model of associative learning that can explain many stereotypical features of experimentally observed connectivity.
Table of Contents

Acknowledgments ii

Abstract of dissertation iii

Introduction 10

Chapter 1. Active learning of neuron morphology for accurate automated tracing of neurites 17
  Background 17
  Initial trace of neurites 20
  Optimization of the initial trace 24
  Learning branching morphology of neurites 26
  Active learning strategy 33
  Evaluating the tracing strategy 34
  Comparison of automated initial traces and manual user traces 35
  Active learning of branching morphology of neurites 39
  Comparisons with other automated tracing tools 42
  Discussion 44

Chapter 2. Computer assisted detection of axonal bouton structural plasticity in in vivo time-lapse images 47
  Background 47
  In vivo imaging 50
  Correlative 3D EM 52
  LM-based bouton detection and measurement of structural changes 52
  Software implementation 59
  Validation of LM-based bouton detection methodology with EM 60
Effects of imaging conditions on bouton analysis 63
Statistical framework for the analysis of structural plasticity of boutons 66
Bouton changes can be resolved despite the uncertainty in measurements 69
Discussion 71

Chapter 3. Anatomical correlates of perceptual learning 74
Background 74
Experiment design 76
Quantifying behavioral performance 77
Dataset of bouton structural dynamics 78
Bouton number and weight are homeostatically regulated 80
Temporal and spatial correlations in bouton structural changes 82
Bouton addition and elimination across animals 84
Discussion 86

Chapter 4. Effects of homeostatic constraints on statistical features of synaptic connectivity 89
Background 89
Biologically constrained model of steady-state learning 93
Model assumptions and approximations 97
Theoretical solution of the model 101
Model parameters 102
Numerical solutions of the model 104
Fitting distributions of connection weights 106
Effects of homeostatic constraints on network capacity and connectivity 107
Distribution of connection weights 111
Comparison of experimental and theoretical connection weight distributions 113
List of Figures

Figure 1.1 Automated traces of neurites may contain hidden topological errors 19
Figure 1.2 The initial trace can be based on the solution of the Eikonal equation 22
Figure 1.3 Morphological features for classification of branch merging scenarios 27
Figure 1.4 Branch merging scenarios 29
Figure 1.5 How to choose best classification parameters 31
Figure 1.6 Assessing the quality of automated traces 36
Figure 1.7 Assessment of initial traces of multiple neuron axons 38
Figure 1.8: Online training can be used to reduce the numbers of topological errors present in initial traces 41
Figure 2.1 Challenges in LM-based bouton detection and measurement 49
Figure 2.2 Detection of putative boutons as peaks in axon intensity profiles 54
Figure 2.3: Correlative light and electron microscopy 60
Figure 2.4 Normalized intensity profile is correlated with axon cross-section area, while bouton weight is indicative of bouton volume 62
Figure 2.5 Probabilistic definition of an LM bouton based on measurement uncertainty derived from short-term imaging experiments 64
Figure 2.6 Structural change in boutons can be resolved in long-term in vivo imaging experiments despite measurement noise 70
Figure 3.1 Timelines for different phases of the experiment in 5 animals 77
Figure 3.2 Matching boutons in time-lapse images 79
Figure 3.3 Homeostasis in bouton number and bouton weight 81
Figure 3.4 Bouton weight changes are anti-correlated at 4 day lag 82
Figure 3.5 Bouton structures along the same axon are correlated 83
Figure 3.6 Fraction of boutons added to the circuit in different time intervals 84
Figure 3.7 Fraction of boutons eliminated from the circuit in different time intervals 86
Figure 4.1 Associative memory storage in local cortical circuits 90
Figure 4.2 Validation of theoretical results with numerical simulations 105
Figure 4.3 A-C. Critical capacity and overall connection probability and overall connection weight as for different model parameters 108
Figure 4.4 Effects of I0 and I1 constraints on critical capacity and connection probability 109
Figure 4.5 Comparison of theoretical distributions of connection weights to numerical simulations 112
Figure 4.6 Comparison of theoretical and experimental distributions of connection weights 115
Figure A.1 Optimization is required for trace-based bouton detection and measurement 124
Figure A.2 Trace optimization reduces variability in bouton detection and weight measurement 126
Figure B.1 Compatible image stack file formats and naming conventions 129
Figure B.2 Naming convention for axon traces 130
Figure B.3 BoutonAnalyzer main window 131
Figure B.4 Optimize Trace & Generate Profile GUI 132
Figure B.5 Detect & Track Boutons GUI 134
List of Tables

Table 1.1: Comparisons of manual and automated traces. 43
Table 2.1: Comparison of LM-based and EM measurements. 68
Table 3.1: Dataset summary. 80
Table B.1: List of BoutonAnalyzer shortcuts 136
Introduction

Towards the late 19th century, Santiago Ramón y Cajal utilized the newly discovered Golgi stain to study the nervous system (Golgi, 1873; Ramón y Cajal, 1888). The sparse labeling achieved by this stain enabled Cajal to observe neurons and their arbors in isolation and, thus, in much detail. This led him to unequivocally recognize neurons as distinct entities in the brain, and to predict many functional properties of their arbors based on structural observations alone (DeFelipe, 2006; Llinas, 2003; Shepherd, 2016). The idea that structural properties of neurons can reveal insight about their functional roles has continued to shape modern approaches in neuroscience.

We now know that there is incredible diversity of neurons in the mammalian brain in terms of their morphology, gene expression profiles and electrophysiological properties. Obtaining a catalogue of distinct neuron types can help us understand the functional contributions of these populations to the circuits they constitute (DeFelipe et al., 2013; Harris and Shepherd, 2015; Masland, 2004). Advances in genetic technology (Feng et al., 2000) and light microscopy (Huang and Zeng, 2013; Lichtman and Denk, 2011; Luo et al., 2008; Wilt et al., 2009) have made it possible to study the detailed morphology of genetically defined neuron sub-populations. Neurons can be labeled with fluorescent proteins or dyes and subsequently imaged with sub-micrometer resolution light microscopy. Obtaining the 3D layout of labeled neurites, i.e. axonal and dendritic arbors, is a critical step in identifying different morphological neuron types based on such imaging experiments (Armananzas and Ascoli, 2015; Costa et al., 2016; Meijering, 2010; Svoboda, 2011; Wan et al., 2015). Since physical proximity of an axon of one neuron and a dendrite of another is a necessary condition for a functional connection to exist between the neurons, the layout of neurites can also be used to infer statistical patterns of connectivity (Hill et al., 2012; Reimann et al., 2017; Stepanyants et al., 2002; Stepanyants et al., 2004). Moreover, the morphology of axons
and dendrites influences the properties of electrical signals they carry (Hausser et al., 2000). For these reasons, knowledge of the layout of neurites is an essential component of computational modeling efforts at single neuron as well as network levels (Hines and Carnevale, 1997; Markram et al., 2015).

A juxtaposition of an axon of one neuron and a dendrite of another can develop into a connection between the neurons known as a synapse. Characteristic electrical signals called action potentials generated by one neuron travel along its axon, and are transmitted to the dendrite of another neuron via synapses. A synapse is composed of two distinct structures, presynaptic structure on the axon and postsynaptic structure on the dendrite. Presynaptic structures generally appear as swellings along the axon (boutons) which are visible in light microscopy images (Hellwig et al., 1994; Somogyi, 1978). In electron microscopy images, presynaptic structures can be identified based on their shape, presence of synaptic vesicles, and presence of active zones (Dufour et al., 2016; Gray, 1959; Harris and Weinberg, 2012; Shepherd and Harris, 1998). Postsynaptic structures are located on the dendritic shaft or on protrusions (spines) arising from the dendritic shaft which are visible in light microscopy (Ramón y Cajal, 1888). Postsynaptic structures may be identified in electron microscopy images based on presence of postsynaptic density (Harris and Weinberg, 2012; Shepherd and Harris, 1998) (see also (Burette et al., 2015) for a discussion regarding imaging-based definitions of synapses).

Neural circuits continually remodel through addition, elimination, and changes in the efficacy of synaptic connections. Such remodeling events are reflected in the appearance, disappearance and modifications to the pre- and postsynaptic structures. It has become clear that synaptic structural changes depend on experience of the animal and are of fundamental importance for learning and memory (Bailey and Kandel, 1993; Barnes and Finnerty, 2010; Caroni et al., 2012; Chklovskii et
al., 2004; Holtmaat and Caroni, 2016; Holtmaat and Svoboda, 2009). One way to unravel these relationships is to monitor structural modifications in brains of behaving animals. Presently, light microscopy is the only feasible way to image neurons at high enough spatial resolution to observe changes in synaptic structures in vivo. Multi-photon microscopy (Denk et al., 1990; Denk and Svoboda, 1997; Helmchen and Denk, 2005) is now routinely used to inspect the structural dynamics of fluorescently labeled neurons and their synapses in the mammalian cortex (Svoboda and Yasuda, 2006). Optical access to the brain is achieved by surgically implanting a cranial window (Holtmaat et al., 2009), thinning the skull to an extent that renders it transparent (Drew et al., 2010; Yang et al., 2010), or with a microendoscope implant (Barretto et al., 2011). The same volume of brain tissue can be imaged in a time-lapse manner over a period of many weeks or months. A single snapshot of brain volume obtained using such techniques (~1 × 1 × 0.3 mm$^3$) can result in several gigabytes of imaging data.

The physical dimensions of synaptic structures are comparable to the diffraction limit in a typical light microscopy experiment (~0.3 µm). Spines and boutons are usually defined and classified in images based on their apparent geometrical shape, or their intensity relative to a chosen reference, e.g. the average intensity of the neurite they belong to. The finite resolution in light microscopy experiments and uncertainty in measuring low number of photons impose fundamental limitations on the signal-to-noise ratio in resulting images. Furthermore, measurement of synaptic structures based on intensity in the images is hindered by technical complications related to chronic imaging over extended periods of time. Typically, each 3D image (stack) is manually annotated to identify synaptic structures and quantify their structural dynamics. As such experiments continue to mature and produce increasingly large imaging datasets, it is becoming critical to develop robust methods that can quantify structural changes efficiently and systematically (Canty and De Paola, 2011).
One of the central goals in neuroscience is to relate synaptic level changes to behavior of the animal. A common strategy in animals with relatively small nervous systems, such as *Caenorhabditis elegans, Aplysia californica,* and *Drosophila melanogaster,* is to identify highly specialized circuits involved in stereotypical, quantifiable behaviors. Neurons within these circuits can then be experimentally manipulated to evaluate how structural synaptic dynamics and neuronal activity relate to variations in behavior. Mammalian brains however, are more challenging to study due to the sheer number of synapses and neurons that are expected to be involved in generating even the simplest of behaviors. Moreover, information representation and processing are thought to occur in a sparse and distributed manner (Olshausen and Field, 2004; Romo and Salinas, 2003). This presents the problem of identifying populations of neurons and synapses that are large enough to be informative of behavior, yet small enough to be measurable with available experimental tools.

Studies in the 1990’s (Romo et al., 1998; Salzman et al., 1990) indicated that direct electrical microstimulation of cortical neurons could produce behaviorally relevant percepts in primates and rodents. Refinements of *in vivo* electrophysiological techniques made it possible to show that rats could be trained to report microstimulation of even single neurons in the somatosensory cortex (Houweling and Brecht, 2008). While these developments showed that animals could learn to identify and use information provided to only few neurons in the cortex to perform behavioral tasks (see (Doron and Brecht, 2015) for review), they did not explain how circuit changes facilitate learning.

The development of optogenetics, coupled with advances in genetic engineering techniques (Boyden et al., 2005; Packer et al., 2013), enabled encoding and expression of light gated ion channels in specific subpopulations of neurons. Optical microstimulation, i.e. shining light at
wavelengths that trigger opening of light gated ion channels, could then be used to evoke action potentials in neurons expressing these channels. Huber et al. (Huber et al., 2008) expressed green fluorescent protein (GFP) and a light gated ion channel, channelrhodopsin (ChR), simultaneously in a small population of mouse somatosensory cortex neurons. The animals were trained to associate optical microstimulation of neurons with water reward that was dispensed at one of two choice ports. With training, mice learned to perceive optical microstimulation and choose the correct port to obtain reward more frequently, demonstrating that stimulating a genetically defined population of neurons can drive learned behavior.

To understand how structural changes may be related to such perceptual learning, our collaborators\(^1\) periodically imaged axons of mouse somatosensory cortex layer 2/3 pyramidal neurons that were co-labeled with ChR and GFP, while training the mice to perform a task similar to the one in Huber et al. (Huber et al., 2008). Since only the labeled axons carry information necessary to perform the task, quantifying structural changes along these axons may provide insight into synapse-level rules that support perceptual learning.

While specific patterns of connectivity are continually changing as evidenced by many experiments (Chambers and Rumpel, 2017; Holtmaat and Svoboda, 2009), many statistical features of connectivity within local cortical networks remain conserved. For example, connectivity between excitatory pyramidal neuron pairs is sparse, inhibitory neurons tend to have much denser connectivity, and connection strength distributions have stereotypical shapes. Such features may have emerged to facilitate memory storage and can be evaluated using theoretical frameworks (Barbour et al., 2007). A general framework for the analysis of memory storage

\(^1\) D. Lebrecht\(^a\), M. Dimanico\(^a\), S. Pages\(^a\), K. Morandell\(^a\), D. Huber\(^a\) and A. Holtmaat\(^a\)

\(^a\) University of Geneva
capacity of networks of perceptron-like binary neurons was presented by Gardener et al. (Gardner, 1988; Gardner and Derrida, 1988). This framework has served as a starting point to incorporate important biological constraints into analytically solvable models to explain experimentally observed statistical features of connectivity (Brunel et al., 2004; Chapeton et al., 2012).

In chapter one of this thesis I describe an automated algorithm to trace neurites, with an emphasis on recovering the correct branching topology. The algorithm uses a machine learning approach in which topology is determined based on features such as distances between branches, branch orientations, intensities, calibers, and tortuosities. These features were devised to capture implicit considerations that guide decisions made by neuroanatomists while manually tracing neurites. We show that when the density of labeled neurites is sufficiently low, automated traces are not significantly different from manual reconstructions obtained by trained users. Work in this chapter was previously published in (Gala et al., 2014).

In chapter two, I discuss computational methodology for detection and measurement of structural changes in axonal boutons imaged in vivo with time-lapse two-photon laser scanning microscopy (2PLSM). The methodology was validated with correlative 3D electron microscopy of in vivo imaged axons, and by applying it to images acquired under various conditions that mimic challenges in long-term imaging experiments. We propose a statistical model to incorporate measurement uncertainty into quantitative evaluation of structural plasticity, and show that the model can be used to detect and quantify bouton changes in long-term imaging experiments. Results of this chapter have been submitted for publication (Gala et al., 2017).

In chapter three, I present results based on long-term imaging data from the perceptual learning experiment. We evaluate bouton structural changes in animals trained on a perceptual learning task
and discuss the observed spatial and temporal correlations. We find that animals that learned to perform the task have generally more plastic circuits than those that showed no improvement in performance. This research project is ongoing, and results of this chapter are unpublished.

In chapter four, I discuss a theoretical approach to evaluate the effects of homeostasis on statistical features of connectivity. We consider a recurrent network of perceptron-like neurons in a steady state, in which learning new associations requires forgetting old ones. To uncover connectivity features that characterize such steady-state circuits, we developed a biologically constrained, exactly solvable model of associative memory storage. The model is applicable to networks consisting of multiple neuron classes and can account for homeostatic constraints on the number and the overall weight of functional connections received by each neuron. Work in this chapter was published in (Chapeton et al., 2015).
Chapter 1. Active learning of neuron morphology for accurate automated tracing of neurites

Background

Our understanding of brain functions is hindered by the lack of detailed knowledge of synaptic connectivity in the underlying neural network. With current technology it is possible to sparsely label specific populations of neurons in vivo and image their processes with high-throughput optical microscopy [see e.g. (De Paola et al., 2006; Lichtman et al., 2008; Ragan et al., 2012; Stettler et al., 2002; Trachtenberg et al., 2002; Wickersham et al., 2007; Wilt et al., 2009)]. Imaging can be done in vivo for circuit development or plasticity studies (Trachtenberg et al., 2002), or ex vivo for circuit mapping projects (Lu et al., 2009). In the latter case, an unprecedented resolution can be achieved by first clarifying the tissue (Chung et al., 2013; Hama et al., 2011), followed by imaging the entire brain from thousands of optical sections (Ragan et al., 2012). The overwhelming obstacle remaining on the way to brain mapping is accurate, high-throughput tracing of neurons (Gillette et al., 2011b; Helmstaedter and Mitra, 2012; Kozloski, 2011; Lichtman and Denk, 2011; Lichtman et al., 2008; Liu, 2011; Miller, 2010; Perkel, 2013; Sporns et al., 2005; Svoboda, 2011; Van Essen and Ugurbil, 2012). Presently, accurate traces of complex neuron morphologies can only be obtained manually, which is extremely time consuming (Shepherd et al., 2005; Stepanyants et al., 2008; Stepanyants et al., 2004), and thus impractical for large reconstruction projects.

Many automated tracing algorithms have been developed in recent years [see e.g. (Al-Kofahi et al., 2002; Al-Kofahi et al., 2008; Bas and Erdogmus, 2011; Bas et al., 2012; Choromanska et al., 2012; Losavio et al., 2008; Peng et al., 2011; Peng et al., 2010; Schmitt et al., 2004; Srinivasan et al., 2010; Turetken et al., 2012; Turetken et al., 2011; Wang et al., 2011; Xie et al., 2011; Zhang
et al., 2007) and (Acciai et al., 2016; Donohue and Ascoli, 2011; Meijering, 2010; Parekh and Ascoli, 2013) for review. In general, existing algorithms can accurately capture the geometrical layout of neurites but are not guaranteed to recover their correct branching topology (Figure 1.1). Topological errors are inevitably present in traces obtained from low signal-to-noise images, images of non-uniformly labeled neurites, or images with high density of labeled structures. Close examination of such traces often reveals topological errors such as broken branches, missing branches, and incorrectly resolved branch crossover regions (stolen branches). This is a particularly a problem for high-throughput projects where topological errors can accumulate over multiple stacks. For example, while tracing a long-range axon from one optical section to the next, even a very low error-rate, say 5% per section, will almost certainly lead to erroneous connectivity after about 20 sections (typically about 10 mm), rendering the trace unusable for brain mapping projects. Clearly, the rate of topological errors in automated reconstruction projects must be carefully controlled (Chothani et al., 2011).

Here we describe an active machine learning approach (Settles, 2012) that has the potential to significantly reduce the number of topological errors in automated traces. Our algorithm first detects a geometrically accurate trace with the Fast Marching method (Cohen et al., 1994; Cohen and Kimmel, 1997; Mukherjee and Stepanyants, 2012; Sethian, 1999), which was extended to incorporate multiple seed points. Next, the initial trace is dismantled to the level of individual branches, and active learning is applied to reconnect this trace based on knowledge of neuron morphology. We show that active learning does not require large sets of training examples, and the results generalize well on image stacks acquired under similar experimental conditions. What is more, when the density of labeled neurites is sufficiently low, automated traces are not significantly different from reconstructions produced manually by trained users.
Figure 1.1: Automated traces of neurites may contain hidden topological errors. A. Example of an automated trace of drosophila olfactory neuron axon (Jefferis et al., 2007). For better visibility, the initial trace (red line) is shifted down with respect to the maximum intensity projection of the image stack. Due to the simple morphological structure of the neuron, as well as high and uniform contrast of the image, the initial trace accurately captures both the geometrical and topological structure of the arbor. B. Maximum intensity projection of an image stack containing mouse neocortical axons of layer 6 neurons (De Paola et al., 2006). The initial trace (red lines) accurately captures the geometry of the neurites, but contains topological errors. C. These topological errors are revealed after labeling each tree of the initial trace with a different color. The red tree in the image is composed of multiple axons which were erroneously connected to each other. D. To correct such topological errors the initial trace is taken apart up to a level of branches and subsequently put together by using the knowledge of neuron morphology. An active machine learning framework is used to accomplish this task. Scale bars in A and B are 20 μm.
Results of this study are based on the analyses of two datasets featured at the DIADEEM challenge (Brown et al., 2011). The OP dataset includes 9 image stacks containing axons of single olfactory projection neurons from Drosophila (Jefferis et al., 2007), and the L6 dataset consists of 6 image stacks containing axons of multiple layer 6 neurons imaged in layer 1 of mouse visual cortex (De Paola et al., 2006). The NCTracer software was used to trace each image stack automatically, as well as manually. The manual traces were generated independently for each stack by three trained users.

**Initial trace of neurites**

We refer to any trace providing geometrically accurate information about the layout of neurites within an image stack as an initial trace. Numerous segmentation and tracking-based methods [see e.g. (Al-Kofahi et al., 2002; Al-Kofahi et al., 2008; Bas and Erdogmus, 2011; Bas et al., 2012; Choromanska et al., 2012; Losavio et al., 2008; Peng et al., 2011; Peng et al., 2010; Schmitt et al., 2004; Srinivasan et al., 2010; Turetken et al., 2012; Turetken et al., 2011; Wang et al., 2011; Xie et al., 2011; Zhang et al., 2007)] can be used to produce initial traces. In this study we adapt the Fast Marching method (Cohen et al., 1994; Cohen and Kimmel, 1997; Mukherjee and Stepanyants, 2012; Sethian, 1999) to grow the initial trace from multiple seed points (Figure 1.2), analogous to the way light from multiple point sources spreads through a non-uniform medium. This process is described by the Eikonal boundary value problem (Sethian, 1999):

\[
\left\| \nabla T(r) \right\| I(r) = 1 \\
T(\partial S) = 0
\]

(1.1)
In this expression, vector \( \mathbf{r} \) represents a position in the image stack (or non-uniform medium), \( I \) is the image intensity normalized to the 0-1 range (analog of the speed of light in the medium), and \( \nabla \) denotes the gradient operator. Light rays originate from the boundary, \( \partial S \), at time zero, and the time map, \( T(\mathbf{r}) \), provides information about the shortest time of arrival of these rays to various locations in the image. Because higher image intensities correspond to faster speeds of light propagation, the arrival time front in the image will preferentially spread along the high intensity structures of neurites (see Figure 1.2C).

The Fast Marching algorithm of Sethian (Sethian, 1999) is an efficient numerical scheme for solving the Eikonal boundary value problem, Eq. (1.1). Since the speed function in our problem is defined by the image intensity, it is always positive. For positive speed functions, the Eikonal boundary value problem can be solved more efficiently than the commonly used alternative – the Hamilton-Jacobi problem of the Level Set method (Sethian, 1999). One reason is that the stability condition required for a numerical solution of the time-dependent Level Set equation is more stringent than that used to solve the Eikonal problem. Specifically, this condition requires very small time steps and thus the Level Set method is expected to be more time consuming. The second advantage of Fast Marching is related to the outward only propagation of the fronts, which can be used to find new front points very efficiently (Sethian, 1999).
**Figure 1.2**: The initial trace can be based on the solution of the Eikonal equation. A. Maximum intensity projection of an OP image stack. Green dots mark the locations of seed points used to initialize the Fast Marching algorithm ($T = 0$ points). The white square outlines the image section used in panels C-E to illustrate the main steps of the algorithm. B. The initial trace resulting from Fast Marching is shown with a red line. C. Arrival time map for the front originating at $T = 0$ from the location marked by the green seed point. Front propagation stops when the front reaches a user-defined Euclidean distance (cyan point). Red arrow shows the path of shortest time delay connecting the two points. D. This path is obtained with the gradient descent algorithm. E. The arrival time map along the path of shortest time delay is re-initialized to $T = 0$ (red line), and the Fast Marching algorithm is restarted. Note that brighter intensities correspond to smaller arrival times.
We implement the Fast Marching algorithm (Sethian, 1999) on a discrete lattice defined by the centers of image voxels, $I = (i, j, k)^T$. Here the time map is evolved from the boundary at $T = 0$ by taking the upwind solution of a discretized version of Eq. (1.1):

$$
\begin{align*}
\frac{1}{s_x} \left[ \text{max} \left( T(i, j, k) - T(i-1, j, k), 0 \right) \right]^2 + \frac{1}{s_y} \left[ \text{max} \left( T(i, j, k) - T(i+1, j, k), 0 \right) \right]^2 + \\
\frac{1}{s_y} \left[ \text{max} \left( T(i, j, k) - T(i, j-1, k), 0 \right) \right]^2 + \frac{1}{s_y} \left[ \text{max} \left( T(i, j, k) - T(i, j+1, k), 0 \right) \right]^2 + \\
\frac{1}{s_z} \left[ \text{max} \left( T(i, j, k) - T(i, j, k-1), 0 \right) \right]^2 + \frac{1}{s_z} \left[ \text{max} \left( T(i, j, k) - T(i, j, k+1), 0 \right) \right]^2
\end{align*}
$$

(1.2)

Parameters $(s_x, s_y, s_z)$ in this expression denote the voxel dimensions which may not be the same due to a typically lower $z$-resolution in confocal and two-photon microscopy images.

The arrival time front is initialized with $T = 0$ at multiple seed points, which are automatically generated along the structure of neurites based on image intensity (Figure 1.2A). As was previously described (Mukherjee and Stepanyants, 2012), the arrival time front is allowed to propagate to a specified distance, $D_{max}$, to establish a local time map. The value of $D_{max}$ was chosen based on two considerations: $D_{max}$ should be larger than the caliber of neurites (3-5$s_x$ for OP and L6) not to produce short spurious branches and, at the same time, not much larger than the shortest branch that needs to be resolved by the algorithm (10$s_x$ in this study). $D_{max} = 15s_x$ was used throughout this study. The path connecting the farthest point of the front to the $T = 0$ boundary is then found by performing gradient descent on $T(i, j, k)$ (see Figure 1.2C-E). Next, the gradient descent path is added to the boundary $\partial S$ and the Fast Marching algorithm is re-initialized from the new boundary. This process continues until a stopping condition is reached, at which point the final $\partial S$ defines the initial trace. The stopping condition used in this study is based on the average
intensity of the last added branch. When this intensity falls below a set threshold (typically 20% of the average intensity of the existing trace), Fast Marching is paused and can then be continued or terminated by the user.

As long as the seed points used to initialize Fast Marching are located in the foreground, and are connected by higher than background intensity paths, their respective Fast Marching fronts are guaranteed to collide. The gradient descent algorithm is invoked in this case as well. Here, gradient descent paths originating from the collision voxel back-propagate into every colliding region, thus connecting their $T = 0$ boundaries. If there is a break in intensity along a neurite linking two seed points, the Fast Marching algorithm may terminate before the fronts have a chance to collide. In addition, high levels of background intensity may lead to erroneous front collisions. These and other topological errors in the initial trace will be corrected as described in the following sections.

**Optimization of the initial trace**

We represent the initial trace as a graph structure consisting of nodes linked by straight line segments. Each node, $k$, is described by its position in the stack, $r^k = (x^k, y^k, z^k)^T$, and the caliber, $R^k$, of the neurite at that location. Information about connectivity among the nodes is stored in the adjacency matrix, $A$. We find this representation to be more convenient than the traditional SWC format of neuron morphology (Cannon et al., 1998) because the latter cannot be used to describe structures containing loops.

Because the initial trace lies sufficiently close to the centerline of neurites, this trace can be optimized by monitoring its fitness in response to small changes in the position and caliber of every node. The fitness function used in this study, $\mathcal{F}(\{r^k, R^k\})$, consists of the intensity integrated along the trace and regularizing constraints on the positions and calibers of the connected nodes:
Here, vectors $r^k$ specify position of the trace vertices, while vectors $l^m$ denote position of voxel centers in the image stack. Index $k'$ enumerates neighbors of vertex $k$. Parameter $\lambda$ denotes the average density of nodes in the trace, i.e. the number of nodes per voxel. Lagrange multipliers $\alpha_r > 0$ and $\alpha_R > 0$ control stiffness of the regularizing constraints. The first term in this expression is the convolution of the image with the Laplacian of Gaussian. This convolution can be performed by using the Fast Fourier Transform (Press, 2007) or, in case of relatively small density of trace nodes, it may be faster to perform explicit summation over the index $m$. In this case, due to the fast decay of the Gaussian factor, the summation can be restricted to a small number of voxels in the vicinity of the trace [see (Chothani et al., 2011) for details].

Maximization of the fitness function, $\mathcal{F}$, is performed with Newton’s method (Press, 2007):

$$
\left\{r^k(n+1), R^k(n+1)\right\} = \left\{r^k(n), R^k(n)\right\} - \beta \left( \hat{H} \mathcal{F} \left( \left\{r^k(n), R^k(n)\right\} \right) \right)^{-1} \nabla \mathcal{F} \left( \left\{r^k(n), R^k(n)\right\} \right)
$$

(1.4)

Variable $n$ in this expression enumerates iterations of the algorithm, parameter $\beta > 0$ controls the iteration step size, $\hat{H}$ denotes the Hessian operator acting on all the node variables $\{r^k(n), R^k(n)\}$, and -1 in the exponent denotes matrix inversion. The position and caliber of each trace node, including branch and terminal points, are synchronously updated in every iteration. The values of all three terms in the fitness function are monitored during optimization. Optimization is
terminated once the relative changes in all three quantities fall below $10^{-8}$. For the OP and L6 datasets considered in this study, the optimization procedure typically converges to a solution in less than 50 iterations. Optimization improves the layout of branches as well as the placement of branch and terminal points in the initial trace (Chothani et al., 2011; Vasilkoski and Stepanyants, 2009). The values of parameters $\alpha_r$, $\alpha_R$, and $\beta$ are constrained by the considerations of algorithm stability, speed of convergence, and accurate representation of curvature and caliber of the neurites. Some of these issues were discussed in (Chothani et al., 2011; Vasilkoski and Stepanyants, 2009).

**Learning branching morphology of neurites**

As shown in Figure 1.1, even when the initial trace accurately describes the geometry of neurites, it often fails to capture the correct branching topology. To address this problem, we disconnect branches of the initial trace from one another and then assemble them into tree-like structures based on prior knowledge of neuron morphology. To discriminate between correct and erroneous ways to assemble branches, different branch merging scenarios are evaluated in a machine learning approach by combining information about various features of the trace. Such features may include distances between branches, branch orientations, average intensities, intensity variations, branch thicknesses, curvatures, tortuosities, colors, and presence of spines or boutons. Features 1-9 of Figure 1.3 were used to produce the results of this study. These features were selected based on our knowledge of neuroanatomy and intuition gained from manual neuron tracing. We carefully examined the decisions we make when faced with branch merging tasks and initially created a list of 17 features, shown in Figure 1.3. Features #15 and #16 are not applicable for the OP and L6 datasets as these datasets include grayscale images of axons only. Features #10 - #14 and #17 were tested but did not improve performance of the classifiers. Hence features #10 - #17 were left out.
of the analysis. This is not to say that these features are not important; they may be useful for other dataset types.

<table>
<thead>
<tr>
<th>#</th>
<th>Feature</th>
<th>Equation</th>
<th>Illustration</th>
<th>Active weights, w</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>L6</td>
</tr>
<tr>
<td>1</td>
<td>Distance between terminal points</td>
<td>D</td>
<td></td>
<td>0.013</td>
</tr>
<tr>
<td>2</td>
<td>Overrun for connected terminal points</td>
<td>OR</td>
<td></td>
<td>0.079</td>
</tr>
<tr>
<td>3</td>
<td>Offset of connecting terminal points</td>
<td>OS</td>
<td></td>
<td>0.011</td>
</tr>
<tr>
<td>4</td>
<td>Deviation from perfect branching angles: 2 branches</td>
<td>[</td>
<td>\theta - 180^\circ</td>
<td>]</td>
</tr>
<tr>
<td>5</td>
<td>Deviation from perfect branching angles: 3 branches</td>
<td>(\langle(\theta - 120^\circ)^{1/2}\rangle)</td>
<td></td>
<td>0.14</td>
</tr>
<tr>
<td>6</td>
<td>Deviation from perfect branching angles: 4 branches</td>
<td>(\langle(\theta - 90^\circ)^{1/2}\rangle)</td>
<td></td>
<td>0.36</td>
</tr>
<tr>
<td>7</td>
<td>Variation in intensity</td>
<td>(\delta I/I\rangle)</td>
<td></td>
<td>0.12</td>
</tr>
<tr>
<td>8</td>
<td>Variation in branch caliber</td>
<td>(\delta R/R\rangle)</td>
<td></td>
<td>-0.88</td>
</tr>
<tr>
<td>9</td>
<td>Number of free terminal points</td>
<td>(N_1)</td>
<td>(N_1 = 1)</td>
<td>0.17</td>
</tr>
<tr>
<td>10</td>
<td>Number of bifurcations</td>
<td>(N_3)</td>
<td>(N_1 = 1)</td>
<td>-</td>
</tr>
<tr>
<td>11</td>
<td>Number of trifurcations</td>
<td>(N_4)</td>
<td>(N_1 = 0)</td>
<td>-</td>
</tr>
<tr>
<td>12</td>
<td>Variation in tortuosity</td>
<td>(\delta T/T\rangle)</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>13</td>
<td>Variation in curvature</td>
<td>(\delta K/K\rangle)</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>14</td>
<td>Planarity of branching</td>
<td>(P)</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>15</td>
<td>Presence of spines or boutons</td>
<td>(s,b)</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>16</td>
<td>Variation in color</td>
<td>(\delta R, \delta G, \delta B)</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>17</td>
<td>Extra dimensions</td>
<td>(x^i, x^j)</td>
<td></td>
<td>-</td>
</tr>
</tbody>
</table>

**Figure 1.3:** Morphological features for classification of branch merging scenarios. Features 1-9 were used to generate the results of this study. The weights of Perceptron and SVM classifiers, obtained by active training on 100 clusters of branch points, show high degree of correlation. As expected, within-dataset correlations were higher than between-dataset correlations indicative of the fact that the classifier algorithms can learn details of neuron morphology that are dataset-specific.
To evaluate different branch merging patterns in the disconnected initial trace, we cluster branch terminal points based on their relative distances. For this, we first create an all-to-all connected graph in which nodes represent the end points of all branches. Next, links between distant nodes (>10sₓ) are removed, exposing clusters of nearby branch points. The threshold distance of 10sₓ was chosen based on two considerations. First, this distance must be larger than the voxel size (sₓ) and the size of a typical gap in intensity resulting from imperfect labeling of branches (0 for OP and ~5sₓ for L6). Second the threshold distance must be smaller than the typical branch length (20sₓ - 50sₓ for OP and L6). Results of branch merging are not sensitive to the precise value of this parameter in the 5sₓ - 15sₓ range. Branch merging is examined within each cluster of branch terminal points independently.

Within a given cluster, all possible branch merging scenarios are considered (Figure 1.4A), and the correct merging pattern is determined in a classification framework. Clusters containing 2 terminal points lead to two scenarios, i.e. to connect or not to connect the terminal points. Three terminal point clusters result in 5 scenarios, 4 terminal point clusters lead to 15 (Figure 1.4A), and the number of scenarios increases exponentially with the complexity of clusters (Figure 1.4B). This exponential increase gives a unique advantage to our classification approach to branch merging.

Generally, machine learning applications require large sets of labeled data. Creating such sets can be very time-consuming and, in many cases, impractical. Our training strategy circumvents this problem by exploiting the large numbers of branch merging scenarios. Labeling the correct branch merging scenario in a single cluster can provide thousands of training examples. Hence, it becomes possible to train the classifier in real time and obtain accurate results by labeling only 10-100 clusters of branch terminal points.
Figure 1.4: Branch merging scenarios. A. Illustration in the middle shows the maximum projection image of two neurites from Figure 1.1B. The neurites appear fused in 3D, leading to a provisional branch point which must be resolved automatically. The initial trace of these neurites (lines are shifted down and to the right for clarity) has been disconnected at the branch point resulting in a terminal point cluster which has to be merged according to one of the 15 possible scenarios (circled traces). Each scenario, \( i \), is described by a unique multi-dimensional feature vector, \( x^i \). B. The number of possible branch merging scenarios increases exponentially with the number of terminal points, leading to large numbers of classification examples.

All possible branch merging scenarios are evaluated within a given cluster of branch terminal points. Each scenario, \( i \), is characterized by a feature vector \( x^i \) (Figure 1.4A) whose components consist of features of the trace that may be important for selecting the correct branch merging scenario (Figure 1.3). The problem is thus reduced to learning the best set of weights, \( w \), for discriminating between the correct and erroneous scenarios within every cluster,

\[
\mathbf{w}^T x^{\text{all erroneous mergers}} > \mathbf{w}^T x^{\text{correct merger}}
\]

(1.5)
This formulation leads to another important advantage for the implementation of the classification strategy. Because the problem is linear, Eq. (1.5) can be rewritten as:

$$w^T \left( x^{\text{all erroneous mergers}} - x^{\text{correct merger}} \right) > 0$$

(1.6)

This results in a subtractive normalization of the feature vectors within individual clusters. Because branch merging scenarios are only compared within clusters, Eq. (1.6) effectively normalizes for variations in image intensity and density of neurites across clusters.

The classification problem of Eq. (1.6) is solved with sign-constrained perceptron (Engel and Broeck, 2001) or SVM classifiers (Wang, 2005), which were modified to be able to account for the relative importance of some training examples. The sign-constrained perceptron algorithm was previously described in (Chapeton et al., 2012):

$$\frac{1}{N} w^T \Delta x^\mu > \frac{\kappa}{\sqrt{N}}, \quad \mu = 1, 2, \ldots, m$$

$$w_k g_k \geq 0, \quad k = 1, 2, \ldots, N$$

(1.7)

where $w$ is the weight vector of the perceptron classifier, $\Delta x^\mu$ is the difference between the feature vectors for the erroneous merger $\mu$ and the correct merger from the same cluster, $N$ is the number of features (9 features were used in this study), and $m$ is the number of comparisons made (total number of scenarios minus number of clusters). The value of the parameter $g_k$ can be set to -1 or 1, constraining the corresponding weight, $w_k$, to be negative or positive, or set to 0, in which case the weight is unconstrained. Because larger distances, overruns, and offsets of terminal points (see
Figure 1.3) decrease the likelihood that branches should be merged, the weights of these features were constrained to be positive. In addition, the weight associated with the number of free terminal points was constrained to be positive to promote branch merging. All other weights were left unconstrained as we did not have clear motivation for doing otherwise. Hence, \( \mathbf{g} = (1,1,1,0,0,0,0,0,1)^T \) was used in this study. Parameter \( \kappa \) is referred to as the perceptron robustness (analogous to SVM margin). Increasing \( \kappa \) should initially improve the generalization ability of the perceptron, but as the perceptron fails to correctly classify a progressively increasing number of training examples, this generalization ability should decrease. We used the leave-one-out cross-validation scheme to examine this trend. In this scheme, training is done on all but one labeled example, and the remaining example is used for validation.

\[ \text{Figure 1.5: How to choose best classification parameters.} \ A. \text{Leave-one-out cross-validation error-rate as function of the perceptron robustness parameter, } \kappa \text{ [see Eq. (1.7)].} \ B. \text{Same error-rate as function of the SVM parameter, } C \text{ [see Eq. (1.9)]. The inset shows how SVM margin, } M, \text{ depends on } C. \text{ Solid and dotted lines show the results for the OP and L6 datasets respectively. Large empty circles indicate the parameter values that were used throughout this study, } \kappa = 2^0 \text{ and } C = 2^{20}. \]
In Figure 1.5 each branch merging cluster was used once for validation and the results were averaged. Figure 1.5A shows that there is a large range of $\kappa$ for which the perceptron performs reasonably well for both L6 and OP datasets. The value of $\kappa$ was set to 1 throughout this study.

The sign-constrained perceptron problem of Eq. (1.7) was solved by using a modified perceptron learning rule (Engel and Broeck, 2001):

$$
\Delta w = \theta \left( \frac{\kappa}{\sqrt{N}} - \frac{1}{N} w^T \Delta x^{\mu} \right) \frac{1}{\sqrt{N}} \Delta x^{\mu}
$$

$$
w_k = w_k \theta(w_k g_k), \quad k = 1, 2, ..., N
$$

(1.8)

In this expression, $\Delta w$ denotes the change in the perceptron weight vector in response to presentation of the training example $\mu$; $\theta$ is the Heaviside step function, which is defined to be 1 for non-negative arguments and zero otherwise. The step functions in Eq. (1.8) ensure that training is not done on learned examples, and that the perceptron weights violating the sign-constraints are set to zero at every step of the algorithm. Perceptron weights are updated asynchronously by training on examples, $\mu$, that are drawn from the set of all examples with probabilities proportional to user-defined cluster weights, $Q_\mu$. All cluster weights are initially set to 1 and can be modified by the user to increase the probabilities with which examples from some clusters come up for training. This makes it possible to enforce learning of certain rare branch merging topologies. Though user-defined cluster weights may be used to improve the outcome of training, this feature was not examined in the present study to avoid subjectivity associated with different choices of $Q_\mu$. 

32
An SVM classifier can also be used to solve the system of inequalities in Eq. (1.6). To incorporate the used-defined cluster weights, $Q_\mu$, we modified the standard formulation of the SVM problem (Wang, 2005), and in this study, maximize the following dual Lagrangian function to obtain the SVM weight vector $w$:

$$L_d(a) = \frac{1}{m} \sum_{i=1}^{l} \alpha_i - \frac{1}{2Nm} \sum_{i,j=1}^{l} (\Delta x^i)^T \Delta x^j \alpha_i \alpha_j$$

$$0 \leq \alpha_i \leq CQ_\mu, \quad i = 1, 2, ..., l$$

$$w = \frac{1}{m} \sum_{i=1}^{l} \alpha_i \Delta x^i$$

(1.9)

In these expressions, $l$ is the number of SVM support vectors and $C$ is the SVM margin (see the inset in Figure 1.5B). Like the perceptron robustness, there is a large range of values of $C$ for which the SVM produces reasonably good generalization results for both datasets. $C = 2^{20}$ was used to produce results of this study. Again, all used-defined cluster weights, $Q_\mu$, were set to 1 during training.

**Active learning strategy**

Here we describe a pool-based sampling approach (Lewis and Gale, 1994) that can be used to actively train the Perceptron and SVM classifiers on branch merging examples. In this approach the user selectively draws queries from the pool of all branch merging clusters based on the value of the confidence measure:

$$Confidence = \frac{e^{-w^T x_{source\ merge}/\tau}}{\sum_{i\ all\ mergers} e^{-w^T \hat{z}_i/\tau}}$$

(1.10)
This measure assigns low confidence values (in the 0 – 1 range) to clusters in which the erroneous merging scenarios are located close to the decision boundary defined by \( w \). Parameter \( T \) controls the spread of confidence values but does not affect their order. This parameter was set to 1 throughout the study. Training can be performed after labeling a single or multiple low confidence clusters, and the confidence measure is updated after each training step. It is essential that clusters in which the correct merging scenario cannot be identified with high certainty should not be used for training, as a small number of errors in the labeled set may significantly worsen the performance of classifiers.

\textbf{Evaluating the tracing strategy}

The methodology described in this chapter is implemented in the NCTracer software for automated tracing of neurites. This methodology consists of two major parts – initial tracing and branch merging. In the first part, an initial trace is created by using the Voxel Coding (Vasilkoski and Stepanyants, 2009; Zhou et al., 1998; Zhou and Toga, 1999) or the Fast Marching (Cohen et al., 1994; Cohen and Kimmel, 1997; Mukherjee and Stepanyants, 2012; Sethian, 1999) algorithm, and optimized to ensure that the trace, including its branch and terminal points, conforms well to the intensity in the underlying image. Below we examine the initial traces from two very different dataset types: axons of single olfactory projection neurons from Drosophila (OP dataset, \( n = 9 \) image stacks) (Jefferis et al., 2007) and axons of multiple layer 6 neurons imaged in layer 1 of mouse visual cortex (L6 dataset, \( n = 6 \) image stacks) (De Paola et al., 2006). These datasets were featured at the DIADEDM challenge (Brown et al., 2011) and serve as benchmarks for automated reconstruction algorithms. Figure 1.1A and 1.1B show representative image stacks from the OP and L6 datasets. The initial traces are superimposed on the maximum intensity projections of the image stacks, and are slightly shifted for better visibility. As can be seen, these initial traces
accurately represent the geometry of neurites contained in the image stacks. However, a closer examination of the L6 trace topology reveals numerous erroneously merged (stolen) branches. Such errors in the initial trace often occur when the neurites belonging to different trees appear to be in contact due to poor z-resolution or due to high density of labeled structures. Presence of these topological errors becomes evident after labeling distinct tree structures with different colors (Figure 1.1C). The second part of our automated tracing algorithm uses a machine learning approach that actively learns the morphology of neurites in an attempt to resolve the errors present in the initial trace (see Figure 1.1D).

Comparison of automated initial traces and manual user traces

Below we evaluate how well automated and manual traces capture the layout (geometry) of the neurites in the image stack, and how well they represent the morphology of branching tree structures (topology). Similar comparisons have been carried out in other studies (Choromanska et al., 2012; Gillette et al., 2011a; Mayerich et al., 2012). Each OP and L6 image stack was traced automatically using the Fast Marching algorithm as well as manually by three trained users. Figure 1.6A shows an example of the resulting four traces of a single OP stack. Inevitably, imperfect labeling and limited resolution of optical microscopy lead to uncertainties in tracing. Trained users often resolve such uncertainties differently from one another, and hence no single trace can be viewed as a gold standard. Thus, we had to first establish quantitative measures describing the baseline inter-user variability, and only then evaluate the performance of the automated tracing algorithm in comparison to this baseline. To this end, each manual trace was chosen to be the gold standard and compared to the automated trace and the remaining two manual traces. This led to 6 inter-user and 3 automated-to-user trace comparisons for each stack.
Figure 1.6: Assessing the quality of automated traces. A. Three manual traces (green, blue, and yellow) and one automated trace (red) are superimposed on a maximum intensity projection of an OP neuron. The traces are staggered upward for better visibility. The inset shows a zoomed view of the boxed region. Scale bar is 20 μm. B. Several geometrical and topological features are used to compare traces. Gold standard trace (yellow) and test trace (blue) are shown. Both traces are composed of nodes connected by edges of length $d$. Nodes on these traces are referred to as corresponding nodes if they are located within distance $h$ of each other ($d < h$). Circles highlight false negative and false positive branch and terminal points. C–E. Automated traces reliably capture the geometry of neurites. Nine OP axons were reconstructed with NCTracer, first automatically and then manually by three trained users. The probability densities for distances between the corresponding trace nodes (C), terminal points (D), and branch points (E) were used as metrics for geometrical comparisons. Red lines show the results of automated-to-user trace comparisons. Here, all user traces for every stack were used one by one as the gold standard, leading to 27 automated-to-gold standard trace comparisons.
comparisons. The results were pooled. Blue lines show similar results based on 54 user-to-user trace comparisons. **F-H.** Automated traces accurately represent the topology of OP neurons. Three topological measures were compared: false positive/negative trace lengths (F), numbers of false positive/negative terminal (G) and branch (H) points. Red and blue bars show the fractions of automated-to-user (n = 27) and user-to-user (n = 54) comparisons for different error types. The fractions for false positive and false negative errors are indicated with the bars above and below the x-axes.

To ensure the uniformity of the reconstructed dataset, all traces were subdivided into segments of equal length (d = 0.25 voxels). To compare a pair of traces (a test trace and a gold standard trace) we perform a bi-directional nearest neighbor search to find corresponding nodes, i.e. nodes on the two traces separated by less than h = 10 voxels (see Figure 1.6B). A node in the test trace which has (does not have) a corresponding node in the gold standard trace is referred to as a true (false) positive node. A node in the gold standard trace for which there is no corresponding node in the test trace is referred to as a false negative node. Short terminal branches (less than 12 voxels) and dim branches (average intensity less than 0.12) were excluded from the comparisons.

Results of the geometrical comparisons between automated initial traces and manual traces for the OP image stacks are shown in Figure 1.6C-E. The plots show probability densities of distances between corresponding nodes, corresponding branch points, and corresponding terminal points for both inter-user (blue lines), as well as automated-to-user comparisons (red lines). The geometrical precision of the automated and manual traces is evidenced by the fact that 95% of distance values lie below 2.3 voxels in Figure 1.6C, 7.3 voxels in Figure 1.6D, and 6.6 voxels in Figure 1.6E. More importantly, the difference between mean distances for the inter-user and automated-to-user comparisons (0.19, 0.51, and 0.65 voxels respectively) is smaller than the resolution of the image, and thus should have little bearing on trace dependent measurements. Similar conclusions were drawn from the geometrical comparisons of automated and manual traces of the L6 dataset (Figure 1.7A-C).
Figure 1.7: Assessment of initial traces of multiple neuron axons. Six L6 image stacks were reconstructed manually and automatically with NCTracer (see Figure 1.6 legend for details). A–C. Automated traces reliably capture the geometry of neurites. The probability densities for distances between the corresponding trace nodes (A), terminal points (B), and branch points (C) were used as metrics for geometrical comparisons. Red lines ($n = 18$) and blue lines ($n = 36$) show the results of automated-to-user and user-to-user trace comparisons. D–F. While the automated traces capture the geometry of the neurites well, they contain a markedly large number of false positive branch points (F). These topological errors result from erroneous mergers of distinct axons that pass in close proximity of one another.

Topological errors that occur due to incorrectly merged branches are more difficult to detect and can be detrimental to circuit reconstruction projects. Three measures were selected to quantify the extent of such errors: false positive/negative trace lengths, numbers of false positive/negative terminal points, and numbers of false positive/negative branch points. The results of comparisons for the OP dataset (Figure 1.6F–H) show that similar numbers of topological errors were made by the algorithm and the users, and these numbers were generally small (less than one false
positive/negative branch or terminal point per stack). For the L6 image stacks, the mismatches in length for the automated and manual traces were similar (Figure 1.7D), indicating that the automated algorithm performed as well as trained users in tracing the majority (in terms of length) of labeled structures. However, in contrast to manual traces, automated traces contained more false positive/negative terminal points (Figure 1.7E) and markedly larger number of false positive branch points (Figure 1.7F). The former errors result from branches that are broken due to imperfect labeling, while the latter arise from a specific artifact of the Fast Marching algorithm, i.e. merging nearby, but distinct branches. In particular, lower z-resolution of an image stack makes such mergers more prevalent, leading to larger numbers of false positive branch points.

**Active learning of branching morphology of neurites**

To resolve the above mentioned topological errors, branches of the initial trace were disconnected from one another and merged into tree-like structures in an active learning approach described previously. Briefly, the positions of branch and terminal points were clustered based on distance, and branch merging was performed within every cluster independently (see Figure 1.4). Perceptron and SVM classifiers were designed and trained online to accomplish the branch merging task generating the final traces of the OP and L6 image stacks.

To assess the performance of the classifiers, their generalization error rates were monitored as functions of the number of training examples. Figures 1.8A-B compare the performance of the classifiers trained on randomly selected branch merging examples with that of classifiers trained in an active learning approach. The plots show that the active learning approach provides a clear advantage in terms of the number of training examples required to reach a given error rate. For each dataset, error rates of less than 5% were achieved by both classifiers with less than 40 actively chosen training examples. The rapid decline of the generalization error rate validates our choice
of features used for the branch merging task (see Figure 1.3). As expected, trained SVM and perceptron classifiers established nearly identical decision boundaries, as judged by the distance between their normalized weight-vectors (0.29 for OP and 0.10 for L6). In contrast, between-dataset distances were larger (0.63 for perceptron and 0.63 for SVM), indicative of the fact that classifiers were able to capture dataset specific morphological information.

Geometry and topology of the final automated traces produced by the branch merging algorithm were compared to the user traces in the manner described in Figures 1.6 and 1.7. No significant geometrical changes resulted from automated branch merging. This was expected, as trace modifications that accompany branch merging are confined to very local regions in the vicinity of branch or terminal points. In addition, automated branch merging did not alter the topology of initial traces of OP neurites. The reason is that the initial traces of these morphologically simple structures did not contain significant topological errors in the first place (Figures 1.6F-H). As for the topology of L6 traces, no significant changes were observed in false positive/negative lengths (Figure 1.7D) and terminal point numbers (Figure 1.7E).
Figure 1.8: Online training can be used to reduce the numbers of topological errors present in initial traces. A. Generalization error-rate as function of the number of training clusters for the perceptron classifier. Black lines (mean) surrounded by gray margins (standard errors) show the results of random training. For each number of training clusters, training sets were generated at random 1000 times. Training was performed on each set of clusters, while testing was done on the remaining clusters. Results for all 1000 experiments were averaged. Solid and dotted lines show the results for the OP and L6 datasets respectively. Red lines show the corresponding results for active training experiments. B. Same for the SVM classifier. C. The number of false positive branch points present in the initial trace of L6 dataset (Figure 1.7F) is greatly reduced by the branch merging algorithm. D. The sum of false positive and false negative branch point errors is an increasing function of length density of labeled neurites. Length density is calculated as the average length of neurites traced by the users divided by the image stack volume. Error-bars indicate the range of branch point errors. Automated tracing algorithm performs as well as trained users when the density of labeled neurites is low (< 0.003 μm²).
As was intended, automated branch merging greatly reduced the number of false positive branch points present in the initial traces (Figure 1.8C vs. Figure 1.7F). Though the reduction in the number of false positive branch points was large (about two-fold), the branch merging algorithm failed to achieve the level of user performance (Figure 1.8C). To examine the reason behind this disparity we plotted the sum of false positive and false negative branch point errors for every L6 stack as function of length density of neurites contained in the stack (Figure 1.8D). The length density is defined as the total length of traced neurites (in μm) divided by the stack volume (in μm³) and was calculated for each image stack by averaging over all user traces. These comparisons show that in every stack the branch merging algorithm substantially reduced the total number of errors present in the initial trace. What is more, when the density of labeled neurites was small (less than 0.003 μm², e.g. Figure 1.1D), the resulting final automated traces were on par with user reconstructions.

**Comparisons with other automated tracing tools**

The geometrical and topological measures used to evaluate the quality of automated traces were also used to compare the performance of NCTracer, Vaa3D (Xiao and Peng, 2013), and NeuronStudio (Rodriguez et al., 2009). To this end, automated traces of OP and L6 image stacks were obtained with Vaa3D and NeuronStudio. We visually inspected these traces and varied the software parameters to achieve good coverage and performance. Inter-user and automated-to-user comparisons were performed as previously described. To evaluate the geometry of automated traces we calculated the mean distances between corresponding nodes and corresponding terminal and branch points. To assess the topology of automated traces, we obtained the Miss-Extra-Score (MES) for trace length and for the numbers of terminal and branch points (Xie et al., 2011). Trace MES is defined as the ratio of the gold standard length reduced by the false negative length to the
gold standard length increased by the false positive length. Terminal and branch point MES are defined in a similar manner. The results of these comparisons are shown in Table 1.1.

<table>
<thead>
<tr>
<th></th>
<th>OP dataset (single cell axons)</th>
<th>L6 dataset (axons of multiple cells)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>geometrical measures</td>
<td>topological measures</td>
</tr>
<tr>
<td></td>
<td>trace dist.</td>
<td>TP dist.</td>
</tr>
<tr>
<td>Inter-user</td>
<td>1.03±0.02</td>
<td>2.18±0.08</td>
</tr>
<tr>
<td>NCTracer</td>
<td>1.19±0.02</td>
<td>2.57±0.14</td>
</tr>
<tr>
<td>Vaa3D</td>
<td>1.39±0.03</td>
<td>4.86±0.13</td>
</tr>
<tr>
<td>NeuronStudio</td>
<td>2.19±0.17</td>
<td>4.17±0.24</td>
</tr>
</tbody>
</table>

Table 1.1: Comparisons of manual and automated traces. Manual traces were created by three trained users. Automated traces were generated with NCTracer as described before, as well as with Vaa3D (Xiao and Peng, 2013) and NeuronStudio (Rodriguez et al., 2009) automated tracing software. Parameters for Vaa3D and NeuronStudio software were tuned to the best of our knowledge. Because NeuronStudio does not automatically trace neurites of multiple cells, this software was not used for the L6 dataset (NA’s in the table). Geometrical and topological measures used are described in the text. Their numerical values are given in mean ± s.e.m format (n = 9 for OP and n = 6 for L6). All calculations were performed on a 3.59 GHz, 24 GB workstation running Windows 7. The following abbreviations are used in the table: dist. – distance, TP – terminal points, BP – branch points, MES – Miss-Extra-Score (Xie et al., 2011)

Smaller distance and higher MES indicate greater affinity between test and gold standard traces. Table 1.1 shows that all automated tracing tools were able to capture trace geometry and topology of single OP axons reasonably well. The advantage of the branch merging strategy proposed in this chapter becomes evident from examining the values of topological measures for L6 stacks,
which contain multiple axons. According to these measures, NCTracer significantly outperforms other software. And in general, all geometrical and topological measures of NCTracer are closest to the inter-user measures. Table 1.1 also shows a trade-off between the quality of automated traces and tracing time. Vaa3D and NeuronStudio are 15-20 fold faster than NCTracer. We do not view this as a major drawback because tracing of single stacks with the current version of NCTracer can be easily performed on modern day workstations, while high-throughput projects could still be carried out on computer clusters.

Discussion

Much of our understanding of brain structure and function is derived from quantitative analyses of neuron shapes. Researchers routinely utilize partial or complete single cell reconstructions, as well as reconstructions of multiple cells often spanning several stacks of images to address various questions. Single cell reconstructions are often used in cell classification and comparative neuroanatomy studies, theoretical studies of neuron shapes, and detailed computational models of intracellular activity. Single cell reconstructions are frequently pooled in silico to simulate structural connectivity of local neural circuits. Reconstructions of multiple labeled cells are used for the analyses of synaptic connectivity in local circuits, in vivo studies of circuit plasticity, and large-scale brain mapping projects. There is no doubt that automating the tracing process will advance these studies, significantly increasing their throughput and eliminating the biases and variability associated with manual tracing.

It is important to understand that it is usually not sufficient to obtain the basic layout of all labeled neurites. In particular, projects aimed at the analyses of synaptic connectivity require accurate knowledge of branching morphology of individual cells (Figure 1.1). In this chapter, we use machine learning to evaluate topologically different scenarios of constructing automated traces.
(Figure 1.4A) and then determine the correct branching pattern based on previously learned morphological features. A machine learning approach to image processing typically requires a large labeled set of examples, and creating such a set can be very time-consuming. Our active learning strategy circumvents this problem by taking advantage of the combinatorial nature of the numbers of branch merging scenarios (Figure 1.4B). Another advantage of this strategy is subtractive normalization, Eq. (1.6). Branch merging scenarios are only compared within clusters, normalizing for the variations in local intensity and density of labeled neurites.

The results show that the quality of automated traces is strongly dependent on the length density of labeled neurites. When this density is lower than 0.003 $\mu$m$^{-2}$ the automated tracing algorithm performs on par with trained users (Figure 1.8D); the reliability of automated traces diminishes rapidly with increase in density beyond this point. Hence, proofreading and error-correction may be required for some automated traces. Proofreading must be done in a computer guided manner, which is particularly important for high-throughput reconstruction projects. The confidence measure described in Eq. (1.10) can be used to convey information about the certainty in the outcome of automated tracing. This measure can be calculated for every vertex in the trace and can be used to direct the user’s attention to the most uncertain parts of the trace. Only the lowest confidence mergers will need to be examined by the user, leading to a substantial reduction in proofreading time. Such low confidence regions can be highlighted automatically and the user would choose from an ordered set of best alternative scenarios (based on decreasing confidence).

With the automation of tracing and proofreading it should be possible to map intact, sparsely labeled circuits on the scale of a whole brain, e.g. in the fly or the mouse. Consider a hypothetical experiment of mapping structural connectivity in the mouse brain. The adult mouse brain is roughly 500 mm$^3$ in volume (Ma et al., 2005). Subsets of mouse neurons can be labeled in vivo to
reveal the layout of their axonal and dendritic arbors. The brain can then be divided into \(0.5 \times 0.5 \times 0.1\) mm\(^3\) optical sections, and imaged in 3D with two-photon or confocal microscopy at \(0.5 \times 0.5 \times 1.0\) \(\mu\)m\(^3\) spatial resolution. This procedure would result in 20,000 stacks of images, each composed of \(1000 \times 1000 \times 100\) voxels, totaling 2 TB of raw imaging data. A dataset of this size would have to be reconstructed on a high-performance computer cluster, and the results could be viewed and proofread on modern-day workstations. Depending on the density of labeling, reconstruction of a single stack may take on the order of 1 core-hour, or 20,000 core-hours for the entire brain. Thus, whole mouse brain mapping is no longer an unfeasible goal.
Chapter 2. Computer assisted detection of axonal bouton structural plasticity in in vivo time-lapse images

Work discussed here was performed in collaboration with experimental groups based at University of Geneva and École Polytechnique Fédérale de Lausanne.

Background

The repertoire of synaptic connectivity within neuronal networks is immensely increased through the continuous formation and elimination of synapses (Chklovskii et al., 2004; Stepanyants et al., 2002). Indeed, in vivo imaging studies over the last 15 years have shown that synaptic structures remain dynamic throughout adulthood (Holtmaat and Svoboda, 2009; Trachtenberg et al., 2002). This structural plasticity, i.e. the appearance, disappearance, and the morphological modifications of synapses in the adult brain has been established as a fundamental underpinning of learning and experience-dependent changes in neuronal circuits (Holtmaat and Caroni, 2016; Holtmaat and Svoboda, 2009; Stepanyants and Chklovskii, 2005).

Synapses in the central nervous system are morphologically distinct structures, visible only in electron microscopy (EM). In light microscopy (LM) a synapse can be detected based on the presence of a swelling on the axon, referred to as bouton, or a protrusion from the dendrite, known as spine. In the cerebral cortex, the majority of excitatory synapses and a minority of inhibitory synapses occur on dendritic spines (Gray, 1959). Spines can easily be detected, hence most studies of structural plasticity have used manual or semi-automated tracking of these structures in time-lapse images to infer circuit changes (Holtmaat and Svoboda, 2009). Yet, studies relying on

---

2 D. Lebrecht\textsuperscript{a}, Daniela Sahlender\textsuperscript{b}, Anne Jorstad\textsuperscript{b}, Graham Knott\textsuperscript{b} and A. Holtmaat\textsuperscript{a}

\textsuperscript{a} University of Geneva,\textsuperscript{b} École Polytechnique Fédérale de Lausanne, Switzerland
tracking of dendritic spines may not reveal the full extent of synaptic plasticity because synapses can also occur on dendritic shafts. On the other hand, a dendrite’s presynaptic apposition can be detected as an irregularity or swelling on the axon. Similar to dendritic spines, such axonal boutons have long since been recognized as sites of functional connections between neurons (Van Gehuchten, 1904). Therefore, the detection of these structures would provide a powerful means to analyze synaptic connectivity (Markram et al., 2015; Meyer et al., 2010a).

Many EM studies have revealed a variety of presynaptic morphologies and the arrangements of vesicles, endoplasmic reticulum, and mitochondria therein (Harris and Weinberg, 2012). This is the only method capable of verifying that LM observations of axonal boutons do indeed correspond to synaptic contacts, but applying such a method every time is difficult across large volumes and impossible at more than one time point. Therefore, the few real-time imaging studies of presynaptic plasticity in vivo used time-lapse LM to detect and track individual axonal boutons (Chen et al., 2015; De Paola et al., 2006; Grillo et al., 2013; Holtmaat and Svoboda, 2009; Johnson et al., 2016; Keck et al., 2011; Majewska et al., 2006; Mostany et al., 2013; Qiao et al., 2016; Stettler et al., 2006; Villa et al., 2016). In these studies, visually isolated boutons are typically scored in a binary fashion, i.e. present or absent, by virtue of their integrated fluorescence and arbitrarily defined thresholds. However, due to jitter in fluorescence caused by fluctuations in imaging conditions, binary scoring of boutons can lead to high false positive/negative rates.
**Figure 2.1:** Challenges in LM-based bouton detection and measurement. **A.** Maximum intensity $xy$ projection of an image stack showing axons of fluorescently labeled neurons in superficial layers of mouse barrel cortex. High density of labeled axons makes it difficult to automatically detect boutons and track their structural changes over time. Scale bar is 20 μm. **B.** A subset of labeled axons from the region outlined in (A). To improve visibility, image intensity beyond five voxels from the axon centerlines was set to zero. Bouton detection and bouton size measurement are confounded by large variations in fluorescence levels across axons. **C.** Axons from (B) shown on the $zx$ maximum intensity projection. Horizontal scale bars in (B) and (C) are 5 μm. Vertical scale bar in (C) is 15 μm. Lower resolution in $z$ compared to $xy$ is yet another challenge in bouton analyses. **D-F.** Magnified views of the highlighted boutons from (B). Close proximity of boutons on an axon (D), large range of bouton sizes (E), and large range of bouton fluorescence levels (D-F), present additional obstacles to accurate bouton detection and measurement. Scale bar in (D-F) is 1.25 μm.

Computer-assisted methods can aid bouton detection in sparsely labeled tissue (Song et al., 2016), but many challenges remain. The most difficult challenges to overcome result from the limited resolution of LM (mainly along the optical axis, $z$). For instance, due to the high density of boutons on cortical axons, boutons often lie in close proximity to one another and can appear fused in microscopy images (Figure 2.1). In densely labeled tissue bouton detection is further confounded by the fact that axons can also appear fused to one another, which locally increases the integrated fluorescence. Furthermore, spatially and temporally non-uniform expression levels and variability
in axon caliber complicate the tracking of boutons over time. Such complications lead to inconsistencies and bias in LM-based bouton detection methods. The bias will affect bouton density estimates, while the inconsistencies, combined with binary scoring of boutons, will lead to an apparent increase in the bouton turnover rate. Below, we describe methodology designed to deal with the challenges in bouton analyses based on LM images acquired \textit{in vivo}.

\textbf{In vivo imaging}

All experiments were performed according to the guidelines of the Swiss Federal Act on Animal Protection and Swiss Animal Protection Ordinance. The ethics committee of the University of Geneva and the Cantonal Veterinary Office (Geneva, Switzerland) approved all experiments.

For development of the detection methodology we used data from mice that had been repeatedly imaged as part of a larger optical microstimulation and behavioral study (described in the following chapter). Only the methodology that is relevant for the present study will be mentioned here. Adeno-associated viral (AAV) vectors encoding floxed GFP (AAV2/9.CAG.flex.eGFP.WPRE.bGH2 [Upen]) were co-injected with an AAV vector encoding Cre (AAV2/9.hSynapsin.hGHintron.GFP-Cre.WPRE.SV40 [Upen]) at a ratio of $0.15 \times 10^9 : 1$ (genome copies : genome copies). This produced an average of about 150 GFP-expressing cells per animal, mostly in layer 2/3 and layer 5 of the barrel cortex. We exclusively imaged the processes of these neurons in the upper layers of the cortex. Hence, the source of the fluorescent signal in the green channel is dominated by cytosolic GFP.

Following the AAV injection, we implanted a 3 mm diameter glass window over the barrel cortex as previously described (Holtmaat et al., 2009). Imaging experiments were started 6 weeks after the virus injection. Imaging was performed under anaesthesia (0.5–1.5\% isoflurane). A custom-
built alignment system was used to ensure identical positioning of the mouse’s cranial window at different time points, and to avoid rotations relative to the axial dimension of the objective. The system was based on aligning a beam reflected by the cranial window through 2 apertures, which uniquely determines a line in 3 spatial dimensions. The apertures and the laser position were constant, and the only free variables were the cranial window position and rotation (i.e., the $x$, $y$, and $z$-positions, as well as the row and pitch needed to be identical between sessions to satisfy the passage of light through the 2 apertures). We used a custom built 2-photon laser scanning microscope (2PLSM) (Janelia Research Campus, model Non-MIMMS in vivo microscope) to acquire in vivo anatomical images of the anesthetized animals through the cranial window. We used the ScanImage software (Janelia Research Campus, Vidrio Technologies) (Pologruto et al., 2003) to record the 2PLSM images, and additional custom software to align anatomical structures present on the central image stack by using a red-green overlay between current and past images.

For imaging we used a 20x water immersion objective (NA 0.95, XLUMPlanFI, Olympus, Japan). Images were acquired at a voxel volume of $\approx 0.13 \times 0.26 \times 0.8 \, \mu m^3$ ($x \times y \times z$). We binned the data in the $x$ dimension to generate isometric voxels in $x$ and $y$ for all analyses. The voxel dwell time was 0.8 $\mu s$ for the full resolution images (and thus twice as high after 2x binning of the $x$ dimension). Each ROI comprised an image stack of $\approx 270 \times 270 \times 250 \, \mu m^3$. As the imaging light source we used a Ti:sapphire femtosecond pulsed laser (Chameleon ultra II, Coherent) that was lasing at 1010 nm. Emitted fluorescence light was split into two channels using a dichroic mirror (Semrock, FF735-Di01-35.5 $\times$ 49.0) and two bandpass filters (red channel, Semrock FF01-607/70; green channel, Semrock FF01-530/70). Each channel was equipped with a PMT (red channel, Hamamatsu R3896; green channel, Hamamatsu H10770PA-40SEL). Images were analyzed only in the green channel.
Correlative 3D EM

Electron microscopy was carried out on a 2PLSM imaged brain region using a previously described protocol (Maco et al., 2014). Briefly, an image of the blood vessel pattern below the cranial window was taken immediately after the last 2PLSM imaging session. The animal was then fixed via cardiac perfusion with a buffered mix of aldehydes, and vibratome sections were cut from the imaged region tangential to the cortical surface. The ROI was then located in the sections and the 2-photon laser was used to burn fiducial marks into the fixed tissue immediately adjacent to the group of fluorescent neurites of interest. This marked section was then stained in reduced osmium, followed by osmium and uranyl acetate and embedded in durcupan resin. Once hardened, the resin block was trimmed close to the visible laser marks and was placed inside the focused ion beam scanning electron microscope (FIBSEM; CrossBeam 540, Zeiss NTS, Germany). An electron beam of 1.6 kV with a current of 600 nA was used for imaging with the ESB detector, and images were collected at 6 nm/pixel. The ion beam removed 12 nm of resin between images.

The aligned series of images was then imported into FIJI (Schindelin et al., 2012), and TrakEM2 software (Cardona et al., 2012) was used to reconstruct the 2PLSM imaged axons. The models were then imported in Blender software (Blender Foundation, Amsterdam), and NeuroMorph toolset (Jorstad et al., 2015) was used to measure bouton and mitochondrial volumes as well as PSD surface areas.

LM-based bouton detection and measurement of structural changes

The heuristic strategy to detect and quantify boutons in 2PLSM images consists of the following major steps: (1) tracing axons in 3D, (2) optimization of traces, (3) generation of axon intensity profiles, (4) detection of putative boutons based on the profiles, (5) normalization of intensity
profiles and calculation of bouton weights, and (6) matching putative boutons across time-lapse images.

Axons can be traced automatically or manually with various tools (Acciai et al., 2016; Parekh and Ascoli, 2013). In this study, high density of labeled axons (Figure 2.1A) precluded the possibility of automated tracing. Therefore, axons were traced manually by using NCTracer software, and traces were subsequently optimized (Chothani et al., 2011; Gala et al., 2014). In this study, a simplified version of the optimization algorithm was used in which positions of trace nodes were optimized, but calibers of trace nodes were kept constant at 3 voxels (roughly equal to axon diameter in images). The average density of trace nodes was set to 0.5 per voxel, and the parameter controlling trace stiffness was set to 0.001. Following optimization, two intensity profiles were generated for each axon by convolving specifically designed filters with the image at all trace node positions and scaling the results to unit means. While various filters can be used to generate axon intensity profiles, in this study, we settled on the following two: (1) a modified, multi-scale Laplacian of Gaussian filter \((\text{LoG}_{xy})\) to detect putative boutons and (2) a fixed size Gaussian filter \((G)\) to provide an estimate of axon intensity in the regions devoid of boutons. In the following we will refer to such inter-bouton regions as axon shaft.

\[
\text{LoG}_{xy}(x, y, z | R_{xy}, R_z) = \frac{4e^{-(x^2+y^2)/R_{xy}^2}}{\pi R_{xy}^4} \left(1 - \frac{x^2 + y^2}{R_{xy}^2}\right) \times \frac{e^{-z^2/R_z^2}}{\sqrt{\pi R_z}} \\
G(x, y, z | R) = \frac{e^{-(x^2+y^2+z^2)/R^2}}{(\pi)^{3/2} R^3}
\]

(2.1)
Figure 2.2: Detection of putative boutons as peaks in axon intensity profiles. **A.** Maximum intensity \(xy\) projection of an axon segment showing multiple putative boutons. Yellow line is the optimized trace of this axon. **B.** Putative boutons visible in (A) correspond to the peaks in the \(LoG_{xy}\) intensity profile (black line). Foreground peaks (cyan lines) and local background (red line) are fitted to the intensity profile as described in the text. The overall fit (thick yellow line), which is the sum of foreground peaks and background, closely matches the intensity profile. **C.** \(G\) intensity profile is obtained by sliding a fixed size Gaussian filter along the optimized trace. Small or closely positioned peaks cannot be resolved on the \(G\) profile (arrows). Peak amplitudes from the \(LoG_{xy}\) profile and background from the \(G\) profile are used to define bouton weights.

Figures 2.2A and 2.2B show that distinct putative boutons can be identified as peaks in the \(LoG_{xy}\) profile plotted against node positions along the trace, \(I^{LoG_{xy}}(s_i)\). This is because the \(LoG_{xy}\) filter
is designed to sharpen boundaries between boutons by suppressing intensity in the regions immediately adjacent to boutons. In contrast, the \( G \) filter yields a smoother profile, \( I^G(s_i) \), which is not very useful for resolving putative boutons that are in close proximity (arrows in Figure 2.2C), but is well suited for estimating shaft intensity. For these reasons, \( \text{LoG}_{xy} \) profiles were used to identify putative boutons, while \( G \) profiles were used to determine intensities of axon shafts.

Parameters of filters used to generate the axon intensity profiles, Eq. (2.1), were chosen in the following way. The \( xy \) size of the multi-scale \( \text{LoG}_{xy} \) filter was chosen to span bouton sizes observed in 2PLSM images, \( R_{xy} \in [1.5, 3.0] \) voxels (0.39 \( \mu \)m to 0.78 \( \mu \)m). Since the observed bouton size in the \( z \) dimension is dominated by the point spread function of the microscope, a single size was chosen for the \( z \) component of the filter, \( R_z = 2 \) voxels (1.6 \( \mu \)m). Upon convolution with the image, this multi-scale filter returns the maximum intensity calculated over the specified range of sizes. The Gaussian filter, \( G \), was chosen to have a fixed size of \( R = 2 \) voxels in all three dimensions. This size was selected to be roughly equal to the typical axon shaft radius observed in the images.

To automatically detect putative boutons in an \( \text{LoG}_{xy} \) profile we used an algorithm that is similar to the Backward-Stepwise Subset Selection method (James et al., 2013) Here, a varying number of foreground peaks, \( N_f \), and a constant number of background peaks, \( N_b \), was fitted to \( I^{\text{LoG}_{xy}}(s_i) \) by minimizing the following objective function of peak positions, \( \mu^f_i \) and \( \mu^b_k \), amplitudes, \( a^f_i \) and \( a^b_k \), and widths, \( \sigma^f_i \) and \( \sigma^b_k \):

\[
\text{minimize} \quad \sum_{i} \sum_{j} \left( I^{\text{LoG}_{xy}}(s_i) - \sum_{j} a^f_j \exp \left( -\frac{(s_i - \mu^f_j)^2}{2\sigma^f_j^2} \right) - \sum_{k} a^b_k \exp \left( -\frac{(s_i - \mu^b_k)^2}{2\sigma^b_k^2} \right) \right)^2
\]
The peak detection algorithm, Eq. (2.2), was initialized with a number of foreground peaks, \( N_f = \lceil L / 0.5 \mu m \rceil \), which is much larger than the expected number of putative boutons. In this expression, \( \lceil \cdot \rceil \) denotes the ceiling function and \( L \) is the trace length in micrometers. The number of background peaks, \( N_b = \lceil L / 25 \mu m \rceil \), was determined based on the observed spatial scale of background variability. The peaks were initially distributed uniformly along the entire length of the trace. Both, Gaussian and Lorentzian peak functions were tested, but only the former was used in this study as it provided a better fit to intensity profiles as judged by the value of the objective function.

The objective function was minimized with the gradient descent method. Gradient steps were taken simultaneously along the \( \mu^f_j, \mu^b_k, a^f_j, a^b_k, \sigma^f_j, \) and \( \sigma^b_k \) dimensions. At every gradient step, parameters that moved outside the bounds specified in Eq. (2.2) were set to these bounds. Upon convergence, i.e. when the relative change in the value of the objective function became less than \( 10^{-6} \), one small foreground peak was eliminated or a pair of closely positioned foreground peaks was merged, and the resulting set of peaks was re-optimized. This procedure was continued until there were no peaks left that passed the following heuristic criteria: (1) a single foreground peak is marked for elimination if the peak amplitude is less than 0.3, and (2) a pair of overlapping
foreground peaks is marked for merger if distance between the peaks is less than 1.0 μm or overlap area of the peaks exceeds 50% of their summed area.

Following the detection of putative boutons, intensity of bouton $j$ was defined as the sum of the $j$-th foreground peak amplitude and background intensity at the peak location:

$$I_{j_{Bouton}} = a_j + \sum_{k=1}^{N} a_k^b e^{-\frac{(\mu_j^k - \mu_j^b)^2}{2\sigma_j^k}}$$

(2.3)

Shaft intensity was estimated from the $G$ profile by incorporating information about the positions of detected putative boutons, Figure 2.2C. To that end, we initialized the above described peak detection algorithm with the foreground and background peaks detected on the $LoG_{xy}$ profile, but ran the algorithm on the $G$ intensity profile. Shaft intensity for a given axon was defined as the fitted background intensity on the $G$ profile, averaged over trace nodes,

$$I_{j_{Shaft}} = \left< \sum_{k=1}^{N} a_k^{b,G} e^{-\frac{(\mu_j^k - \mu_j^{b,G})^2}{2\sigma_j^k}} \right>$$

(2.4)

In this expression, index $G$ in the superscripts of peak amplitudes, $a$, positions, $\mu$, and widths, $\sigma$, was added to emphasize that these quantities are calculated based on the $G$ intensity profile.

Our goal is to use intensity profiles to extract structural information related to the physical sizes of boutons. This task is hindered by the facts that axon intensity depends strongly on expression
level of fluorescent molecules (Figure 2.1A) and microscopy conditions. Therefore, to measure unbiased structural information, intensity profiles must be properly normalized. One may consider using mean profile intensity or local shaft intensity for normalization. However, these types of normalizations can lead to errors. For example, if density of boutons varies across axons, normalization with the mean will bias boutons on higher bouton density axons towards lower intensity values. Also, if density of boutons is sufficiently large, normalization with local shaft intensity can lead to variability as the latter cannot be measured reliably between closely positioned boutons.

Our heuristic normalization approach is based on the idea that by convolving the LoG$_{xy}$ or $G$ filter with the image at a trace node position $s_i$ along axon $a$ in imaging session $t$, we obtain a quantity that is proportional to three factors: $M_t$, a factor related to imaging conditions [e.g. laser power, photomultiplier tube (PMT) voltage, and cranial window quality], $\rho_{a,t}$, volume density of fluorescent molecules, and $A_{a,t}(s_i)$, a structural factor which has been linked to axon cross-section area drawn perpendicular to the $xy$ projection of the axon centerline (Song et al., 2016):

$$\widetilde{I}_{a,t}^{\text{LoG}_{xy}, G}(s_i) = M_t \rho_{a,t} A_{a,t}^{\text{LoG}_{xy}, G}(s_i)$$

(2.5)

In creating the $\text{LoG}_{xy}$ and $G$ profiles (Figure 2.2) we rescale $\tilde{I}_{a,t}^{\text{LoG}_{xy}}(s_i)$ and $\tilde{I}_{a,t}^{G}(s_i)$ to unit means in order to minimize the effects related to imaging conditions and expression levels, thus isolating structural information:

$$I_{a,t}^{\text{LoG}_{xy}, G}(s_i) = \frac{\tilde{I}_{a,t}^{\text{LoG}_{xy}, G}(s_i)}{\langle \tilde{I}_{a,t}^{\text{LoG}_{xy}, G}(s_i) \rangle_i} = \frac{A_{a,t}^{\text{LoG}_{xy}, G}(s_i)}{\langle A_{a,t}^{\text{LoG}_{xy}, G}(s_i) \rangle_i}$$
It may be tempting to use putative bouton intensity, $I_{j}^{\text{Bouton}}$ in Eq. (2.3), which is detected based on $I_{a,j}^{\text{LoG}}(s_i)$ as proxy for bouton size. However, this may lead to bias as the denominators in Eq. (2.6) depend strongly on bouton density. To address this issue, we use axon shaft intensity detected from $G$ intensity profiles, $I_{\text{Shaft}}^{j}$ in Eq. (2.4), for normalization. The resulting quantity, referred to as bouton weight, $w_{j}^{\text{Bouton}}$, conveys structural information, which is effectively independent of the above mentioned bias,

$\frac{w_{j}^{\text{Bouton}}}{I_{\text{Shaft}}} = \frac{I_{j}^{\text{Bouton}}}{I_{\text{Shaft}}}$

(2.7)

Custom software was used to match putative boutons detected on the same set of axon segments. This was done for axons repeatedly imaged under different conditions and axons monitored in the long-term imaging experiment. Matched putative boutons were used as fiducial points to perform non-rigid registration of normalized intensity profiles across imaging sessions. For putative boutons detected in some, but not all imaging sessions the missing bouton weights were filled in with the normalized intensity profile values at the corresponding positions.

**Software implementation**

Axons of fluorescently labeled neurons were traced by using the manual tracing module of NCTracer software. Custom software written in MATLAB (BoutonAnalyzer) was used to optimize the traces, generate intensity profiles, detect putative boutons, and match these boutons across multiple user traces and imaging sessions.
Validation of LM-based bouton detection methodology with EM

**Figure 2.3:** Correlative light and electron microscopy. **A.** Maximum intensity xy projection of an image stack used for CLEM analysis. White box demarcates the region imaged with EM. Colored lines are traces of four axon segments chosen for EM reconstruction. Scale bar is 10 μm. **B.** Region outlined in (A) is shown at 4x magnification with background removed. **C.** 3D EM reconstruction of the four axon segments shown in (B). Red areas mark PSDs, and blue volumes outline mitochondria. The majority of varicosities identified in EM are clearly visible in 2PLSM images (B). **D.** Higher magnifications and different orientations of a subset of reconstructed varicosities shows that structural swellings on axons may or may not be associated with PSDs and/or contain mitochondria. Numbers in (B-D) enumerate distinct varicosities identified in EM.
Correlative light and electron microscopy (CLEM) was used to validate the described bouton detection procedure. Four axon segments were selected for this analysis (Figure 2.3A). The axon segments were imaged *in vivo* with 2PLSM, the brain tissue was fixed shortly after, and subsequently imaged with EM. Putative boutons in the 2PLSM stack of images were detected and quantified as described above (Figure 2.3B). Axons in EM images were reconstructed in 3D (Figure 2.3C) and rendered in Blender software for further analysis.

Putative boutons detected in 2PLSM images could be unambiguously matched with varicosities identified in EM (asterisks in Figure 2.4A-D and Table 2.1). Overall, 16 boutons *en passant* and 1 bouton *terminaux* could be identified in EM. The bouton *terminaux* was detected with our method, but was excluded from further analysis because it is not located on the axon centerline. With the omission of this bouton, the LM-based procedure detected 18 putative boutons with 0 false negatives and 2 false positives, both of which were small (1.1 and 2.2 in weight, #18 and #19 in Table 2.1).

We tested the idea that 2PLSM intensity is proportional to the area of axon cross-section drawn perpendicular to the *xy* projection of axon centerline (Song et al., 2016). For this, we calculated the normalized axon intensity profile, $I_{\alpha,z}^{\text{LoG}_z}(s_j) / I^{\text{Shaft}}$, and compared it to EM cross-section areas including and excluding mitochondrial cross-section areas. Both cross-section areas appear to be well correlated with normalized intensity profiles (Figure 2.4A-D). However, it is clear from visual inspection that the normalized intensity profiles do not resolve small changes in axon cross-section, which is the result of limited resolution of 2PLSM.
Figure 2.4: Normalized intensity profile is correlated with axon cross-section area, while bouton weight is indicative of bouton volume. A-D. Top: EM reconstructions of axons are shown next to the corresponding 2PLSM maximum intensity projections. Varicosities identified in EM are marked with grey asterisks, and putative boutons automatically detected based on the intensity profiles are marked with green asterisks. Middle: Axon cross-section areas, including (black) and excluding (blue) mitochondrial cross-section areas, are plotted against distance along the axons. All cross-sections are drawn perpendicular to the 2PLSM xy projections of the axon centerlines. Bottom: Normalized profiles are well correlated with axon cross-section areas. Grey regions in (B) highlight two spurious peaks in the normalized intensity profile, one resulting from a close apposition of two axons and another caused by the presence of a terminal bouton. Such regions were annotated in 2PLSM images and were excluded from all analyses. Bouton weights are highly correlated with their EM-based volumes that include (E) or exclude (F) mitochondrial volumes.

To examine the extent to which LM-based measurements provide information about bouton size, we plotted bouton weight against bouton volumes including (Figure 2.4E) and excluding (Figure
2.4F) mitochondrial volume. The results show high degree of correlation in both cases (Pearson’s \( r = 0.94 \)), supporting the idea that LM-based measurements can be used to quantify volumes of even very small varicosities (\#11, 0.07 \( \mu \text{m}^3 \)). The results also provide support for the choice of filters, Eq. (2.1), and the normalization procedure, Eq. (2.7), showing that the proposed method could overcome axon-specific differences in the expression levels and bouton density, capturing meaningful fine-scale structural information.

**Effects of imaging conditions on bouton analysis**

Next, we sought to evaluate the effects of various imaging conditions on bouton detection and measurement. To that end, a set of fluorescently labeled axons was imaged 7 times in a span of 80 minutes with different laser power (LP) and PMT voltage (see inset in Figure 2.5A). In addition, in condition E, a thin layer of agarose was applied to the cranial window to mimic deterioration of window quality which often accompanies long-term imaging experiments. In the following, we assume that there is negligible structural plasticity of boutons throughout the duration of this short-term imaging experiment, and therefore differences in bouton measurements can be attributed to the effects of imaging conditions and measurement uncertainties. The inset in Figure 2.5B shows the maximum intensity projections of an axon segment imaged in all 7 conditions. This inset illustrates that increasing (decreasing) LP and/or PMT voltage results in an overall increase (decrease) in intensity, and therefore proper normalization procedure must be used to minimize the effects of such changes on bouton measurements.
Figure 2.5: Probabilistic definition of an LM bouton based on measurement uncertainty derived from short-term imaging experiments. The same set of axons was imaged 7 times within 80 minutes with various microscope settings and cranial window conditions (inset in A). Putative boutons detected based on the first imaging session (condition A) were chosen to be the gold standard. Precision (A) and recall (B) in bouton detection were measured under the remaining conditions, B-G. Both precision and recall increase with bouton weight. While for very small boutons \( w < 2.0 \) detection is unreliable, agreement with the gold standard is achieved across all imaging conditions in 95% of boutons with weights greater than 2.0. Numbers of boutons in the gold standard are indicated next to the data points in (A). Inset in (B) shows an example of one axon segment imaged in conditions A-G. C. Bouton weights under different imaging conditions are plotted against the gold standard weight. Best fit lines show no significant bias for conditions B and C, however small, but significant reduction in mean bouton weight was observed in the remaining four conditions (all \( P < 0.03 \), two-sample t-test). Abbreviations used in the inset of B: LP is laser power in mW, PMT denotes photomultiplier tube voltage in Volts, and WC is cranial window condition, where “n” stands for normal and “a” indicates presence of a thin layer of agarose. D. CDFs for differences in bouton weights across imaging conditions. Data from all conditions were pooled. Different lines show CDFs for various intervals of mean bouton weight. E. Variance in bouton weight difference increases linearly with mean bouton weight \( \chi^2 \) linear regression with \( \text{var}(\Delta w) = \alpha\langle w \rangle \), \( P = 0.33 \), \( \alpha = 0.24 \pm 0.01 \), mean \pm s.d.). Error-bars indicate standard deviations obtained with bootstrap sampling with replacement. F. Red line shows the distribution of true bouton weight for a putative bouton of measured weight \( w = 1.5 \). Area under the curve to the right of \( w_{\text{threshold}} = 2.0 \) gives \( P(\text{bouton} | w) = 0.12 \). Large putative boutons (e.g. blue curve, \( w = 3.0 \)) have high probability of being LM boutons.
Putative boutons on 16 axon segments were detected in all conditions independently (400 putative boutons per condition on average). Custom software was used to match the same putative boutons across conditions. Putative boutons detected in condition A were chosen to be the gold standard, and precision/recall in bouton detection in the remaining conditions B-G were evaluated as functions of bouton weight (Figure 2.5A and 2.5B). The results show that for 95% of putative boutons of weight $w > 2.0$ both precision and recall equal 1 in all conditions. This number of unambiguously detected putative boutons goes up to 99% for $w > 2.5$ and 100% for $w > 3.1$. For reference, $w = 2.0$ corresponds to the weights of the two smallest varicosities identified in EM (#3 and #11 in Table 2.1), and the weight of the next smallest bouton (#17) is $w = 2.8$. Therefore, all but very small boutons can be detected with our method with high confidence.

To examine the effects of imaging conditions on bouton weight, we tested a subset of 290 putative boutons that were detected and matched across all 7 experiments. Figure 2.5C shows bouton weights in conditions B-G plotted against the corresponding weights in condition A. In spite of the fact that axon intensities in conditions B and C are drastically different from A (see inset in Figure 2.5B), the normalization procedure was able to correct for these differences (blue and red lines in Figure 2.5C). However, small ($\approx 7\%$) but significant bias was present in conditions D-G, which may have been caused by bleaching due to prolonged imaging. Figure 2.5C also reveals a considerable amount of variability in weight measurements. This variability, reflected in the $R^2$ coefficients, was similar across all comparisons, including the imaging experiments performed under the same conditions (e.g. A and G, cyan points). Therefore, bouton weight measurements are accompanied with uncertainty which is inherent to the LM-based methodology. This
uncertainty cannot be eliminated entirely, and hence, it must be explicitly incorporated into models that derive biological information from bouton measurements.

**Statistical framework for the analysis of structural plasticity of boutons**

To model the variability observed in Figure 2.5C, we examined bouton weight differences for all pairs of imaging conditions, $\Delta w = w_1 - w_2$. Because such changes clearly depend on bouton size, we looked at the statistics of $\Delta w$ in various intervals of average bouton weight, $\langle w \rangle = (w_1 + w_2) / 2$. Figure 2.5D shows the cumulative distribution functions (CDFs) of bouton weight differences in eight intervals of $\langle w \rangle$. These distributions are not significantly different from Gaussian distributions, and their variances are roughly proportional to the mean bouton weights, var($\Delta w$) = $\alpha \langle w \rangle$, Figure 2.5E. Therefore, all CDFs shown in Figure 2.5D could be standardized by rescaling the weight differences as $\Delta w / \sqrt{\alpha \langle w \rangle}$, leading to distributions that are statistically indistinguishable from the Standard Normal CDF (all $P > 0.05$, one-sample KS test).

Based on this result, we propose a statistical model in which measured bouton weight, $w$, is the sum of the true weight, $w_0$, and a noise term, $\delta$. The latter is randomly drawn from a Gaussian distribution with variance proportional to the measured weight, var($\delta$) = $\frac{1}{2} \alpha w$:

$$w = w_0 + \delta; \quad P(\delta) = \frac{e^{-\delta^2 / 2w}}{\sqrt{\pi \alpha w}}$$

(2.8)

Eq. (2.8) quantifies uncertainties in bouton weight measurements, making it possible to define bouton presence probabilistically. To that end, we impose a threshold on true bouton weight,
$w_{\text{threshold}}$, and refer to putative boutons with $w_0 > w_{\text{threshold}}$, as LM (light microscopy) boutons. In the following we set $w_{\text{threshold}} = 2.0$, which is motivated by several considerations. First, this value equals twice the average normalized axon shaft intensity. Therefore, by using $w_{\text{threshold}} = 2.0$ we are only including peaks that are substantially larger than the axon shaft intensity. Second, $w = 2.0$ corresponds to the weights of the two smallest varicosities detected in EM (#3 and #11 in Table 2.1). Finally, detection of putative boutons becomes unreliable for $w < 2.0$ (Figure 2.5A and 2.5B).

We note that as an alternative to choosing a single threshold value to define LM boutons, one could vary the threshold in a certain range (e.g. 1-3) and report results as function of this parameter.

The probability that a putative bouton of measured weight $w$ belongs to the category of LM boutons, $P(\text{bouton} \mid w)$, can be calculated based on the noise model of Eq. (2.8):

$$P(\text{bouton} \mid w) = \frac{1}{2} \left( 1 + \text{erf} \left( \frac{w - w_{\text{threshold}}}{\sqrt{\alpha w}} \right) \right)$$

(2.9)

Shaded regions in Figure 2.5F illustrate these probabilities for two putative boutons of measured weights $w = 1.5$ and $w = 3.0$. Note that even when $w$ is less than $w_{\text{threshold}}$, there is a non-zero probability that the detected peak is an LM bouton. For large $w$ (e.g. greater than 3), this probability approaches unity, and the LM bouton definition becomes virtually deterministic.
<table>
<thead>
<tr>
<th>Bouton ID</th>
<th>2PLSM measurements</th>
<th>EM measurements</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Putative bouton weight, $w$</td>
<td>$P(\text{bouton}</td>
</tr>
<tr>
<td>Axon 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>13.5</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>10.8</td>
<td>1.00</td>
</tr>
<tr>
<td>3</td>
<td>1.98</td>
<td>0.48</td>
</tr>
<tr>
<td>4</td>
<td>10.1</td>
<td>1.00</td>
</tr>
<tr>
<td>5</td>
<td>8.65</td>
<td>1.00</td>
</tr>
<tr>
<td>6</td>
<td>6.25</td>
<td>1.00</td>
</tr>
<tr>
<td>Axon 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>3.63</td>
<td>0.93</td>
</tr>
<tr>
<td>8</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>9</td>
<td>5.57</td>
<td>1.00</td>
</tr>
<tr>
<td>Axon 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>9.54</td>
<td>1.00</td>
</tr>
<tr>
<td>11</td>
<td>1.99</td>
<td>0.49</td>
</tr>
<tr>
<td>12</td>
<td>8.51</td>
<td>1.00</td>
</tr>
<tr>
<td>13</td>
<td>10.8</td>
<td>1.00</td>
</tr>
<tr>
<td>Axon 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>5.80</td>
<td>1.00</td>
</tr>
<tr>
<td>15</td>
<td>4.23</td>
<td>1.00</td>
</tr>
<tr>
<td>16</td>
<td>5.96</td>
<td>1.00</td>
</tr>
<tr>
<td>17</td>
<td>2.85</td>
<td>0.93</td>
</tr>
<tr>
<td>18</td>
<td>1.14</td>
<td>0.01</td>
</tr>
<tr>
<td>19</td>
<td>2.19</td>
<td>0.65</td>
</tr>
</tbody>
</table>

**Table 2.1:** Comparison of LM-based and EM measurements. Bouton IDs match those in Figures 3 and 4. The probability that a putative bouton belongs to the category of LM boutons, $P(\text{bouton} | w)$ was calculated according to Eq. (2.9) with $w_{\text{threshold}} = 2.0$. Bouton #8 (grey) was excluded from the analyses as it is a terminal bouton. Hyphens indicate that boutons, mitochondria, or PSDs were not detected in EM.
LM bouton definition can be used to calculate the probabilities of bouton addition, elimination, potentiation, and depression based on the measured weights in two imaging sessions (initial and final), \( w_i \) and \( w_f \):

\[
P\left(\text{added} \mid w_i \to w_f\right) = (1 - P\left(\text{bouton} \mid w_i\right)) \times P\left(\text{bouton} \mid w_f\right)
\]

\[
P\left(\text{eliminated} \mid w_i \to w_f\right) = P\left(\text{bouton} \mid w_i\right) \times (1 - P\left(\text{bouton} \mid w_f\right))
\]

\[
P\left(\text{potentiated} \mid w_i \to w_f\right) = P\left(\text{bouton} \mid w_i\right) \times P\left(\text{bouton} \mid w_f\right) \times \frac{1}{2} \left(1 + \text{erf}\left(\frac{w_f - w_i}{\sqrt{\alpha (w_i + w_f)}}\right)\right)
\]

\[
P\left(\text{depressed} \mid w_i \to w_f\right) = P\left(\text{bouton} \mid w_i\right) \times P\left(\text{bouton} \mid w_f\right) \times \frac{1}{2} \left(1 + \text{erf}\left(\frac{w_i - w_f}{\sqrt{\alpha (w_i + w_f)}}\right)\right)
\]

(2.10)

We would like to clarify that LM boutons may or may not correspond to varicosities seen in EM (Table 2.1), and the latter may not always be associated with postsynaptic densities (PSDs) and, thus, functional synapses (Shepherd and Harris, 1998). Therefore, the relationship between an LM bouton and a synapse is not deterministic and is likely to contain false-positives and false-negatives. However, the number of such errors is expected to decrease with increasing \( w \). For example, Table 2.1 shows that all LM boutons of \( w > 2.2 \) correspond to EM varicosities, and all LM boutons of \( w > 2.9 \) are varicosities associated with PSDs.

**Bouton changes can be resolved despite the uncertainty in measurements**

Next, we set out to determine if the above described bouton detection procedure can be used to identify significant structural changes in long-term, *in vivo* imaging experiments. To that end, axons of GFP labeled neurons were imaged in superficial layers of barrel cortex in 5 mice. Imaging
was performed at 4 day intervals over a 24 day period (7 imaging sessions). On average, 968 bouton sites were detected on 20 axon segments in a given animal and imaging session. These sites were tracked over the duration of the experiment to quantify bouton addition, elimination, and weight changes as compared to the initial state of the circuit. The short-term imaging experiment of Figure 2.5 was used as control. Here, conditions B-G were compared to condition A, and the results were pooled.

**Figure 2.6:** Structural change in boutons can be resolved in long-term *in vivo* imaging experiments despite measurement noise. **A.** Fraction of boutons that are absent on day 0 and are present at a later time with joint probability of 0.95 or greater (significant bouton addition). Red point (error-bars are too small to be visible) shows this fraction in images acquired within 80 minutes (conditions A-G). Black points show the results for a long-term imaging experiment. Statistically significant fraction of added boutons is detected after 4 days (interval between imaging sessions), and this fraction grows with time, consistent with the idea of gradual modification of the initial circuit. Similar trends were observed for the fractions of significant bouton eliminations (**B**) and significant bouton weight changes (**C**). Dashed red lines in (**A-C**) indicate baseline circuit changes expected from the statistical model. Error-bars indicate standard deviations based on Poisson statistics.

Figure 2.6A shows the fraction of added LM boutons over time, as compared to the first imaging session. Specifically, we only consider significant bouton addition events, i.e. those for which the probability of addition according to Eq. (2.10) is greater than 0.95. Analogous plots for the fractions of significant bouton eliminations and significant bouton weight changes (potentiation
and depression combined) are shown in Figures 2.6B and 2.6C respectively. As was expected, the fractions of significant changes in the short-term imaging experiment (red points in Figure 2.6) are at the chance levels (dashed red lines). The latter was obtained with a bootstrap procedure in which the weights of individual boutons were independently shuffled across conditions. In contrast, in the long-term imaging experiment, the fractions of significant bouton additions, eliminations, and weight changes are significantly larger than chance already by the second imaging session (after 4 days), and these fractions continue to increase with time consistent with the idea of gradual modification of the circuit. These results prove that the described methodology can be used to quantify circuit changes, despite the numerous challenges associated with long-term, in vivo imaging experiments.

**Discussion**

Detection of structural changes in boutons is hindered by various technical challenges and fundamental limitations of light microscopy. Unfortunately, any uncertainty that enters into the analysis of long-term in vivo imaging data manifests itself as spurious structural plasticity. Therefore, it is important to account for all sources of errors, minimize their effect, and incorporate the residual uncertainty into the interpretation of results. Below, we describe various sources of errors affecting bouton measurements.

First, fluorescence based measurements provide only indirect evidence of bouton size. Therefore, such measurements need to be validated by showing that they are informative of bouton structures. In this study, we used CLEM to show that bouton weight, defined based on 2PLSM data, is well correlated with bouton volume (Figure 2.4E and 2.4F). Second, variability in expression levels of fluorescent proteins across axons and within axons over time makes it difficult to compare boutons on different axons and to identify true structural changes. Here, these problems were mitigated by
a specifically designed normalization procedure (Figure 2.2), which was verified with CLEM (Figure 2.4) and was tested in the short-term imaging experiment (Figure 2.5A-C). Third, because the resolution of 2PLSM (≈ 0.3 µm in xy) is not much smaller than bouton size (≈ 1 µm for small boutons in Figure 2.3), there are inherent uncertainties in detection of small boutons, differentiation of closely positioned boutons, and measurement of bouton size changes. These uncertainties of the imaging method cannot be eliminated, and, therefore, they must be modeled to ensure that bouton measurements are interpreted correctly (Figure 2.5D-F).

Many other technical issues can add to the uncertainty and introduce bias in bouton measurements. For example, deterioration of cranial window quality with time, deformation of brain tissue, small changes in brain orientation relative to the microscope, and slight movement of boutons along axons can lead to the perception of increased plasticity. In contrast, limited temporal resolution of long-term imaging experiments, typically few days between imaging sessions, can lead to an underestimate of plasticity, as changes that occur between imaging sessions remain unaccounted. Finally, uncertainty can arise from the computational method used to detect boutons. However, we have verified that the uncertainty of the method described in this study is small and can be ignored in light of contributions from other sources (Figure 2.5C).

It is important to emphasize that not every putative bouton detected in LM will correspond to a varicosity if imaged with EM. This is particularly so for very small putative boutons (w < 2.0, bouton volume < 0.07 µm³) which often result from noise in fluorescence or reflect small, non-synaptic structural swellings on axons. In the absence of a reliable synaptic marker, the only reasonable way to eliminate such functionally irrelevant putative boutons is by imposing a threshold on bouton weight. In this study, we refer to putative boutons whose true weight exceeds the threshold as LM boutons and suggest setting this threshold at 2.0 based on CLEM and short-
term imaging experiments. In practice, one may vary the threshold in a small range to ensure that specific conclusions are robust to the choice of this parameter.

Using a threshold to define an LM bouton as an all-or-none entity is problematic since the true bouton weight is unknown due to measurement uncertainty. In particular, for bouton weights close to the threshold, small fluctuations from one imaging session to the next may be misinterpreted as structural plasticity. Therefore, we define an LM bouton probabilistically, with weight-dependent probability deduced from the short-term imaging experiment (Figure 2.5D-F). This description can be extended to quantify bouton addition, elimination, potentiation, and depression, as well as to establish significance levels for such structural changes, Eq. (2.10). To illustrate the feasibility of this approach, we applied it to a dataset of 4840 bouton sites tracked over 24 days. Figure 2.6 shows that significant changes in LM boutons can be detected in long-term imaging experiments. However, it remains to be seen if the detected changes are informative of circuit alterations that subserve the many functions of the brain.
Chapter 3. Anatomical correlates of perceptual learning

Background

Neocortical circuits in the mammalian brain are known to be plastic, capable of changing in response to salient experiences. These adaptations may be quantified in terms of wiring and weight changes in the circuit (Chklovskii et al., 2004). While dendritic and axonal arbors of adult neurons are thought to be largely stable under normal circumstances (De Paola et al., 2006; Majewska et al., 2006; Mostany and Portera-Cailliau, 2011; Stettler et al., 2006; Trachtenberg et al., 2002), neural circuits retain tremendous capacity for plasticity at the synaptic level (Holtmaat and Svoboda, 2009; Hubener and Bonhoeffer, 2014).

Although each pyramidal excitatory neuron in the mouse neocortex, a type that makes up nearly 80% of all neurons in the cortex, receives and establishes ~10⁴ synapses, a given pair of pyramidal neurons is connected with only ~10 synapses (Feldmeyer et al., 1999; Hill et al., 2012; Markram et al., 1997b; Markram et al., 2015; Reimann et al., 2017; Silver et al., 2003). Thus, neuronal connectivity can be altered by adding, removing or modifying the efficacy of a relatively small number of synapses.

Within the neocortex, the primary somatosensory area preferentially receives and processes information from the sense of touch. In rodents such as mice, a large portion of the somatosensory cortex is devoted to processing tactile input from their whiskers. Input from individual whiskers is sent to anatomically distinct regions within the somatosensory cortex called barrels (Diamond et al., 2008; Petersen, 2007). Previous experiments have shown that neurons in this area demonstrate plasticity that depends on whisker mediated experiences of the animal (Feldman and Brecht, 2005). Directly stimulating neurons in this region is thought to produce subjective whisker touch related
percepts in the animal (see e.g. (Leal-Campanario et al., 2006)). Moreover, mice can learn to perceive direct stimulation of just a few neurons in the somatosensory cortex (Doron and Brecht, 2015; Houweling and Brecht, 2008; Huber et al., 2008). Since mechanisms underlying neuronal plasticity are thought to involve use-dependent modifications of synapses, monitoring synaptic structures along targeted neurons in a time-lapse manner as the animal learns to recognize novel percepts is a promising way to study such synaptic modifications (Holtmaat and Caroni, 2016; Huber et al., 2008).

In the experiment described in this chapter, a sparse population of layer 2/3 pyramidal neurons in mouse somatosensory cortex was labeled with green fluorescent protein (GFP) and a light gated ion channel, channelrhodopsin (ChR). A surgically implanted cranial window provided optical access through which labeled neurons could be stimulated with light. The animals were trained to report perception of this optical microstimulation, and axons of these neurons were periodically imaged over a period of nearly two months. Structural changes of en passant boutons along the labeled axons were quantified based on the long-term, time-lapse imaging experiments. Our preliminary results indicate that while overall number and strength of synapses remains stable over the duration of the experiment for all animals, mice that learned to perform the perceptual task had more plastic circuits compared to those that did not learn the task.

2PLSM imaging and behavioral training experiments were performed by our collaborators\(^3\), and a detailed description of the methods can be found in (Lebrecht, 2016). Below, I only provide a

---

\(^3\) D. Lebrecht\(^a\), M. Dimanico\(^a\), S. Pages\(^a\), K. Morandell\(^a\), D. Huber\(^a\) and A. Holtmaat\(^a\)

\(^a\) University of Geneva
simplified overview of the experiment, and describe some preliminary results from ongoing work on this project.

**Experiment design**

Details regarding mice, surgical procedures, and imaging can be found in the *In vivo* imaging section of chapter two. In addition to expressing GFP as described in that section, the same population of neurons was engineered to express membrane bound ChR. Mice were put on a water deprivation schedule prior to training on behavioral tasks. The behavioral tasks consisted of different two-alternative forced choice tasks (Carandini and Churchland, 2013). In these tasks, they were required to report the presence or absence of a stimulus via a licking behavior to receive a water reward. Two lick ports were placed on either side of the snout of head-fixed mice to enable the animals to report a response in each trial. The start of every trial within a training session was signaled with a visual cue. Mice were required to respond to the perceived presence (absence) of stimulus by licking the port placed on the left (right) side. The correct response was rewarded with a small amount of water at the corresponding port.

In the first of these tasks, referred to as the auditory task, the mice were trained to report the presence or absence of a sound stimulus. The second task, referred to as the auditory with random microstimulation task, was the same as the auditory task except that optical microstimulation was delivered in a random subset of trials. The last of the tasks, referred to as the microstimulation task, required mice to report perception of optical microstimulation. Licks were recorded and water was dispensed using custom electronic equipment. For each mouse, the fraction of correct responses in each training session is used as the measure of its performance at the task. Imaging was performed at 4-day intervals, starting from ~8 days before training on the auditory task commenced.
Figure 3.1: Timelines for different phases of the experiment in 5 animals. (A-E) Imaging sessions, separated by 4-day time intervals, are indicated by square markers. The first training session for the auditory, auditory with random microstimulation, and microstimulation tasks are indicated by red, yellow and green vertical lines. Performance at the microstimulation task is measured as the fraction of correct responses in each training session (gray circles). The performance rises above chance level (0.5) for 3 of the 5 animals (A-C). The performance data is fit with sigmoidal functions to estimate the start, inflection point and end of learning for these animals. The performance of 2 animals remained at the chance level till the end of the experiment (D-E).

Quantifying behavioral performance

The fractions of correct responses in the microstimulation task for each of the daily training sessions are shown in Figure 3.1 for the 5 animals considered in the results of this chapter (see
Table 3.1). Choosing a lick port randomly in each trial would lead to the correct choice being made by chance in half of the trials. As expected, all animals performed at roughly this chance level at the start of training on the microstimulation task. 3 out of the 5 animals learned to perform the task, as evidenced by their ability to choose the correct port with ~80% success rate towards the end of the experiment (Figure 3.1A-C). These animals are subsequently referred to as the learners. There was no substantial improvement in performance with training for the remaining 2 animals, and these animals are referred to as non-learners in the following. For the three learners, the fraction of correct responses as a function of time was fit with a sigmoid curve, Eq. (3.1).

\[
L(t) = a + \frac{b}{1 + e^{\frac{t - \mu}{\tau}}}
\]

(3.1)

The inflection point of the sigmoid fits \( t = \mu \) and times indicative of the start and end of improvement in performance \( t = \mu \pm 3\tau \) are shown for the learners in Figure 3.1.

**Dataset of bouton structural dynamics**

Methods described in chapter two were used to analyze the long-term imaging data. Briefly, for each animal the same set of axon segments was manually traced in image stacks acquired at different times. Traces were optimized and used to generate intensity profiles of the axons. Boutons were detected automatically as peaks along the intensity profiles, and tracked in profiles corresponding to different imaging sessions using a greedy algorithm. All matches were validated by visually inspecting maximum intensity projections of the axons, Figure 3.2. A summary of the dataset generated in this manner is provided in Table 3.1.
Figure 3.2: Matching boutons in time-lapse images. A. xy projection of an axon segment that was imaged at 4-days intervals over a period of 68 days. Image intensity in each session is normalized independently as described for intensity profiles in chapter two. Colored lines link corresponding boutons. These lines do not always run in parallel due to small rotations of the brain, nonlinear tissue distortions, and movements of boutons. B. Normalized and aligned intensity profiles of the axon from A. Arrows indicate two locations on the axon where a bouton present on day 0 was eliminated after few imaging sessions.
Table 3.1: Dataset summary. Boutons were tracked in 8 animals. The imaging quality deteriorated rapidly for the last 3 animals (grayed out), and these animals are excluded from the analyses presented in this chapter.

Below we describe preliminary results based on analyses of three learners (DL083, DL085, and DL101), and two non-learners (DL088, and DL102)

**Bouton number and weight are homeostatically regulated**

The total number and weight of boutons remains steady over the duration of the long-term imaging experiments. Figure 3.3A-B illustrate the facts that these quantities are stable even for individual axon segments. The distribution in bouton weights is also conserved in all imaging sessions, as shown for one of the learners in Figure 3.3C. Similar trends were observed for all animals. The overall bouton number and strength density, though stable over time, seems to differ across animals in Figure 3.3D-E. Bouton length density is known to depend on the type of axon (De Paola et al., 2006), and may also depend on distance of the axon segments from the cell body and cortical layer in which the axons lie. Such relationships could also affect bouton weight distributions.
Figure 3.3: Homeostasis in bouton number and bouton weight. A. Length density of bouton number and B. bouton weight on all axons (distinct shades of gray) for one animal (DL083) as a function of time relative to the first day of the auditory with random microstimulation task. C. The distribution of bouton weights across imaging sessions (colors) is stationary. D. Overall bouton number density and E. weight density for 5 animals included in this study. Mean values across imaging sessions are indicated by horizontal lines for each animal (colors). F. Distribution of bouton weights for each animal, obtained by pooling data from all imaging sessions. Legend in (F) applies to panels (D-F).
Temporal and spatial correlations in bouton structural changes

**Figure 3.4:** Bouton weight changes are anti-correlated at 4 day lag. The autocorrelation function of bouton weight changes shows a distinct negative correlation at a lag of 4 days. Thus, boutons that potentiate (depress) in the first 4-day interval are more likely to depress (potentiate) over the next 4 days. Chance level autocorrelations for each animal are shown by points connected with dashed lines.

We wondered if there are any general statistical patterns in the temporal dynamics of boutons. The autocorrelation function of bouton strength changes shows a distinct negative correlation at a lag of 4 days. This means thatboutons that potentiate during the first 4-day interval are more likely to depress over the next 4 days and vice-versa. Bootstrap randomization of strength changes over time produces ~ 0 autocorrelations at non-zero lags (dashed lines).

Nearby boutons on the same axon are known to be correlated in size (see (Bartol et al., 2015)) and may also potentiate/depress in a correlated manner. To see if such correlations are present in our dataset we calculated Pearson correlation coefficient for bouton weight (Figure 3.5A) and bouton weight changes (Figure 3.5B) for every bouton pair on a given axon as a function of inter-bouton distance. The positions of sets of all highly correlated boutons (correlation coefficient > 0.8) are shown for a learner in Figure 3.5C.
Figure 3.5: Bouton structures along the same axon are correlated. Correlations in bouton weights (A) and bouton weight changes (B) as functions of distance along the axon show that boutons separated by <30 μm along the axon are correlated. C. Sets of highly correlated boutons (pairwise correlation coefficient > 0.8) along the same axons in distinct colors for one animal (DL083). All other gray dots mark position of the remaining boutons. There is an over-representation of highly correlated boutons at small separations (e.g. black circles). Trace portions devoid of boutons show regions that were excluded from the analysis due to presence of cross-overs, terminal boutons, blood vessels, or because the region was not clearly visible in at least one of the imaging sessions.

This illustration suggests that boutons with highly correlated weights are clustered along axons, and this is reflected by the positive spatial correlations at small inter-bouton distances in Figure 3.5.
3.5A. Since bouton weights are correlated over time, it is not surprising that bouton weights changes are also correlated over a similar spatial scale.

**Bouton addition and elimination across animals**

![Figure 3.6](image)

Figure 3.6: Fraction of boutons added to the circuit in different time intervals. A-E. We evaluated the fraction of boutons added to an initial circuit by considering the circuit at all later times, i.e. final circuit. The key events in the experimental timeline are shown for all animals. The fraction of boutons added is generally larger for all learners (A-C), compared to non-learners (D-E). Within the group of learners, the fraction of added boutons seems to rise around the inflection point of the learning curve (black outlines), suggesting that circuit modification by means of bouton addition plays a role in improvement in behavioral performance at the perceptual task.

Addition, elimination, and clustered dynamics of dendritic spines have been identified as forms of experience dependent structural plasticity (Fu and Zuo, 2011; Lu and Zuo, 2017; Yuste and
Bonhoeffer, 2001). Relatively less is known about the experience dependent structural plasticity of axonal boutons (Gogolla et al., 2007). Therefore, we explored structural dynamics of axonal boutons in the attempt to identify anatomical correlates of perceptual learning. Bouton additions and eliminations were calculated probabilistically, as described in chapter two. The fraction of boutons added was calculated as the number of boutons absent in the initial circuit and present in the final circuit divided by the average number of boutons in the initial and final circuits. The fraction of boutons eliminated was calculated in an analogous manner. Since circuit modifications related to perceptual learning may occur before, during, or even after improvement in behavioral performance, we measured bouton addition and elimination in all possible time intervals by considering all temporally ordered pairs of imaging sessions (maps in Figures 3.6 and 3.7). Boutons that are randomly added to or removed from a largely stable circuit are expected to contribute to the maps uniformly. If a new circuit is built, or an existing one is dismantled, the systematic addition or removal of boutons is expected to accumulate as the time interval increases.

Figures 3.6 and 3.7 show that the fractions of boutons added and eliminated are generally larger for the learners compared to the non-learners. For the learners, there is a noticeable increase in bouton additions that commences close to the inflection in their performance (black outlines in Figure 3.6A-C). Among the learners, systematic elimination of boutons also occurs close to the performance inflection point in DL083 and DL101, but this trend is not observed in DL085.
Figure 3.7: Fraction of boutons eliminated from the circuit in different time intervals. A-E. We evaluated the fraction of boutons eliminated from the initial circuit by considering the state of the circuit at all later times. Two of the learners, DL083 and DL101 (A, C), show distinctly larger fractions of eliminated boutons compared to non-learners (D-E). However, bouton eliminations mostly occur for initial circuits defined at a time before the start of the microstimulation only task (green horizontal and vertical lines). We do not see a clear trend in bouton eliminations that may be related to behavioral performance of the animal.

Discussion

Bouton structural changes associated with a particular experience are expected to be subtle. Therefore, one must monitor a large population of boutons over a long period of time to distinguish such changes from baseline circuit plasticity. However, imaging artefacts such as those related to cranial window quality and photo bleaching of fluorescent molecules tend to become more severe
over time, which makes it challenging to identify experience related bouton structural changes in time-lapse \textit{in vivo} imaging experiments.

In the experiment described in this chapter, each learner takes a different amount of time to show an improvement in performance once they start being trained on the microstimulation task. The rate of improvement in performance is also different for each learner. Such variability in performance across individual learners may be attributed to various factors. For example, the ability to perceive microstimulation may depend on number of cells expressing ChR in each animal, and frequency and regularity of microstimulation-evoked action potentials in ChR expressing cells (Doron and Brecht, 2015). Moreover, connectivity of stimulated cells within pre-existing circuits, and thus their ability to recruit cell assemblies [see (Holtmaat and Caroni, 2016; Poo et al., 2016)], is different across animals. These differences could determine whether the animals learn the task at all, in addition to characteristic features of individual performance curves for the learners. Structural plasticity in the somatosensory cortex related to learning may occur before or even after behavioral performance rises above chance level. For example, improvement in performance may require reconfiguration of brain areas downstream of the somatosensory cortex (de Lafuente and Romo, 2005). Consequently, one may expect circuit modifications in the somatosensory cortex to occur prior to noticeable improvement in performance. Alternatively, circuit changes in the somatosensory cortex may serve to mainly sharpen perception of microstimulation once the animal has worked out the correct strategy to obtain water rewards. In this case, structural changes in the somatosensory cortex would take place after performance improvement is observed.
Our preliminary results suggest that addition of boutons to the circuit, which occurs in a stereotypic manner across the learners, is a possible correlate of perceptual learning. However, further analyses are required to arrive at a conclusion.
Chapter 4. Effects of homeostatic constraints on statistical features of synaptic connectivity

Background

Learning and long-term memory formation in the brain are accompanied with changes in the patterns and weights of synaptic connections [see (Bailey and Kandel, 1993; Chklovskii et al., 2004; Holtmaat and Svoboda, 2009) for review]. Yet, a detailed understanding of the effects of learning on synaptic connectivity is hindered by an insufficient account of network activity patterns and cell-type specific, experience-dependent learning rules. Thus, it is currently not feasible to directly relate the learning experience of an animal to specific changes in its synaptic connectivity. As an alternative, one may look for basic statistical features of synaptic connectivity which are catalyzed by the learning process, develop over time, and are present in adult circuits. In this study, we examine a biologically motivated, exactly solvable model of associative learning to identify such connectivity features in local cortical circuits. Inspired by the ideas introduced by Gardner and Derrida (Gardner and Derrida, 1988) and further developed by Brunel et al. (Brunel et al., 2004), we hypothesized that a given local circuit of the adult cortex is functioning in a steady-state. In this state the associative memory storage capacity of the circuit is maximal (critical) (Cover, 1965; Gardner, 1988; Gardner and Derrida, 1988; Hopfield, 1982), and learning new associations is accompanied with forgetting some of the old ones (Figure 4.1).
Figure 4.1: Associative memory storage in local cortical circuits. A. A cortical column contains many classes of inhibitory (circles) and excitatory (triangles) neurons. B. By adjusting the weights of their presynaptic connections, $J$, neurons in the column can learn to associate certain input patterns, $X(t)$, with particular outputs, $y(t+1)$. Such changes in the connection weights are constrained by the homeostatic regulation of the overall weight, $N_{w1}$, or number, $N_{w0}$, of non-zero weight connections, as well as by the excitatory/inhibitory nature of individual presynaptic inputs, vector $g$. C. A neuron’s ability to learn a set of presented associations decreases with the number of associations in the set, $m$. The transition from perfect learning to inability to learn an entire set becomes very sharp with increasing number of potential inputs, $N$. Our numerical simulations (dots) are in good agreement with theoretical results (solid lines) (Cover, 1965). Critical capacity, $\alpha_c$, is defined as the number of associations per potential input, $m/N$, which can be learned with 0.5 probability of success. This capacity in the unconstrained perceptron model is 2.
The steady-state learning hypothesis is supported by computational studies conducted in the cerebellar (Barbour et al., 2007; Brunel et al., 2004) and cerebral (Chapeton et al., 2012) cortices. It is also consistent with recent experimental evidence from human subjects, showing that new learning and memory retrieval can be accompanied by forgetting (Kuhl et al., 2010; Wimber et al., 2015). Thus, ongoing activity in the brain is likely to present neurons with sets of associations that are much larger than the learning capacity of the neurons. Consequently, the neurons will learn (presumably in development) as much as they possibly can and from then on (throughout adulthood) will remain at their critical memory storage capacity. This seemingly trivial hypothesis has very powerful implications. Because the properties of the network in the steady-state are independent of the learning path taken by the network to reach that state, one can analyze the steady-state connectivity in the absence of detailed knowledge of the animal’s experience or the learning rules involved. This will remain true as long as the animal’s experience is rich enough to present the network with a number of associations that exceeds the network’s capacity, and the learning rules are versatile enough to learn the critical number of associations within the developmental period. Due to the fundamental nature of this hypothesis its predictions are expected to hold for many species, brain areas, and learning conditions.

The study of associative learning by artificial neural networks has a long history dating back to the work of McCulloch and Pitts who introduced one of the first binary neuron models (McCulloch and Pitts, 1943). Rosenblatt later showed that such a binary neuron, now termed the perceptron, can solve classification problems by learning to associate certain input patterns with specific outputs (Rosenblatt, 1957, 1958). The memory storage capacity of a simple perceptron (no constraints, \( h = 0 \) or learnable threshold, \( \kappa = 0 \)) was calculated by Cover (Cover, 1965), who used geometrical arguments to show that a simple perceptron can learn to associate \( 2^N \) unbiased input-
output patterns ($f_{in} = f_{out} = 0.5$). It was Hopfield who recognized that stable states in recurrent networks of binary neurons can be used as a mechanism for memory storage and recall (Hopfield, 1982). Subsequently, a general framework for the analysis of memory storage capacity was established by Gardner and Derrida (Gardner, 1988; Gardner and Derrida, 1988) who used the replica theory, originally developed for spin glass applications (Edwards and Anderson, 1975; Sherrington and Kirkpatrick, 1975), to solve the problem of robust learning ($\kappa > 0$) of arbitrarily biased associations. To model granule to Purkinje cell connectivity in the cerebellum, Brunel and colleagues constrained Gardner’s solution by fixing the firing threshold and forcing the inputs to be all excitatory ($J_j \geq 0$) (Brunel et al., 2004). These results were then extended by Chapeton et al. (Chapeton et al., 2012) on the case of excitatory and inhibitory inputs and applied to cortical circuits.

In what follows, we extend the steady-state learning model described in (Chapeton et al., 2012) by considering multiple classes of excitatory and inhibitory neurons and by incorporating biologically motivated homeostatic constraints. There is emerging evidence suggesting that despite circuit changes that accompany learning, individual cells may regulate (i) the total number ($l_0$ norm) and/or (ii) the overall weight ($l_1$ norm) of their presynaptic inputs. For example, it has been shown that the numbers of excitatory and inhibitory synapses onto excitatory cells in the adult cortex remain constant over periods of many days to weeks (Brown et al., 2009; Chen et al., 2012; Fuhrmann et al., 2007; Hofer et al., 2009; Holtmaat et al., 2006; Holtmaat et al., 2005; Kim and Nabekura, 2011). Similarly, it has been shown that synapse loss can be counterbalanced by enlargement of other synapses, such that the summed synaptic surface area per length of dendritic segment remains constant across time and conditions (Bourne and Harris, 2011). In addition, long-term imaging studies have reported that total spine volume, as measured by normalized brightness,
remains constant over days (Holtmaat et al., 2006; Kim and Nabekura, 2011). Because spine volume is correlated with synaptic weight (Arellano et al., 2007; Harvey and Svoboda, 2007; Matsuzaki et al., 2004; Zito et al., 2009), these findings suggest that the overall weight of presynaptic inputs remains constant throughout learning.

Below, we provide a detailed formulation of the homeostatically constrained steady-state learning model. The model was solved analytically and the solution was validated numerically. The results were compared with many published experimental studies reporting probabilities of connections and distributions of connection weights for various classes of excitatory and inhibitory neurons in the cerebellum, neocortex, and hippocampus.

In the following we formulate a theoretical model of steady-state learning, which incorporates various classes of neurons and biologically inspired constraints. Related models, which only include some of the constraints considered here, were previously described in a number of studies [see e.g. (Amit et al., 1989a; Brunel et al., 2004; Chapeton et al., 2012; Cover, 1965; Edwards and Anderson, 1975; Gardner, 1988; Gardner and Derrida, 1988; Sherrington and Kirkpatrick, 1975; Viswanathan, 1993)]. Detailed description of theoretical and numerical methods can be found in Appendix C.

**Biologically constrained model of steady-state learning**

Networks in the cortex are thought to be organized in columnar units. Such units may include various functional (Hubel and Wiesel, 1963, 1977) and structural columns (Lübke and Feldmeyer, 2007; Stepanyants et al., 2008), which are typically few hundred micrometers in radius. Analyses of neuron morphology (Binzegger et al., 2004; Kalisman et al., 2003; Shepherd et al., 2005; Stepanyants et al., 2008) have shown that the mesh created by the axonal and dendritic arbors of
cells within such units contains numerous micron-size axo-dendritic appositions, which are called potential synapses. A pair of potentially connected cells can form a synaptic connection through local structural synaptic plasticity (Escobar et al., 2008; Stepanyants et al., 2002; Trachtenberg et al., 2002). Though nearby neurons (e.g. separated by less than 50 µm) within cortical units are typically interconnected in terms of potential synapses, functional synaptic connectivity is invariably sparse (Thomson and Lamy, 2007). For this study, we consider two cells to be functionally connected if an action potential fired by the presynaptic cell elicits a detectable response in the cell body of the postsynaptic neuron. Such a response, measured as a deviation of the membrane potential from its resting value, is referred to as a unitary postsynaptic potential (uPSP). The sign of a uPSP in a cortical neuron is dependent on the class of the presynaptic cell; it is positive if the presynaptic cell is excitatory (uEPSP) and negative if it is inhibitory (uIPSP).

We consider a local cortical network involved in an associative learning task (Figure 4.1A). The network may contain various excitatory and inhibitory neuron classes which are characterized by distinct firing probabilities. The state of the network at time $t$, $X(t)$, is described by the binary (0 or 1) activities of all neurons. The network must learn to associate this state with the subsequent network state $X(t+1)$, and that to the state at the following time step, $X(t+2)$, etc., thus learning a chain of associated network states, $X(t) \rightarrow X(t+1) \rightarrow \ldots \rightarrow X(t+m)$. Assuming the successive network states to be uncorrelated one can reduce the problem of network learning to the problem of learning by individual neurons (Figure 4.1B).

Thus, we consider a single model neuron, which receives $N$ potential inputs from $N$ potentially presynaptic partners and is faced with a task of learning a set of $m$ input-output associations. The inputs, enumerated with index $j$, may come from various excitatory and inhibitory neuron classes
which have characteristic firing probabilities, \( f_i \). The model neuron is motivated by the McCulloch and Pitts model (McCulloch and Pitts, 1943):

\[
\theta \left( \sum_{j=1}^{N} J_{j}X_{j}^{\mu} - h \right) = y^{\mu}, \quad \mu = 1, \ldots, m
\]

(4.1)

Here, \( J_{j} \) is the weight of presynaptic input \( j \), \( h \) is the firing threshold of the neuron, and \( \theta \) denotes the Heaviside function. The inputs, \( X_{j}^{\mu} (\mu = 1, \ldots, m) \), and outputs, \( y^{\mu} \), are binary and their values are randomly drawn from neuron-class dependent probability distributions: 0 with probability \( 1 - f_i \) and 1 with probability \( f_i \). The term \( \sum_{j=1}^{N} J_{j}X_{j}^{\mu} \) plays the role of the postsynaptic potential, and the neuron fires when this potential exceeds \( h \). Eq. (4.1) can be rewritten as a set of inequalities:

\[
(2y^{\mu} - 1) \left( \sum_{j=1}^{N} J_{j}X_{j}^{\mu} - h \right) > 0, \quad \mu = 1, \ldots, m
\]

(4.2)

For this work, we impose the following biologically inspired constraints on the learning of associations \( \{ X_{j}^{\mu}, y^{\mu} \} \):

The weights of presynaptic inputs, \( J_{j} \), are sign-constrained in a way that is determined by the class of individual inputs,

\[
J_{j}g_{j} \geq 0, \quad j = 1, \ldots, N
\]

(4.3)
In these inequalities, \( g_j = 1 \) if input \( j \) is excitatory and \( g_j = -1 \) if it is inhibitory.

The weights of input connections are also constrained to have a fixed norm. In the following we restrict the analysis to two cases: (i) \( l_0 \) norm constraint, which corresponds to learning with a fixed number of non-zero weight inputs and is defined in the limit, \( \lim_{l \to 0} \sum_{j=1}^{N} |J_j| = Nw_0 \), and (ii) \( l_1 \) norm constraint, which corresponds to learning with a fixed overall magnitude of the input weights, \( \sum_{j=1}^{N} |J_j| = Nw_1 \). In these expressions, \( w_0 \) is referred to as the overall connection probability, while \( w_1 \) is the average absolute connection weight. For conciseness, the \( l_0 \) and \( l_1 \) norm constraint can be combined into a single equation:

\[
\sum_{j=1}^{N} |J_j| = Nw_l, \quad l = 0, 1
\]

(4.4)

The firing threshold of the neuron, \( h \), is fixed and does not change during learning.

Associations, \( \{X_{j}^{\mu}, y^{\mu}\} \), must be learned robustly, which means that the postsynaptic potential must be somewhat above (below) the firing threshold if \( y^{\mu} = 1 \) (0). This imposed minimal deviation from the threshold is referred to as the robustness parameter, \( \kappa \geq 0 \). To incorporate this parameter, we modify the r.h.s of Eq. (4.2), making the inequalities more stringent:

\[
(2y^{\mu} - 1) \left( \sum_{j=1}^{N} J_j X_j^{\mu} - h \right) > \kappa, \quad \mu = 1, \ldots, m
\]

(4.5)
To summarize, the full model can be reduced to the following:

\[
\left(2y^\mu - 1\right)\left(\sum_{j=1}^N J_j X_j^\mu - h\right) > \kappa, \quad \mu = 1, \ldots, m
\]

\[
J_j g_j \geq 0, \quad j = 1, \ldots, N
\]

\[
\sum_{j=1}^N |J_j|^l = N w_l, \quad l = 0, 1
\]

\[
\text{Prob}\left(X_j^\mu\right) = \begin{cases} 1 - f_j, & X_j^\mu = 0 \\ f_j, & X_j^\mu = 1 \end{cases}; \quad \text{Prob}\left(y^\mu\right) = \begin{cases} 1 - f_{out}, & y^\mu = 0 \\ f_{out}, & y^\mu = 1 \end{cases}
\]

(4.6)

Any set of connection weights, \(J_j\), which satisfy Eq. (4.6) is a valid solution of model.

**Model assumptions and approximations**

The steady-state learning model relies on several assumptions and approximations. Here we describe these assumptions and provide experimental evidence supporting the approximations made:

1. We discretized time into finite-size bins and describe the activity of neurons in the network with binary values: 1 if a neuron is firing and 0 if a neuron is silent. This approximation is reasonable so long as one can choose an integration window which is larger than the duration of a typical upSPS (\(\tau\)), yet small enough not to encompass successive action potentials fired by any given cell. Denoting the typical firing rate for a cell class with \(r\), such binning of activity should be possible when \(r \times \tau\) is smaller than one. In fact, many classes of neurons maintain in vivo firing rates that are low enough for this condition to be valid. Specifically, uEPSPs and uIPSPs in pyramidal cells typically have \(\tau\) in the 40-60 ms range (Gonzalez-Burgos et al., 2005; Lefort et al., 2009; Markram et al., 1997a; Sayer et al., 1990; Sun et al., 2006), while the
spontaneous firing rates of these cells in vivo are \( r \sim 1-3 \) Hz [(Csicsvari et al., 1999; Hromadka et al., 2008; Puig et al., 2003; Yazaki-Sugiyama et al., 2009), also see (Barth and Poulet, 2012) for review]. These observations put cortical pyramidal cells well within the range of validity of the above approximation. For inhibitory cells the data is more variable, but generally also supports the approximation. For example, the reported spontaneous firing rates in vivo are 9.2 Hz for FS cells in mouse visual cortex (Yazaki-Sugiyama et al., 2009), 7.6 Hz for FS cells in cat striate cortex (Azouz et al., 1997), about 3 Hz for PV cells and <1 Hz for SOM cells in mouse visual cortex (Ma et al., 2010), and 13-14 Hz in CA1 interneurons of rat hippocampus (Csicsvari et al., 1999). We note that it is not clear if the activity of neurons during associative learning resembles low firing rate spontaneous activity, or whether it is similar to the bursting activity of subsets of neurons recorded in animals actively engaged in trained behaviors. Nonetheless, because the fraction of bursting neurons at any given time is small (Barth and Poulet, 2012), the average network firing rate is expected to be low. For example, in vivo imaging studies, in which the activities of large ensembles of cortical neurons are monitored over time, have reported population average firing rates of < 1 Hz (Golshani et al., 2009; Greenberg et al., 2008; Kerr et al., 2005).

2. We used linear summation to approximate integration of uPSPs in the cell body. This has been shown to be a good approximation in the neocortex (Araya et al., 2006; Leger et al., 2005; Tamas et al., 2002), cerebellum (Brunel et al., 2004), and hippocampus (Cash and Yuste, 1998, 1999).

3. We assumed that the threshold of each neuron remains fixed throughout learning. This assumption was motivated by the fact that coefficients of variation in the values of firing thresholds of cortical excitatory and inhibitory neurons are several-fold smaller than the
corresponding numbers for the uPSP amplitudes. For example, coefficients of variation for the numerous cortical projections summarized in Supporting Tables 1 and 2 of Chapeton et al. (Chapeton et al., 2012) have the following average values: 0.17 ± 0.02, (mean ± SE, \(n = 9\) systems) for firing thresholds and 0.94 ± 0.03 (\(n = 52\) systems) for connection weights.

4. We followed Dale’s principle (Dale, 1935) and assumed that the weights of excitatory/inhibitory inputs remain positive/negative throughout learning.

5. The activities of all neurons in the network \((j = 1, \ldots, N)\) at every time step, \(\mu\), were randomly drawn from neuron-class specific probability distributions, \(\text{Prob}(X_j^\mu)\), leading to successive network states that are (i) independent and (ii) random. With this approximation, the problem of learning by the network was decoupled and reduced to the problem of learning by \(N\) independent neurons. This approximation is supported by the following experimental observations. (i) For cortical neurons in vivo, serial correlation coefficients of inter-spike intervals are known to be small. For example, correlations of all lags greater than one are not significantly different from zero (Engel et al., 2008; Nawrot et al., 2007). Although small, but significant, lag one correlations (-0.2) are observed at high firing rates (Nawrot et al., 2007), these correlations vanish at < 2 Hz (Engel et al., 2008). (ii) Correlations between the activities of pairs of cells in vivo are known to be small. For example, low pairwise correlations have been reported for pyramidal cells in rat olfactory (~0.05) (Miura et al., 2012) and visual (~0.033) (Greenberg et al., 2008) cortices. Weak pairwise correlations have also been found in the sensorimotor cortex of behaving monkeys and humans (Truccolo et al., 2010). In addition, extracellular recordings from layer 2/3 of somatosensory cortex have shown that correlation coefficients between regular spiking cells are small during periods of spontaneous and evoked activity (0.04 and 0.02) (Middleton et al., 2012). Similar results have been obtained for the
correlations between regular spiking and fast spiking cells (0.11 and 0.01) (Middleton et al., 2012).

6. The $l_0$ and $l_1$ norm constraints were motivated by the following experimental evidence. (i) The density of spines on excitatory neuron dendrites remains constant over days to weeks in many areas of the adult cortex (Brown et al., 2009; Fuhrmann et al., 2007; Hofer et al., 2009; Holtmaat et al., 2006; Holtmaat et al., 2005; Kim and Nabekura, 2011). Likewise, the number of inhibitory synapses onto excitatory dendrites (Chen et al., 2012) and the number of spines on some inhibitory cell dendrites (Keck et al., 2011) remain nearly constant over days. Together, these studies suggest that homeostatic mechanisms may regulate the number of synapses received by excitatory and inhibitory neurons ($l_0$ norm constraint). (ii) It has been reported that the total size of spines remains constant over several days as measured by the normalized spine brightness (Holtmaat et al., 2006; Kim and Nabekura, 2011). Because the normalized spine brightness is correlated with spine volume (Holtmaat et al., 2005) and the latter is correlated with synaptic weight (Arellano et al., 2007; Harvey and Svoboda, 2007; Matsuzaki et al., 2004; Zito et al., 2009), the overall weight of the presynaptic inputs of a pyramidal cell may be conserved. Another study (Bourne and Harris, 2011) has reported that by two hours after LTP induction dendrites of CA1 pyramidal neurons in the hippocampus lose some of their small dendritic spines. However, this loss is balanced by an enlargement of the surface area of other excitatory synapses in such a way that the summed surface area of excitatory synapses remained constant across time and conditions. A similar trend was observed for the inhibitory synapses (Bourne and Harris, 2011). These observations imply that dendrites may use local protein synthesis to maintain the overall weight of excitatory and inhibitory inputs ($l_1$ norm constraint).
7. We assumed that associative memories can be recalled robustly in the presence of small noise in synaptic transmission, e.g. failures in generation or propagation of presynaptic action potentials, spontaneous neural activity, synaptic failure, and fluctuations in synaptic weight. In order to incorporate this feature into the model we assumed that an association was robustly learned by a neuron if it could be correctly recalled even in the presence of fluctuations in postsynaptic potential of size $\kappa$.

**Theoretical solution of the model**

The theoretical solution of the model, Eq. (4.6), is governed by four variables ($u_+, u_-, z,$ and $x$), which are implicitly defined by the following system of equations:

$$
\begin{align*}
    f_{out} I_1 (-u_-,0) &= (1-f_{out}) I_1 (-u_+,0) \\
    \frac{1}{N} \sum_{i=1}^{N} \left( \frac{1}{\sqrt{f_i(1-f_i)}} \right) I_1 \left[ g_i z \sqrt{\frac{f_i}{1-f_i}} + \left( \frac{\sqrt{N} \kappa x - h z}{N w_i \sqrt{f_i(1-f_i)}} \right) \delta_{i,1}, \sqrt{2 \left( \frac{\sqrt{N} \kappa x - h z}{hw_0} \right) Q} \delta_{i,0} \right] = \left( \frac{N}{h} \right)^i w_i Q^i \\
    \frac{1}{N} \sum_{i=1}^{N} \frac{f_i g_i}{\sqrt{f_i(1-f_i)}} I_1 \left[ g_i z \sqrt{\frac{f_i}{1-f_i}} + \left( \frac{\sqrt{N} \kappa x - h z}{N w_i \sqrt{f_i(1-f_i)}} \right) \delta_{i,1}, \sqrt{2 \left( \frac{\sqrt{N} \kappa x - h z}{hw_0} \right) Q} \delta_{i,0} \right] = Q \\
    \frac{1}{N} \sum_{i=1}^{N} I_2 \left[ g_i z \sqrt{\frac{f_i}{1-f_i}} + \left( \frac{\sqrt{N} \kappa x - h z}{N w_i \sqrt{f_i(1-f_i)}} \right) \delta_{i,1}, \sqrt{2 \left( \frac{\sqrt{N} \kappa x - h z}{hw_0} \right) Q} \delta_{i,0} \right] = \frac{2Q^2 N \kappa^2}{(u_+ + u_-)^2 h^2} \\
    Q &= 2h \frac{(u_+ + u_-) f_{out} I_0 (-u_-,0) + (1-f_{out}) I_0 (-u_+,0)}{\sqrt{N} \kappa} \\
    & \times \frac{f_{out} I_1 (-u_-,0) + (1-f_{out}) I_1 (-u_+,0)}{x} \\
    u_+ + u_- &\geq 0; \quad x \geq 0; \quad \left( \frac{\sqrt{N} \kappa x - h z}{hw_0} \right) \delta_{i,0} \geq 0
\end{align*}
$$

(4.7)

Detailed derivation of these equations, together with the definitions of the special functions $I_{0,1,2}$, can be found in Appendix C.
The critical capacity of a neuron, the probabilities of its non-zero weight connections for different input classes (denoted with $i$), $P_i^{\text{con}}$, and the probability density functions for its non-zero input weights, $p_i(J)$, can be expressed in terms of these four variables:

\[
\alpha_c = 2x^2 \left( \frac{f_{\text{out}} I_2(-u,0) + (1 - f_{\text{out}}) I_2(-u,0)}{(f_{\text{out}} I_1(-u,0) + (1 - f_{\text{out}}) I_1(-u,0))^2} \right)
\]

\[
P_i^{\text{con}} = I_0 \left( g_i z \sqrt{\frac{f_i}{1 - f_i}} + \frac{\sqrt{N_{\text{KX}} - h z}}{N w_i \sqrt{f_i(1 - f_i)}} \delta_{i,1} + \sqrt{\frac{2(\sqrt{N_{\text{KX}} - h z})Q}{h w_0}} \delta_{i,0,0} \right)
\]

\[
p_i(J) = \frac{\sqrt{f_i(1 - f_i)}}{2\sqrt{\pi} I_0 \left( g_i z \sqrt{\frac{f_i}{1 - f_i}} + \frac{\sqrt{N_{\text{KX}} - h z}}{N w_i \sqrt{f_i(1 - f_i)}} \delta_{i,1} + \sqrt{\frac{2(\sqrt{N_{\text{KX}} - h z})Q}{h w_0}} \delta_{i,0,0} \right)} \times \frac{Q - \left( \frac{2(\sqrt{N_{\text{KX}} - h z}) h \delta_{i,0}}{f_i(1 - f_i) w_0 N^2 J^2} \right)^\frac{1}{2} \left( h \frac{\delta_{i,1}}{w_0} \right)^\frac{1}{2} \left( h \frac{\delta_{i,0}}{w_0 J} \right)^\frac{1}{2}}{J^{1/2}} \right) \left( \frac{2(\sqrt{N_{\text{KX}} - h z})Q}{h w_0} \right)^\frac{1}{2} \delta_{i,0,0} \right) \right)
\]

(4.8)

Plus-sign in the subscript of the last equation denotes the positive part function. Corresponding results for the unconstrained case are included in Appendix C. Eqs. (4.7, 4.8) were solved with custom MATLAB code to produce the results shown in Figures 4.2, 4.3 and 4.4.

**Model parameters**

Results of the model, Eq. (4.8), depend on the following dimensionless parameters: fraction of potential inputs of each class, $N_i/N$, firing probabilities of these input classes, $f_i$, robustness of the postsynaptic neuron, $\sqrt{N_{\text{K}}}/h$, and the values of norm constraints, $w_0$ and $N w_1/h$. In the results discussed in this chapter, we only consider two classes of inputs, inhibitory and excitatory, and thus, the number of independent parameters in the unconstrained model reduces to four ($N_{\text{inh}}/N$, $w_0$, $N w_1/h$).
1. The fraction of potential inhibitory inputs received by a neuron in the network, $N_{inh}/N$, can be approximated by the average fraction of inhibitory neurons in the cortical column. The latter is known to be in the 0.11-0.20 range (Braitenberg and Schüz, 1998; Lefort et al., 2009; Meyer et al., 2011; Sahara et al., 2012). Thus, we used $N_{inh}/N = 0.15$ in Figures 4.4 and 4.5. Firing probabilities can be estimated based on the expression $f = r \times \tau$. Numerical values of firing rates, $r$, and integration windows, $\tau$, for excitatory and inhibitory neurons are given in point 1 of Model Assumptions and Approximations section. Based on these numbers we estimated that $f_{exc} \approx 0.1$, while $f_{inh}$ is expected to be larger due to generally higher firing rates of inhibitory neuron classes. However, because the exact values of firing probabilities are not known, we show the results for the unbiased case, $f_{inh} = f_{exc} = 0.5$ in Figure 4.2, and in the results shown we decided to adopt $f_{inh} = f_{exc} = 0.15$ (Figures 4.3, 4.4 and 4.5). We did not find a clear way to determine the value of robustness parameter, $\sqrt{N \kappa / h}$, from experimental data. Hence, we first show that the results of the model depend on the value of this parameter in a predictable way (Figures 4.3, 4.4), and then set $\sqrt{N \kappa / h} = 3$ (Figure 4.5).

2. The biologically plausible ranges for the dimensionless constraints, $w_0$ and $Nw_1/h$, were approximated based on their definitions. For two classes of presynaptic inputs these definitions yield:

\[
w_0 = \frac{N_{inh}}{N} P_{inh}^{con} \left(1 - \frac{N_{inh}}{N}\right) P_{exc}^{con} w_1 \left(1 - \frac{N_{inh}}{N}\right) P_{exc}^{con} J_{inh} + \left(1 - \frac{N_{inh}}{N}\right) P_{exc}^{con} J_{exc}\]

\[
\frac{Nw_1}{h} = \frac{N}{h} \left(\frac{N_{inh}}{N} P_{inh}^{con} \langle J_{inh} \rangle + \left(1 - \frac{N_{inh}}{N}\right) P_{exc}^{con} \langle J_{exc} \rangle\right)
\]
3. Here $P_{\text{inh,exc}}^{\text{con}}$ and $\langle J_{\text{inh,exc}} \rangle$ are the connection probabilities and the average uPSP amplitudes of inhibitory and excitatory inputs. To estimate the values of the constraints we combined the dataset compiled in Chapeton et al. (Chapeton et al., 2012) with a recent study of inhibitory connectivity (Packer and Yuste, 2011) and then restricted the analysis to neocortical systems. The 95% confidence intervals were then obtained using bootstrap sampling with replacement ($n = 10,000$ samples). Parameters $h$, $P_{\text{inh,exc}}^{\text{con}}$, and $\langle J_{\text{inh,exc}} \rangle$ were sampled with weights proportional to the numbers of experimental counts, whereas $N$ and $N_{\text{inh}}/N$ were sampled uniformly from 5,000-10,000 (Lefort et al., 2009; Meyer et al., 2010b) and 0.11-0.20 (Braitenberg and Schüz, 1998; Lefort et al., 2009; Meyer et al., 2011; Sahara et al., 2012) intervals. This procedure resulted in 95% confidence intervals of $[0.1 - 0.4]$ for $w_0$ and $[20 - 190]$ for $Nw_1/h$. In Figure 4.4 we show how results of the model depend on the values of these constraints, while in Figures 4.3 and 4.6 we opted to use the average values obtained from the bootstrap sampling, $w_0 = 0.25$ and $Nw_1/h = 70$.

**Numerical solutions of the model**

Since the analytical calculations used to produce the results of this study are very involved we used numerical simulations as an additional validation step. Details of these numerical algorithms can be found in Appendix C. Standard convex optimization methods were used to produce numerical solutions for the unconstrained and $l_1$ norm constrained problems. Since the $l_0$ norm constrained problem is non-convex, numerical solutions in this case were performed with a modified perceptron learning rule. The critical capacity (Figure 4.2A-C) and the distributions of connection
weights (Figure 4.2D-F) resulting from these simulations are in good agreement with the theoretical calculations.

**Figure 4.2**: Validation of theoretical results with numerical simulations. A-C. Critical capacity as a function of the fraction of potential inhibitory inputs, \(N_{inh}/N\). Theoretical results (solid lines) are generally consistent with the results of numerical simulations performed for \(N = 500\) potential inputs (solid dots). Results for various values of the robustness parameter \(\kappa\) are shown with different colors. Due to the difficulties associated with solving non-convex optimization problems, comprehensive numerical simulations in the \(l_0\) case (B) were performed for two values of \(\kappa\) only. Numerous spot-checks were performed to validate the general trends of other theoretical curves in the \(l_0\) case (results not shown). D-F. Distributions of excitatory and inhibitory connection weights for \(N_{inh}/N = 0.15\), \(\sqrt{N\kappa}/h = 0.5\), and \(f_{inh,exc} = 0.5\). Values of the constraints are \(w_0 = 0.2\) in E and \(Nw_1/h = 3\) in F. The theoretical results are in green and the results of numerical simulations performed for \(N = 500\) potential inputs are in blue. Small deviations of numerical distributions from theoretical are attributed to relatively small numbers of excitatory and inhibitory inputs used in simulations. Corresponding theoretical and numerical fractions of non-zero weight connections, \(P^{con}\), are shown in green and blue respectively.
Numerical simulations were also used to illustrate plausibility of the steady-state learning hypothesis, which relies on the assumption that a network can reach the state of maximum memory storage capacity. To this end, we used perceptron-type learning rules and biologically plausible model parameters to reproduce theoretical results of all three cases (Figure 4.5).

**Fitting distributions of connection weights**

Theoretical probability density functions of Eq. (4.8) were used to fit simulated and experimental distributions of connection weights. To this end, these equations were rewritten by using the experimentally determined values of $P_i^{\text{con}}$ and introducing two parameters: $\sigma_i$, which describes the width of the distribution and $G_i$, which is the magnitude of the minimum non-zero connection weight present in the $l_0$ model. The resulting probability density functions are governed by two parameters ($\sigma_i$ and $G_i$) in the $l_0$ case and one parameter ($\sigma_i$) in the $l_1$ case:

$$\sigma_i = \frac{\sqrt{2h}}{\sqrt{f_i(1-f_i)QN}}; \quad G_i = \frac{2(\sqrt{N\kappa x} - hz)}{f_i(1-f_i)w_0Q} \frac{h}{N^2}$$

$$p_i(J) = \frac{1}{\sqrt{2\pi}P_i^{\text{con}}} \frac{h}{N\sigma_i} \left(1 - \frac{G_i^2}{J^2} \delta_{i,0}\right) e^{-\left[\frac{g_iJ}{\sqrt{2\sigma_i}} \text{erf}(1-2P_i^{\text{con}}) + \frac{\sqrt{2\sigma_i}}{2g_iJ} \delta_{i,0}\right]^2}$$

(4.10)

Fitting the simulated distributions of inhibitory/excitatory connection weights shown in Figure 4.5 was done with one parameter ($\sigma_{\text{inh}}/\sigma_{\text{exc}}$) in the unconstrained and $l_1$ constrained models, and two parameters ($\sigma_{\text{inh}}/\sigma_{\text{exc}}$ and $G_{\text{inh}}/G_{\text{exc}}$) in the $l_0$ case. Fitting was done in MATLAB by using nonlinear least squares fit. We note that the functional form of the distribution of connection weights in the unconstrained model (Chapeton et al., 2012), written in the notation of Eq. (4.10), is identical to that of the $l_1$ case. For this reason, the unconstrained and $l_1$ models produce identical fits.
A similar fit of experimental distributions of uEPSP amplitudes is shown in Figure 4.6. Fitting with the $l_0$ model produced the following best fit parameters: $\sigma = 1.06 \ [0.88 - 1.23]$ mV (mean [95% confidence interval]), $G = 0.055 \ [0.045 - 0.064]$ mV in Figure 4.6A and $\sigma = 0.79 \ [0.71 - 0.88]$ mV, $G = 0.051 \ [0.041 - 0.060]$ mV in Figure 4.6B. For the $l_1$ model we discarded weak, unreliable connections (gray regions in Figure 4.6) and thus introduced a normalization factor $A$ as an additional fitting parameter. Hence, fitting in this case was also performed with two parameters. The resulting best fit values of $\sigma$ were: $1.02 \ [0.92 - 1.13]$ mV in Figure 4.6A and $0.81 \ [0.70 - 0.92]$ mV in Figure 4.6B.

**Effects of homeostatic constraints on network capacity and connectivity**

The general solution of the model is described in Appendix C. Since this solution is very involved, theoretical results were validated with numerical simulations (see Figure 4.2). Here we illustrate the theoretical results by considering a single cell receiving inputs from two classes of presynaptic neurons, one inhibitory and one excitatory. The critical associative memory storage capacity of this cell, $\alpha_c$, was calculated by solving the system of Eqs. (4.7, 4.8). Figure 4.3A-C shows the dependence of $\alpha_c$ in the unconstrained and $l_0, 1$ norm constrained models on the fraction of potential inhibitory inputs, $N_{inh}/N$, and the robustness parameter, $\kappa$. Though the results for the three models are distinctly different, there are notable common trends. First, in all three models, the critical capacity is a decreasing function of $\kappa$, indicating the trade-off between the maximum number of associations a neuron can learn and the robustness of the learned associations. Second, in the case of robust memory storage ($\kappa > 0$), adding a small fraction of inhibitory neurons increases $\alpha_c$. This, however, comes at the expense of the total number of non-zero weight connections, $N_{w0}$, and/or the overall connection weight, $N_{w1}$ (Figure 4.3D-F).
Figure 4.3: A-C. Critical capacity as a function of the fraction of potential inhibitory inputs for $f_{inh} = f_{exc} = 0.15$ and various values of the robustness parameter $\kappa$. A. The unconstrained model. At certain values of $N_{inh}/N$, the curves merge with the asymptotic solution (black curve) corresponding to the limit of large robustness, $\sqrt{N_{inh}/N}/h \gg 1$. For smaller fractions of inhibitory inputs, the critical capacity is a decreasing function of $\kappa$ and an increasing function of $N_{inh}/N$. B. Qualitatively similar results were obtained in the $l_0$ constrained model. C. In the $l_1$ constrained model the critical capacity curves are slightly skewed to the left and have a maximum at $N_{inh}/N < 0.5$ for all values of $\kappa$. Values of the constraints are $w_0 = 0.25$ in B and $Nw_1/h = 70$ in C. D-F. Overall connection probability (dashed lines) and overall connection weight (solid lines) as a function of $N_{inh}/N$. Note the different y-axis scales for $w_0$ (linear, left) and $Nw_1/h$ (logarithmic, right). In the constrained models, the overall connection probability (E) or the overall connection weight (F) is fixed for all values of $N_{inh}/N$ (horizontal lines).

Next, we evaluated the effects of the $l_{0,1}$ norm constraints on the critical capacity and connection probabilities for various input classes. Figure 4.4 shows the results for two classes of inputs, excitatory and inhibitory, with $N_{inh}/N = 0.15$. This numerical value, as well as the values of other parameters of the theory, is based on published experimental data. As expected, the critical capacity of the constrained models is maximal when $w_0$ (Figure 4.4A) and $w_1$ (Figure 4.4B) match the corresponding values of these parameters in the unconstrained case (diamonds in the figure).
This is because at these exact values of $w_0$ and $w_1$ the norm constraints are effectively removed, and the solutions of the constrained models reduce to that of the unconstrained case, which naturally has the maximum critical capacity.

**Figure 4.4:** Effects of $l_0$ and $l_1$ constraints on critical capacity and connection probability. **A, B.** Critical capacity of the $l_0$ and $l_1$ constrained models plotted as a function of the constraints, $w_0$ and $Nw_1/h$. Diamonds denote the corresponding value of the critical capacity in the unconstrained model. The critical capacity is at its maximum when $w_0$ and $w_1$ match the corresponding values calculated for the unconstrained model. **C, D.** Excitatory (solid lines) and inhibitory (dashed lines) connection probabilities in the $l_0$ and $l_1$ constrained models. Gray histograms in (C) represent excitatory (solid outline) and inhibitory (dashed outline) connection probability data from a large set of experimental studies (see text for details). Note that the histogram counts are shown at the top of panel C.
Increasing $w_0$ beyond this point in the $l_0$ model has no effect on critical capacity (Figure 4.4A) because it is always possible to start with the connectivity of the unconstrained network, and then add a small number of infinitesimally weak connections which will have no effect on the learned associations, but will increase the overall connection probability to the desired value. As the capacity of the $l_0$ model cannot be greater than the capacity of the unconstrained model, solutions constructed in this way have the maximum possible capacity, and thus are valid for large values of $w_0$ (to the right of the diamonds in Figure 4.4A). However, because multiple solutions of this type exist, excitatory and inhibitory connection probabilities cannot be defined uniquely.

Numerous experimental studies have shown that the probabilities of local excitatory and inhibitory connections onto the principal cortical neurons (pyramidal and spiny stellate cells in the cerebral cortex and Purkinje cells in the cerebellum) are distinctly different. In particular, excitatory connections are sparse, with connection probabilities well below 0.5, while the inhibitory connection probabilities are generally much higher. This difference can be seen in the histograms of Figure 4.4C which summarize connection probabilities compiled in (Chapeton et al., 2012) together with the data from a large study of parvalbumin positive to pyramidal cell connectivity (Packer and Yuste, 2011). Consistent with these observations, the probabilities of excitatory and inhibitory connections in the unconstrained model have been shown to be distinctly different (Chapeton et al., 2012), $P_{\text{exc}}^{\text{con}} < 0.5$ and $P_{\text{inh}}^{\text{con}} > 0.5$ (diamonds in Figure 4.4C, D). Therefore, we decided to examine if the constrained models considered in this study produce a similar trend. Figure 4.4C shows $P_{\text{exc}}^{\text{con}}$ and $P_{\text{inh}}^{\text{con}}$ in the $l_0$ norm constrained model plotted as functions of the overall connection probability, $w_0$. Both $P_{\text{exc}}^{\text{con}}$ and $P_{\text{inh}}^{\text{con}}$ increase with $w_0$, however, $P_{\text{exc}}^{\text{con}}$ always remains below 0.5, while $P_{\text{inh}}^{\text{con}}$ exceeds 0.5 beyond certain values of $w_0$. The range of $w_0$ values
estimated for excitatory cells in the neocortex is 0.1 – 0.4. In this range $P_{\text{exc}}^{\text{con}} < 0.5$, while $P_{\text{inh}}^{\text{con}}$ is higher than $P_{\text{exc}}^{\text{con}}$ in the case of robust memory storage ($\kappa > 1$). Connection probabilities in the $l_1$ norm constrained model depend on the value of $Nw_1/h$ (Figure 4.4D). This parameter, estimated from the experimental data, is in the 20 – 190 range. In this range $P_{\text{exc}}^{\text{con}} < 0.5$ and $P_{\text{inh}}^{\text{con}} > 0.5$ for all values of robustness. Thus, for biologically realistic values of $w_0$ and $Nw_1/h$ the connection probabilities produced by the homeostatically constrained models are consistent with the experimentally observed difference in probabilities of excitatory and inhibitory connections onto principal cells.

**Distribution of connection weights**

In this section we compare the probability densities of input connection weights at critical capacity for the unconstrained and $l_0,1$ norm constrained models [see Appendix C and Eqs. (4.8, 4.10)]. In the unconstrained and $l_1$ norm constrained cases these probability densities consist of Gaussian functions, truncated at zero, and finite fractions of zero-weight connections (Figure 4.5 A,C). The distribution of connection weights in the $l_0$ norm constrained model also contains a finite fraction of zero-weight connections. However, the probability density function for non-zero connection weights is non-Gaussian (Figure 4.5B). Interestingly, this function has a gap for weak input connections, i.e. it does not contain non-zero connection weights below a certain threshold. We would like to point out that this feature of connection weights constrained by the $l_0$ norm was previously reported by Bouten et al. (Bouten et al., 1990), who considered associative learning by a neuron receiving a single class of sign-unconstrained inputs, i.e. the inputs were not constrained to be excitatory or inhibitory.
Figure 4.5: Comparison of theoretical distributions of connection weights to numerical simulations. A-C. Probability density functions obtained with perceptron-type learning rules for $N = 500$, $N_{\text{inh}}/N = 0.15$, $\sqrt{N\kappa}/h = 3$, and $f_{\text{inh, exc}} = 0.15$ are shown with blue bars. Values of the constraints are $w_0 = 0.25$ in B and $Nw_1/h = 70$ in C. Green lines show theoretical fits of these probability density functions with Eq. (4.10). Goodness of the fits is captured by the high adjusted $R^2$ coefficients. The theoretical distributions of excitatory and inhibitory connection weights in the unconstrained (A) and $l_1$ constrained (C) models consist of Gaussians truncated at $J = 0$ and finite fractions of zero-weight connections. The distribution in the $l_0$ constrained model (B) also contains a finite fraction of zero-weight connections, but is non-Gaussian. This distribution has gaps between zero and non-zero connection weights for excitatory and inhibitory inputs. Parameters $P_{\text{inh, exc}}^{\text{con}}$ give theoretical fractions of inhibitory and excitatory non-zero weight connections. D-F. Same probability density functions plotted on a log-log scale show deviations between theory and numerical simulations in the head and tail regions of the distributions.
Since the steady-state learning hypothesis relies on the assumption that the network can achieve the state of maximum associative memory storage capacity, we set out to show that this can be done with a biologically plausible learning rule. To this end, we attempted to reproduce the theoretical critical capacities and the shapes of theoretical connection weight distributions in the three models by using modified perceptron learning rules. The simulations were performed for biologically plausible values of model parameters and about 95% of theoretical, maximum memory storage capacity was reached in all three models: $\alpha_c = 0.95/0.99$ (numerical/theoretical) in the unconstrained model, 0.94/0.99 in the $l_0$ model, and 0.97/1.01 in the $l_1$ model.

The overall shape of numerical distributions generated at theoretical critical capacity (Figure 4.5A-C) was in good agreement with the theory, Eq. (4.10). However, small deviations between theory and numerical simulations were observed in the head and tail regions of these distributions. To examine these deviations in more detail we re-plotted the distributions of connection weights on a log-log scale (Figure 4.5D-F). In the unconstrained and $l_1$ norm constrained models the tails of the numerical distribution appear to be slightly heavier than theoretically predicted Gaussian tails (Figure 4.5D, F), while in the $l_0$ case (Figure 4.5E) there is a slight deviation in the regions of weak inhibitory and excitatory connections. It is likely that this deviation results from the fact that the numerical simulations were performed for a large, yet finite number of potential inputs, $N = 500$, while theoretical distributions were obtained in the limit of infinite $N$.

**Comparison of experimental and theoretical connection weight distributions**

The two homeostatically constrained models produce distinctly different distributions of connection weights. Below, we investigate how these distributions compare with the distributions of uPSP amplitudes reported in experimental studies. For this purpose, we selected two studies with very high counts of recorded uEPSPs, one performed in rat visual cortex ($n = 931$, layer 5)
(Song et al., 2005) and the other in mouse barrel cortex (n = 909, all layers) (Lefort et al., 2009). The two models (green and red lines in Figure 4.6A, B) were used to fit both experimental distributions (blue histograms in Figure 4.6A, B). Despite the fact that the goodness of these fits was high as measured by the adjusted $R^2$ coefficients, significant deviations were observed between the distributions ($P < 10^{-12}$ for both models in Figure 4.6A, B; Kolmogorov-Smirnov test), most noticeably in the head and tail regions. To focus on these differences, Figure 4.6C, D shows the distributions on a log-log scale.

Due to the fluctuations in electrophysiological recordings, very weak connections between neurons cannot be detected reliably. Such connections are often missed or ignored, leading to a systematic underestimate of weak connection counts. The gray regions in Figure 4.6 highlight the values of uEPSP amplitudes which fall below the reliable uEPSP detection threshold [0.1–0.25 mV in rodent neocortex (Berger et al., 2009; Feldmeyer et al., 1999; Markram et al., 1997a; Mason et al., 1991)]. Unfortunately, the difference between the distributions produced by the two models only becomes apparent inside the gray regions, and thus, it cannot be directly tested. In these regions of small uEPSP amplitudes the $l_0$ model provides seemingly better fits to the experimental distributions, but the statistical significance of these results could not be verified based on the available data.
Figure 4.6: Comparison of theoretical and experimental distributions of connection weights. A, B. Blue bars show distributions of uEPSP amplitudes in layer 5 of rat visual cortex (Song et al., 2005) and all layers of mouse barrel cortex (Lefort et al., 2009). These distributions were fitted with both the \( l_0 \) (green) and \( l_1 \) (red) constrained models (see text for details). Goodness of the fits is captured by the adjusted \( R^2 \) coefficients, which are close to 1. C, D. To examine how the two models fit the heads and the tails of the distributions of uEPSP amplitudes (blue line), panels A and B are re-plotted on a log-log scale. Blue numbers indicate the counts in the logarithmic bins. Data points, which fall below the uEPSP detection threshold and are thus unreliable, are highlighted in gray. The probability density in the \( l_0 \) model has a gap between zero and non-zero connection weights; there is no such gap in the \( l_1 \) model.
Both models of steady-state learning underestimate the counts of strong synaptic connections (uEPSP amplitudes > 1 mV). Though not very numerous (blue numbers in Figure 4.6C, D), these strong connections appear to be a characteristic feature of cortical connectivity. Sub-criticality of neural networks in the brain, non-linearity in the summation of presynaptic inputs (Brunel et al., 2004), as well as the effects of the finite size of local cortical networks (Chapeton et al., 2012) (see e.g. Figure 4.5D, right) have been proposed as possible explanations for the discrepancy between the Gaussian tails of the theoretical probability density functions and the much heavier distribution tails observed experimentally. At present, we are unable to differentiate between these explanations.

Discussion

In this study, we generalize the model of (Chapeton et al., 2012) by incorporating multiple classes of excitatory and inhibitory neurons and including two types of experimentally motivated homeostatic constraints. The constraints were designed to ensure that individual neurons maintain a fixed total number or a fixed overall weight of their non-zero inputs throughout learning. Both constrained models were solved analytically by using the replica theory. The results were validated with numerical simulations and compared to the available data on cortical connectivity.

Our theory produced two specific results regarding the connectivity among potentially connected cells in steady-state networks. First, we showed that functional excitatory connections onto principal cells should be realized with less than 50% probability, while the probabilities of inhibitory connections should be higher (Figure 4.4C, D). Because in cortical systems inhibitory cells account for only 11-20% of all neurons, functional connectivity in a steady-state network is expected to be sparse, i.e. it must contain a large fraction of zero-weight connections. This theoretical finding is in qualitative agreement with a dataset compiled from 38 published studies
(62 projections in total) in which connection probabilities have been measured in various cortical systems (histograms in Figure 4.4C). It is important to note that a zero-weight connection between neurons does not necessarily imply that the structural connection is absent, as the neurons may still be connected with synapses that are silent (synapses devoid of AMPA receptors (Malinow and Malenka, 2002)). This detail, however, did not factor into the comparison because neither the theory nor the electrophysiological recordings discriminate between the two alternatives. Furthermore, it has been shown that the fraction of silent synapses in adult cortex is low [see e.g. (Busetto et al., 2008)].

Second, we derived the shapes of the connection weight distributions in a steady-state [see Eq. (4.10) and Figure 4.5, 4.6]. Similar to the unconstrained model, distribution in the $l_1$ case consists of Gaussians truncated at zero and a finite fraction of zero-weight connections. The distribution in the $l_0$ model also contains a finite fraction of zero-weight connections, but the shape of the distribution for non-zero connection weights is no longer Gaussian. It is characterized by a gap between zero and non-zero connection weights. Hence, we predict that a network operating in a steady-state, subject to a constraint on the number of functional connections cannot contain arbitrarily small connection weights. Rather, there should be no functional connections weaker than a certain threshold (~0.05 mV for uEPSP amplitudes in the neocortex, Figure 4.6). It is not impossible to envision a biological mechanism by which weak connections can be silenced and/or completely eliminated (Oh et al., 2013). In practice, however, it may be difficult to test this hypothesis because the connection weight threshold is too small to be measured reliably with current experimental techniques (Berger et al., 2009; Feldmeyer et al., 1999; Markram et al., 1997a; Mason et al., 1991). Interestingly, the value of the connection weight threshold obtained in this study is in agreement with the smallest quantal sizes and miniature EPSP amplitudes recorded
in cortical pyramidal neurons [−0.1 mV (Hardingham et al., 2010)], responses believed to be produced by the release of neurotransmitter from a single presynaptic vesicle.

On the whole, the shapes of model distributions for non-zero connection weights are consistent with the experimental distributions of uEPSP amplitudes in rodent neocortex (Figure 4.6). However, significant discrepancies were observed in the head and tail regions of the distributions ($P < 10^{-12}$, Kolmogorov-Smirnov test). Due to the uncertainties in electrophysiological recordings of very weak connections (below 0.1–0.25 mV in the neocortex), there is still controversy regarding the shape of the uPSP amplitude distribution in this region. Does the distribution smoothly approach zero with decreasing uPSP amplitude as suggested by the log-normal fit performed in (Song et al., 2005) and predicted based on the model of multiplicative spine size dynamics (Loewenstein et al., 2011), does it increase as implied by the exponential distributions of spine head volumes (Mishchenko et al., 2010; Stepanyants and Escobar, 2011) and predicted by the unconstrained and $l_1$ norm constrained models, or does it have a gap near zero as predicted by the $l_0$ model? New, more sensitive experimental measurements are required to provide a definitive answer to these questions.

Both models of steady-state learning described in this study predict Gaussian decay of connection weight distributions in the region of strong connections. Contrary to this, distributions of uEPSP amplitudes in Figure 4.6 exhibit much heavier tails in the region beyond 1 mV. Several explanations have been previously proposed in order to account for this feature theoretically. Neurons may be operating below their critical capacity, or individual inputs to a neuron may be non-linearly transformed in the dendrites before they are summed in the cell body (Brunel et al., 2004). Heavy distribution tails have also been attributed to the fact that the number of potential presynaptic connections received by cortical neurons, though large [$N \sim 5,000$-10,000] (Lefort et
al., 2009; Meyer et al., 2010b), is finite, while the theoretical results of this study were obtained in the $N \rightarrow \infty$ limit. In fact, numerical simulations performed for $N = 500$ potential inputs can lead to heavier tails of connection weight distributions, supporting this interpretation [see e.g. Figure 6D in (Chapeton et al., 2012)].

It was previously shown [see Figure 4.6C in (Chapeton et al., 2012)] that the unconstrained model provides a good fit to the IPSP distribution reported in Holmgren et al. (Holmgren et al., 2003). Unfortunately, due to generally low counts in published PSP data for inhibitory neurons we had to refrain from examining such connections in this study. With future advances in optical methods such as glutamate uncaging (Packer and Yuste, 2011) and optogenetic tagging of genetically defined interneurons (Kvitsiani et al., 2013), it should be possible to study connections between multiple inhibitory and excitatory classes. Here, significant deviations in connection probabilities and shapes of PSP distributions from what is predicted by the theory could reveal connection types that are not directly involved in associative learning.

Our results show how neural networks may benefit from the presence of a small fraction of inhibitory neurons and connections. Figure 4.2 illustrates that a small fraction of inhibitory connections increases the capacity of the neurons for robust associative memory storage. This increase, however, comes at the expense of the overall connection weight in the $l_0$ model or total number of functional connections in the $l_1$ model, quantities that are likely to be directly related to the metabolic cost of the brain. It would be interesting to find out how different cortical areas balance network performance, measured in terms of information flow or memory storage capacity, with the metabolic cost associated with neuron firing, and number and weight of functional synaptic connections.
In this study, it was assumed that memory recall is a dynamic event in which activity of the network steps through a chain of predefined states. It is well known that in binary neural networks, such as the ones considered here, a chain of network states will inevitably terminate at an attractor in the form of a fixed point or a limit cycle. Hence, memory recall is bound to lead to a frozen network state, or states of cycling activity. With no external input, network that is robust to small fluctuations could remain in the attractor state over a prolonged period of time, which is unrealistic. Thus we think that external input, delivered via inter-areal, inter-hemispheric, and subcortical projections, must be responsible for reinitializing the network activity and initiating the recall of new associative memories.

The theoretical model presented in this study describes the effects of associative learning on synaptic connectivity in a steady-state of maximum memory storage capacity. We would like to emphasize that network connectivity in such a state need not be static. As the network is continually learning new associative memories (while forgetting some of the old ones) individual synaptic connections may potentiate or depress, new synaptic connections may be created, and the existing connections may be eliminated. Yet, the average features of steady-state connectivity must remain constant. What is more, these features are independent of the path taken by the network to reach the steady-state, making the problem theoretically tractable. Clearly, not all changes in synaptic connectivity can be described within the steady-state framework presented here. Changes which occur during development (van Ooyen, 2011), follow injury or lesion (Butz and van Ooyen, 2013), or accompany learning of new skills may perturb the network from the steady-state for prolonged periods of time (Ruediger et al., 2011). It is more difficult to model these non-equilibrium processes, as they require detailed knowledge of animal’s experience and the learning rules involved.
Conclusion

In this thesis, I first described the design of computational methods to perform analyses of neuronal connectivity based on light microscopy imaging experiments. Next, I applied these methods to examine structural dynamics of neuronal connectivity based on time-lapse imaging experiments performed in vivo in the context of behaviorally measured perceptual learning. Finally, I discussed theoretical approaches to show that networks constructed to optimize memory storage under biologically plausible homeostatic constraints can explain various experimentally observed features of neuronal connectivity.

As sample preparation and image acquisition techniques achieve greater automation, the time it takes to generate imaging datasets to map neuronal connectivity at the whole brain level continues to decrease. Protocols and equipment to obtain fluorescence images of neurites vary from one lab to another, which hinders efforts to create standardized, annotated datasets that can be used to design automated tracing algorithms. This problem is now being addressed by large-scale data curation efforts (Peng et al., 2015a; Peng et al., 2015b; Roskams and Popovic, 2016). In the absence of such datasets, one could incorporate knowledge about neuron morphology explicitly into algorithms that parse raw imaging data of neurites into traces. We implemented such an algorithm with the focus on inferring correct trace topology, an issue that is central to the success of large-scale brain mapping projects (Economo et al., 2016).

To understand the role of synaptic plasticity in learning and memory, imaging experiments require reliable detection of subtle structural changes in neuronal circuits. Manually scoring synaptic structures is not only time consuming, but can also lead to user dependent bias and variability in measurements. We developed automated methodology to detect and measure axonal en passant
boutons in time-lapse imaging experiments. We validated this methodology with correlative light and electron microscopy, and evaluated measurement uncertainty that can be expected due to differences in imaging conditions, an issue that is unavoidable in long-term imaging experiments.

In the experiment described in chapter three, the population of neurons imaged in a time-lapse manner was also the one that provided mice with information necessary to perform the perceptual task. We generated a large dataset of bouton structural changes in the animals being trained on the task. Despite the facts that animals experienced a novel environment and underwent training on a task that involved optogenetic manipulation of labeled neurons, the overall number and weight of boutons along axons of these neurons remained stable over the duration of the experiment. We found that both, bouton weights, and bouton weight changes are correlated over tens of micrometers along the axons. We quantified bouton addition and elimination, and found circuits to be more plastic in animals that learned to perform the task.

In the last chapter, we approached the problem of learning and memory from a theoretical perspective. We examined an exactly solvable model of associative learning that is based on the hypothesis that local cortical circuits operate in a steady state. In this steady state, individual neurons in the network maximize their associative memory storage capacity. We generalized the approach of (Chapeton et al., 2012) by incorporating multiple classes of excitatory and inhibitory neurons and including two types of biologically motivated homeostatic constraints. The analytical results were validated with numerical simulations, and predictions of the model were compared with a dataset of cortical connectivity compiled from many experiments.

Advances in experimental techniques have continued to enable observation and manipulation of increasingly large ensembles of neurons. It has become critical to devise computational methods
to extract quantitative information from enormous volumes of raw data generated by such experiments. Translating quantitative data into conceptual understanding requires principled exploration, analysis and modeling of the data, guided by theoretical frameworks. The blurring of boundaries between traditional disciplines in this process is fascinating, and one cannot help but be excited about what piece of the puzzle is discovered next in the quest to understand general principles that govern brain function.
Appendices

Appendix A

Optimization of manual traces

Figure A.1: Optimization is required for trace-based bouton detection and measurement. A. 5 users traced the same set of 16 axon segments using manual tracing module of NCTracer software. Manual traces (colored lines) are shown superimposed on zx (top), yz (left), and xy (center) maximum intensity projection images of an axon segment. To improve visibility, image intensity beyond five voxels from the axon centerline was set to zero. While on the xy projection, manual traces closely match the underlying axon intensity, variability between traces becomes apparent by viewing the zx and yz projections. B. User traces after optimization are virtually indistinguishable on all projections. C. Axon intensity profiles can be obtained by sliding various filters along manual traces. Here, we show LoGxy intensity profiles for the 5 manual user traces. Note that slight tracing inaccuracies may lead to large variability in axon intensity profiles, e.g. absent peak in the User 4 trace profile (dashed line). D. Inter-user variability in axon intensity profiles is virtually eliminated after trace optimization.
Figure A.1A shows three maximum intensity projection views of an axon segment, which was traced independently by five users. Inter-user trace variability, which is usually more pronounced along the z-dimension (optical axis), can hinder bouton detection and measurement. Therefore, it is essential to optimize the layout of manual traces, ensuring that they accurately follow axon centerlines in the underlying image.

Trace optimization was performed in custom software written in MATLAB (MathWorks Inc., Natick MA). Detailed description of the optimization algorithm can be found in (Chothani et al., 2011; Gala et al., 2014). In this study, a simplified version of the optimization algorithm was used in which positions of trace nodes were optimized, but calibers of trace nodes were kept constant at 3 voxels (roughly equal to axon diameters in images). The average density of trace nodes was set to 1 per voxel, and the parameter controlling trace stiffness was set to 0.001. Results of this optimization procedure (Figure A.1B) show marked improvement over manual traces. The optimized traces are smoother, closer to one another in z, and follow the underlying axon intensity in the image more accurately. Following optimization, all traces were subdivided to a higher density of nodes (4 per voxel).

Figure A.1C shows LoGxy profiles for the five manual traces from Figure A.1A. Small differences in the layout of manual traces can lead to significant variability in intensity profiles, which is undesirable. Trace optimization reduces this variability to an extent undetectable by visual inspection, Figure A.1D.

Effects of trace variability on automated bouton detection

To understand the effects of small trace inaccuracies on the results of the method, putative boutons were detected automatically, both with and without trace optimization, on the same set of 16 axon
segment traced manually by 5 users. In the absence of trace optimization, a total of 578 candidate bouton sites were identified and matched across all 5 user traces (Figure A.2A).

**Figure A.2:** Trace optimization reduces variability in bouton detection and weight measurement. A. Putative boutons were automatically detected based on manual traces of 5 users. Differences in user traces lead to variability in putative bouton detection. The fraction of inter-user conflicts is particularly high for low weight boutons. B. Trace optimization reduces the fraction of inter-user conflicts. C. Trace optimization also reduces inter-user bias and variability in bouton weight measurements. D. RMS weight difference, calculated for putative boutons detected on manual (red points) and optimized traces (blue points), shows that optimization reduces variability in the entire range of bouton weights.
A candidate bouton site here is defined as a position on an axon where a putative bouton was detected based on at least one user trace. For the majority of candidate bouton sites, 348 (60%), a putative bouton was present in all 5 user traces, and thus, there was a full consensus in bouton detection. However, 141 (24%) candidate bouton sites had one conflict and 89 (16%) had two. A conflict is defined as a dissent from the majority, and given 5 traces, the maximum number of conflicts is two. In contrast, after trace optimization, 474 candidate bouton sites were detected, of which 409 (86%) had full consensus, 39 (8%) had one conflict, and 26 (6%) had two (Figure A.2B). With trace optimization it was possible to obtain complete agreement in all putative boutons of weights greater than 2.5. This is a marked improvement over bouton detection with no trace optimization where 17 putative boutons of weights greater than 2.5 had one or more conflicts.

Tracing inaccuracies also affect the weights of detected putative boutons. Figure A.2C, shows that inaccurate manual traces can contribute to bias (slope difference from 1) and variance (deviation from the best fit line) inbouton weight, and that trace optimization significantly reduces such errors. To quantify the uncertainty in bouton weight we calculated root mean square (RMS) weight difference for putative boutons detected with no conflicts (Figure A.2D). As was expected, trace optimization dramatically reduces this uncertainty in the entire range of bouton weights.

We would like to point out that some amount of uncertainty in bouton detection (Figure A.2B) and measurement (Figure A.2D) remains even after trace optimization. However, as shown in Figure 2.5A-C this uncertainty of the method is much smaller than variability originating from the experimental sources, and therefore, it is not a limiting factor in structural plasticity measurements.
Appendix B

User manual for BoutonAnalyzer

BoutonAnalyzer is software for detection and tracking of structural changes in en passant boutons in time-lapse light-microscopy stacks of images. Technical details and validation of the algorithms implemented in this software can be found in chapter two.

System requirements and installation

BoutonAnalyzer is designed to run on Windows and Mac installations of MATLAB (versions 2015a to 2017a). To install BoutonAnalyzer, download BoutonAnalyzer.zip file from http://www.northeastern.edu/neurogeometry/resources/bouton-analyzer/ and extract its contents. Start MATLAB, navigate to the \src folder, and run BoutonAnalyzer in the MATLAB command window. Do not change MATLAB path while running the software.

Directory structure, file formats, and naming conventions

BoutonAnalyzer requires paths for loading image files and traces, and for saving axon intensity profiles and results. The Optimize Trace & Generate Profile GUI (Figure B.4) uses an image stack and an axon trace as inputs and generates an output file containing optimized trace and axon intensity profile. The Detect & Track Boutons GUI (Figure B.5) loads these data for an axon in multiple imaging sessions and allows the user to detect, edit, and match boutons across sessions. A strict file naming convention must be followed to allow BoutonAnalyzer to search and organize data. In the following, we describe the various allowed file types and naming conventions for images, traces, and profiles.

Images

Image stacks can be loaded in .mat, multi-layer .tif (or .tiff), and single-layer .tif (or .tiff) file formats (Figure B.1). The .mat file is expected to contain grayscale intensity values of the image
stack voxels in a 3d array named IM. Image file names must include animal identity (e.g. DL1), order within the time-lapse sequence (e.g. B), image stack name (e.g. S2), and channel (e.g. Gr). Each identifier must begin with an upper case letter and can be followed by an arbitrary number of alphanumeric characters or underscores. Two examples of image names consistent with this format are DL1BS2Gr.mat and DL_01BS2G.mat.

![Image format](image.png)

**Figure B.1:** Compatible image stack file formats and naming conventions

**Traces**

Traces of axons are expected to be in the .swc file format with coordinates given in voxel units. NCTracer or other software can be used to create traces and export them as .swc files. Each .swc file should correspond to a unique axon segment without branch points. Axons traced in a single image stack must have names in the range A001.swc to A999.swc. Name of the parent folder should correspond to the name of the image stack without the channel identifier. An axon traced in multiple imaging sessions must have the same name in all folders corresponding to these imaging sessions. For example, axon traces A002.swc in folders DL1AS2 and DL1BS2 correspond to the same axon imaged in sessions A and B (Figure B.2).
Figure B.2: Naming convention for axon traces

Profiles

Axon intensity profiles are .mat files created by BoutonAnalyzer. The naming convention and directory structure of profile files are the same as those for traces.

Results

Once boutons have been detected and matched for a given axon across multiple imaging sessions, files containing bouton weights ($w$) and bouton probabilities ($p$) are created in both .txt and .mat formats. The .txt file can be opened with a text editor, and the values can be imported in Excel for the analysis of structural plasticity. The .mat file contains additional information related to flags assigned during matching ($flag$), 3D position of boutons in the stack ($rx$, $ry$, $rz$), bouton positions along the axon ($d$), and other variables that can be used for a more detailed analysis.

Bouton Analysis Workflow

BoutonAnalyzer can be used to perform the following operations:

1. Trace optimization
2. Generation of axon intensity profiles
3. Registration of traces across multiple imaging sessions
4. Annotation of traces
5. Detection, editing, and matching boutons in multiple imaging sessions

**Figure B.3:** BoutonAnalyzer main window.

Steps 1-2 are performed in the Optimize Trace & Generate Profile GUI, and steps 3-5 are completed in the Detect & Track Boutons GUI. Running BoutonAnalyzer in the MATLAB command window opens the main BoutonAnalyzer window, which can be used to set paths and launch one of the two GUIs (Figure B.3).

**Optimize Trace & Generate Profile GUI**

Pressing the Optimize Trace & Generate Profile button opens a window to specify the animal, imaging session, image stack, channel, and axon identifiers. A single axon trace and the corresponding image stack (up to 3 stacks if data from more than one channel is available) can be loaded (Figure B.4).
**Navigation**

Default zoom and pan figure tools of MATLAB can be used to independently focus on different regions of interest on the three projections. **Note that all shortcut keys are disabled if any MATLAB figure tool is active.** Alternatively, mouse scroll and arrow keys can also be used to perform zoom and pan operations (see Table B.1 for the complete list of shortcuts). The View Options panel contains additional display options. Full and Tube Projections show the full stack.
projection or the background-removed projection of the axon. Brightness can be adjusted by changing values in the Intensity Range field and pressing the return key, or by using the “−” and “=” shortcut keys. The Channel option will only be activated if multiple channels are loaded.

**Trace optimization**

The Optimize button initiates trace optimization. Once trace optimization is completed, the radio button related to the optimized trace view will be enabled. Toggling between the Initial and Optimized trace view options will display the corresponding trace on all projections. The Settings menu contains a link to the parameter file, which can be edited to change the parameters of all BoutonAnalyzer functions.

**Generation of axon intensity profiles**

After trace optimization, intensity profiles can be generated and saved for all loaded channels by pressing the Generate Profile & Save button.

**Detect & Track Boutons GUI**

Pressing the Detect & Track Boutons button in the main BoutonAnalyzer window opens the load data panel for the selection of axon profiles in one or more imaging sessions. Axon traces will be displayed superimposed on the maximum intensity projection of the image stack (Figure B.5).

**Navigation**

The zoom, pan, and rotate figure tools of MATLAB perform the expected operations. Mouse scroll and arrow keys can also be used to zoom and pan respectively. Note that all shortcut keys are disabled if any MATLAB figure tool is active. The Intensity Range field allows one to adjust the range of intensities displayed on the screen. Note that this operation only affects the display and
not the data used to perform calculations. The Shift field makes it possible to change the relative positions of axons in the image. Axons can be viewed on raw and normalized intensity projections.

**Figure B.5:** Detect & Track Boutons GUI

*Registration of traces across multiple imaging sessions*

Multiple traces can be registered by identifying corresponding vertices in all traces. The currently selected vertex will be indicated by a bright green circle. Dark green circles will appear on the remaining traces as suggested corresponding vertices. Positions of the selected vertices can be
adjusted by using “,” and “.” keys. Upon satisfactory alignment of the selected vertices, traces can be registered by right-clicking on one of the selected vertices and choosing the Add Landmark option in the context menu. Placing Landmarks every ~15μm along the traces should be sufficient for the subsequent automated matching of boutons.

**Annotation of traces**

Cross-over regions, terminal boutons, and locally noisy regions along an axon must be excluded from the analysis. Such regions can affect the normalization and, hence, detection and measurement of boutons. Such a region can be demarcated by placing a pair of vertices on one of the traces. Positions of these vertices can be adjusted by using “,” and “.” keys. Right-clicking on a selected vertex will open a context menu with options to ignore or add the enclosed region in one or all imaging sessions.

**Detection, editing, and matching boutons in multiple imaging sessions**

Boutons can be automatically detected as peaks along the intensity profiles by pressing the Detect Peaks button. Once bouton detection is completed, the Edit Peaks and Match Peaks modes will be enabled. In the Edit Peaks mode, positions of the detected peaks are indicated by triangles. Individual trace locations can be examined on the intensity profile by choosing the Show on Profile context menu option. A peak can be added or removed by selecting a vertex and using the appropriate context menu option. The detected peaks are matched automatically by a greedy nearest-neighbor algorithm using the landmarks provided in the trace alignment step. These matches can be reviewed in the Match Peaks mode. Matches can be broken by clicking on lines connecting pairs of peaks. New matches can be created by selecting multiple peaks, followed by choosing the Add Peaks context menu option. Upon saving changes, results are compiled in .txt and .mat files located in the \Results folder.
Shortcuts

<table>
<thead>
<tr>
<th>Key</th>
<th>Optimize Trace &amp; Generate Profile GUI</th>
<th>Detect &amp; Track Boutons GUI</th>
</tr>
</thead>
<tbody>
<tr>
<td>=</td>
<td>Increase brightness</td>
<td>Increase brightness</td>
</tr>
<tr>
<td>-</td>
<td>Decrease brightness</td>
<td>Decrease brightness</td>
</tr>
<tr>
<td>Mouse scroll</td>
<td>Zoom in/out</td>
<td>Zoom in/out</td>
</tr>
<tr>
<td>Up arrow</td>
<td>Pan up</td>
<td>Pan up</td>
</tr>
<tr>
<td>Down arrow</td>
<td>Pan down</td>
<td>Pan down</td>
</tr>
<tr>
<td>Left arrow</td>
<td>Pan left</td>
<td>Pan left</td>
</tr>
<tr>
<td>Right arrow</td>
<td>Pan right</td>
<td>Pan right</td>
</tr>
<tr>
<td>d</td>
<td>De-select all vertices</td>
<td></td>
</tr>
<tr>
<td>Comma (,) &amp; period (.)</td>
<td>Move selected vertex along the trace</td>
<td></td>
</tr>
<tr>
<td>v</td>
<td>View/hide overlay elements</td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>Add landmark, peak, or matches</td>
<td></td>
</tr>
<tr>
<td>r</td>
<td>Remove peak</td>
<td></td>
</tr>
</tbody>
</table>

Table B.1: List of BoutonAnalyzer shortcuts

List of Steps

1. Launch BoutonAnalyzer from MATLAB command window.
2. Set paths for Images, Traces, Profiles, and Results. Default paths point to the sample data.
3. Choose the Optimize Trace & Generate Profile GUI.
4. Load a single trace and image stack(s).
5. Perform optimization, inspect the optimized trace, generate and save the profile.
6. Repeat step 5 for traces of the same axon in all available imaging sessions.
7. Launch BoutonAnalyzer again and choose the Detect & Track Boutons GUI.
8. Specify imaging sessions to simultaneously load profiles of an axon in multiple sessions. Upon loading the data, traces will be shown overlaid on the background-removed xy projections of the axon in different imaging sessions.
9. Add a landmark to align the traces. To do so, click on any vertex of one of the axon traces. A bright green circle will mark the currently selected location. Dark green circles will indicate guesses for the corresponding locations on the remaining traces.
10. Select and adjust the positions of these markers by using “,” and “.” keys until the correspondence is precise.
11. Right-click on one of the markers and select Add Landmark. A line connecting these positions will appear. Assigned landmark can be removed by right-clicking on one of the lines and selecting the Remove Landmark option.

12. Once at least two landmarks are added, the Annotate Traces mode will be enabled.

13. If there are regions on the axon that must be ignored from further analyses switch to the Annotate Traces mode. Select two vertices on a single trace that demarcate the region to be ignored. Right-click on one of the vertices to open the context menu. Select an option to either ignore the region in the current trace only or in all the loaded traces. Ignored regions can be readmitted by choosing the Add context menu options.

14. Select the Detect Peaks mode to fit the intensity profiles and detect boutons. The Edit Peaks and Match Peaks modes will be enabled after peak detection is completed.

15. In the Edit Peaks mode, detected peaks appear as triangles on the traces. Right-click on a trace vertex or a peak to open the context menu. Select the appropriate option to add new peaks or remove existing ones.

16. Inspect locations along the intensity profile using the Show on Profile option in the context menu.

17. The Match Peaks mode shows boutons that were matched based on the previously provided alignment.

18. Select multiple peaks on different traces. Right-click on a selected peak to match or flag the selected peaks. Click on lines that connect peaks to delete the matches.

19. Save the results.

**Sample Data**

Image stacks acquired under three different microscopy conditions within a short period of time (< 1 hour) are included in the .mat format in the \sampledata\Images folder. Details regarding the dataset can be found in chapter two. Animal identifier is DL1, section identifier is S2, and channel identifier is Gr. The 3 imaging sessions are identified by letters A, B, and C. Three axons (A001, A002 and A003), manually traced in all three imaging sessions, are provided in the \sampledata\Traces folder.
About

BoutonAnalyzer is developed by the Neurogeometry group at the Department of Physics and the Center for Interdisciplinary Research on Complex Systems at Northeastern University, Boston.

Acknowledgments

We would like to thank Seyed Mousavi Kahaki and Shih-Luen Wang for testing BoutonAnalyzer. This software is supported by the NIH grant R01 NS091421.
Appendix C

Here we formulate and solve a model of associative memory storage which incorporates a number of biologically motivated features of synaptic connectivity. This model is related to the perceptron (Minsky and Papert, 1988; Rosenblatt, 1957), but includes various generalizations and constraints. Specifically, the postsynaptic neuron can receive inputs from multiple excitatory or inhibitory cell classes which have distinct firing rates. The postsynaptic neuron robustly learns associations by modifying the input weights, while its firing threshold remains fixed. Finally, we consider two types of constraints on the input connection weights, $l_0$ norm constraint (fixed number of non-zero weight inputs), and $l_1$ norm constraint (fixed overall magnitude of input weights). The unconstrained case was previously solved by Chapeton et al. (Chapeton et al., 2012), and the results are included in chapter four for comparison. Other models, which include only some of the constraints listed above, have been considered in a number of studies [see e.g. (Amit et al., 1989b; Brunel et al., 2004; Cover, 1965; Edwards and Anderson, 1975; Gardner, 1988; Gardner and Derrida, 1988; Sherrington and Kirkpatrick, 1975; Viswanathan, 1993)].

The model

Consider a postsynaptic neuron receiving $N$ potential inputs belonging to distinct classes (see Figure 4.1). The inputs are classified based on their excitatory/inhibitory nature and their firing probabilities. The postsynaptic neuron must learn a set of $m$ input-output associations, given the following:

The weights of input connections, $J_j$ ($j = 1, \ldots, N$), are sign-constrained as defined by their class, $J_j g_j \geq 0$, where $g_j$ is +1 for excitatory inputs and -1 for inhibitory.
The neuron is constrained to have a fixed number of non-zero weight connections ($l_0$ norm constraint) or a fixed overall magnitude of connection weights ($l_1$ norm constraint).

The inputs, $X^\mu_j$ ($\mu = 1, \ldots, m$), and outputs, $y^\mu$, are binary (0 or 1) and are randomly drawn from input-class dependent probability distributions.

The firing threshold, $h$, of the postsynaptic neuron is fixed and does not change during learning.

The associations, $\{X^\mu_j, y^\mu\}$, must be learned robustly, which is enforced through the robustness parameter, $\kappa$.

The model can be summarized mathematically as:

$$
\left(2y^\mu - 1 \right) \left( \sum_{j=1}^{N} J_j X^\mu_j - h \right) > \kappa \geq 0, \quad \mu = 1, \ldots, m
$$

$$
\sum_{j=1}^{N} |J_j|^l = N w_l, \quad l = 0, 1
$$

$$
J_j, g_j \geq 0, \quad j = 1, \ldots, N
$$

$$
\text{Prob}\left(X^\mu_j = 0\right) = \frac{1 - f_j}{f_j}, \quad X^\mu_j = 1
$$

$$
\text{Prob}\left(y^\mu = 0\right) = \frac{1 - f_{out}}{f_{out}}, \quad y^\mu = 1
$$

(C.1)

In these expressions, $f_j$ is the firing probability of input class $j$, $f_{out}$ is the firing probability of the postsynaptic neuron, and $l = 0$ in the $l_0$ norm is defined in the limit of $l \to 0$.

In the large $N$ limit, and under the assumption that the firing probabilities $f_j$ and $f_{out}$ do not scale with $N$, it can be shown that the average input to the postsynaptic neuron is of order $h$, and that the fluctuations about this average are of order $\kappa$ (Chapeton et al., 2012). This fact motivates the following scaling of the model variables:
Here, the rescaled weights, $J_j$, robustness, $\kappa$, and parameter $w_i$ do not scale with $N$.

Substituting the rescaled variables into Eqs. (C.1) we arrive at expressions which are governed by only intensive parameters $f_j, f_{out}, \kappa$, and $w_i$:

\[
J_j = \frac{h}{N} J_j; \quad \kappa = \frac{h}{\sqrt{N}} \kappa; \quad w_i = \frac{h}{N} \beta_i
\]

(C.2)

\[
\begin{align*}
\left(2y^\mu - 1\right) \left(\frac{1}{N} \sum_{j=1}^{N} J_j X^\mu_j - 1\right) > \frac{\kappa}{\sqrt{N}}, & \quad \mu = 1, m \\
\frac{1}{N} \sum_{j=1}^{N} |J_j| = w_i, & \quad l = 0,1 \\
J_j g_j \geq 0, & \quad j = 1, N \\
\text{Prob}\left(X^\mu_j\right) = \begin{cases} 
1 - f_j, & X^\mu_j = 0 \\
f_j, & X^\mu_j = 1 
\end{cases}; & \quad \text{Prob}\left(y^\mu\right) = \begin{cases} 
1 - f_{out}, & y^\mu = 0 \\
f_{out}, & y^\mu = 1 
\end{cases}
\end{align*}
\]

(C.3)

**Solution of the model**

In this section we outline the main steps for solving the model for the $l_0$ and $l_1$ cases. A more detailed calculation, which follows similar derivation steps, was previously published for the unconstrained case (Chapeton et al., 2012).

We begin by calculating the fraction of the connection weight space, $\Omega(X^\mu_j, y^\mu)$, in which Eqs. (C.3) hold for a given set of associations, $\{X^\mu_j, y^\mu\}$.
\[
\Omega(X_j^\mu, y^\mu) = \frac{\int \prod_{j=1}^N dJ_j \prod_{\mu=1}^m \theta \left( (2y^\mu - 1) \left( \frac{1}{N} \sum_{j=1}^N X_j^\mu - 1 \right) \right) \prod_{j=1}^N \theta (J_j g_j) \delta \left( \frac{1}{N} \sum_{j=1}^N J_j^\nu - w_l \right)}{\int \prod_{j=1}^N dJ_j \delta \left( \frac{1}{N} \sum_{j=1}^N J_j^\nu - w_l \right)}
\]

(C.4)

Here, \( \theta \) denotes the Heaviside step function. The denominator in Eq. (C.4) can be calculated for arbitrary values of \( l > 0 \),

\[
S_l = \int \prod_{j=1}^N dJ_j \delta \left( \frac{1}{N} \sum_{j=1}^N J_j^\nu - w_l \right) = \frac{\left( 2 \Gamma \left( \frac{1}{l} \right) \left( w_l N \right)^{\frac{1}{l}} \right)^N}{w_l \Gamma \left( \frac{N}{l} \right)}
\]

(C.5)

The typical fraction of the solution space volume, \( \Omega_{\text{typical}} \), is defined through the averaging of \( \ln \left( \Omega(X_j^\mu, y^\mu) \right) \) over the set of associations and is calculated by introducing \( n \) replica systems,

\[
\ln \left( \Omega_{\text{typical}} \right) = \left\langle \ln \left( \Omega(X_j^\mu, y^\mu) \right) \right\rangle_{X_j^\mu, y^\mu} = \lim_{n \to 0} \left\langle \Omega(X_j^\mu, y^\mu)^n \right\rangle_{X_j^\mu, y^\mu} - 1
\]

(C.6)

The quantity \( \left\langle \Omega(X_j^\mu, y^\mu)^n \right\rangle_{X_j^\mu, y^\mu} \) in this expression is expressed in terms of a single multidimensional integral:
Below, we outline the main steps of calculation of Eq. (C.7). We begin by decoupling the averaging over the inputs and outputs, \( X_j^\mu \) and \( y^\mu \), through the introduction of a new variable \( \lambda^{a,\mu} = \frac{1}{\sqrt{N}} \sum_{j=1}^{N} J^{a}_{j} X_{j}^\mu - 1 \):

\[
\left\langle \Omega(X^\mu_j, y^\mu) \right\rangle_{X^\mu_j, y^\mu} = S^{-n}_i \int \prod_{a,j=1}^{n,N} dJ_{a,j}^a \prod_{\mu,a=1}^{m,n} \theta \left( (2 y^\mu - 1) \left( \frac{1}{N} \sum_{j=1}^{N} J^{a}_{j} X_{j}^\mu - 1 \right) - \frac{\kappa}{\sqrt{N}} \right) \times \\
\prod_{j,a=1}^{N,n} \theta \left( \frac{1}{N} \sum_{j=1}^{N} |J^{a}_{j}| - w_{j} \right) \right\rangle_{X^\mu_j, y^\mu}
\]

(C.7)

Next, the step functions and the \( \delta \)-functions are replaced with their Fourier representations,

\[
\left\langle \Omega(X^\mu_j, y^\mu) \right\rangle_{X^\mu_j, y^\mu} = S^{-n}_i \int \prod_{a,j=1}^{n,N} dJ_{a,j}^a \prod_{\mu,a=1}^{m,n} \frac{d \lambda^{a,\mu}}{\sqrt{N}} \left\langle \prod_{\mu,a=1}^{m,n} \theta \left( (2 y^\mu - 1) \lambda^{a,\mu} - \kappa \right) \right\rangle_{x, y} \times \\
\prod_{j,a=1}^{N,n} \delta \left( 1 + \frac{\lambda^{a,\mu}}{\sqrt{N}} - \frac{1}{N} \sum_{j=1}^{N} J^{a}_{j} X_{j}^\mu \right) \prod_{a=1}^{n} \theta \left( J^{a}_{j} g_{j} \right) \prod_{a=1}^{n} \delta \left( \frac{1}{N} \sum_{j=1}^{N} |J^{a}_{j}| - w_{j} \right) \right\rangle_{x, y}
\]

(C.8)

Next, the step functions and the \( \delta \)-functions are replaced with their Fourier representations,

\[
\theta \left( (2 y^\mu - 1) \lambda^{a,\mu} - \kappa \right) = \int \frac{d' u^{a,\mu} d\hat{u}^{a,\mu}}{2\pi} e^{i \hat{u}^{a,\mu} \left( (2 y^\mu - 1) \lambda^{a,\mu} - \kappa - u^{a,\mu} \right)}
\]

(C.9)

Symbol \( d' \) in this expression and thereafter is designated for 0 to \( \infty \) integration, whereas \( d \) is used for integration from -\( \infty \) to \( \infty \).

After performing the averaging over the associations, we arrive at:
\[
\left\langle \Omega \left( X_{j}^\mu, y^\mu \right) \right\rangle_{X_{j}^\mu, y^\mu} = S^{-1} \int \prod_{\mu, a=1}^{m, n} \frac{d \hat{\lambda}^{a,\mu} d \hat{a}^{a,\mu}}{2\pi} \prod_{\mu, a=1}^{m, n} \frac{d^2 u_{a,\mu} d^2 \hat{u}_{a,\mu}}{2\pi} \prod_{a=1}^{n} \frac{d^2 k_{a}}{2\pi / N} \times \\
\sum_{\mu, a=1}^{m, n} \left[ \int \prod_{\mu=1}^{m} f_{out}^\mu e^{i \sum_{a=1}^{k_{a}} \hat{g}_{a,\mu}^j} + (1 - f_{out}) e^{-i \sum_{a=1}^{k_{a}} \hat{g}_{a,\mu}^j} \right] \times \prod_{j=1}^{N} \prod_{a=1}^{n} \left[ e^{-\frac{1}{N} \sum_{a=1}^{k_{a}} \hat{\lambda}_{a,\mu, j}^j + \frac{1}{2} \sum_{a=1}^{k_{a}} \hat{g}_{a,\mu, j}^j} \right] e^{\prod_{a=1}^{n} \theta(J_{j}^a g_{j}) \prod_{a=1}^{n} e^{ik_{a}^j}}
\]

(C.10)

Variables \( \hat{\lambda} \) and \( \hat{k} \) in this expression arise from Fourier representations of the two sets of \( \delta \)-functions in Eq. (C.8).

At this point we introduce two sets of order parameters which make it possible to decouple the products containing indices \( j \) and \( \mu \),

\[
\frac{1}{N} \sum_{j=1}^{N} f_{j} J_{j}^a = 1 + \frac{S_a}{\sqrt{N}}, \quad \frac{1}{N} \sum_{j=1}^{N} f_{j} \left( 1 - f_{j} \right) J_{j}^a J_{j}^b = q_a^{ab}
\]

(C.11)

The resulting integrals are calculated with the steepest descent method, assuming the existence of a replica symmetric saddle point, i.e. \( \hat{k}^a = \hat{k}, \hat{s}^a = \hat{s}, \hat{q}^{aa} = q_0, \hat{q}^{ab} = q, \hat{q}^{aa} = \hat{q}_0 \), and \( \hat{q}^{ab} = \hat{q} \). Based on the insight from previous calculations (Chapeton et al., 2012), we introduce the following substitutions, \( t = -i \hat{q} \), \( y = \sqrt{f_{j} \left( 1 - f_{j} \right)} J \), \( u_{z} = \frac{\kappa \pm s}{\sqrt{2q}} \), \( \varepsilon = \frac{q_0 - q}{\sqrt{2q}} \), \( z = \frac{i \hat{g}}{2\sqrt{t}} \),

\[
\delta = \frac{t + i \hat{q}_{0}}{t}, \quad \eta = -i \kappa t^{-1/2}, \quad \text{and give a summary of the results:}
\]
\[
\frac{\ln(\Omega_{\text{typical}})}{N} = -\ln \left(\frac{2 \Gamma \left(\frac{1}{2}\right)}{\Gamma \left(\frac{1}{l}\right)}(lw_i, e)^\frac{1}{l}\right) - w_i \eta^2 + 2z\sqrt{t} + \frac{2\kappa^2}{(u_+ + u_-)^2}(\delta - \epsilon + \epsilon\delta)t + \alpha G_E(\epsilon, u_-, u_+) + G_s(z, \delta, \eta, t)
\]

\[
G_E(\epsilon, u_-, u_+) = f_{\text{out}} \int \frac{dx}{\sqrt{\pi}} e^{-x^2} \ln \left(1 + \text{erf}\left(\frac{x - u_-}{\sqrt{\epsilon}}\right)\right) + (1 - f_{\text{out}}) \int \frac{dx}{\sqrt{\pi}} e^{-x^2} \ln \left(1 + \text{erf}\left(\frac{x - u_+}{\sqrt{\epsilon}}\right)\right)
\]

\[
G_s(z, \delta, \eta, t) = \frac{1}{N} \sum_{j=1}^{N} \int \frac{dx}{\sqrt{\pi}} e^{-x^2} \ln \left(\int d'ye^{-x^2} \ln \left(\frac{-\delta^2 + f_j(1-f_j)^2}{\sqrt{\epsilon}} \right) + \left(\frac{-f_j \delta}{\sqrt{\epsilon}} \right)\right) - \ln \sqrt{f_j(1-f_j)}t
\]

(C.12)

In this expression \( \alpha = m/N \) is referred to as the associative memory storage capacity, while the variables \( \{\epsilon, u_-, u_+, z, \delta, \eta, t\} \) are defined through the following seven saddle-point equations:

\[
\frac{\partial G_E(\epsilon, u_-, u_+)}{\partial u_-} + \frac{\partial G_E(\epsilon, u_-, u_+)}{\partial u_+} = 0
\]

\[
\frac{\partial G_E(\epsilon, u_-, u_+)}{\partial \epsilon} = \frac{2\kappa^2}{\alpha(u_+ + u_-)^2} t(1 - \delta)
\]

\[
\frac{\partial G_E(\epsilon, u_-, u_+)}{\partial \epsilon} \left(1 + \epsilon\right) + \frac{1}{2} \frac{\partial G_E(\epsilon, u_-, u_+)}{\partial u_-} u_- + \frac{1}{2} \frac{\partial G_E(\epsilon, u_-, u_+)}{\partial u_+} u_+ = \frac{2\kappa^2}{\alpha(u_+ + u_-)^2} t
\]

\[
\frac{\partial G_s(z, \delta, \eta, t)}{\partial \eta} = w_j
\]

\[
\frac{\partial G_s(z, \delta, \eta, t)}{\partial \eta} = -2\sqrt{t}
\]

\[
\frac{\partial G_s(z, \delta, \eta, t)}{\partial \delta} = -\frac{2\kappa^2}{(u_+ + u_-)^2}(1 + \epsilon)t
\]

\[
\frac{\partial G_s(z, \delta, \eta, t)}{\partial \delta} \left(1 - \delta\right) - \frac{\partial G_s(z, \delta, \eta, t)}{\partial \eta} \frac{l}{2} \eta + \frac{\partial G_s(z, \delta, \eta, t)}{\partial t} t = -\frac{2\kappa^2}{(u_+ + u_-)^2} t
\]

(C.13)
Critical capacity

At the critical associative memory storage capacity, the saddle point Eqs. (C.13) can be simplified because $\Omega_{\text{typical}}$ tends to zero, and therefore, $(q_0 - q)$ goes to zero as well. In this limit the seven saddle point equations can be expanded asymptotically to the leading orders in $1/(q_0 - q)$. After eliminating $t$, $\delta$, and $\eta$ from these equations, and introducing a new variable $x$, which is related to $\alpha$ through Eq. (C.17), we arrive at the final result:

\begin{align*}
\lambda &= 0,1 \text{ norm constrained cases:} \\
I_0 I_i (-u_-,0) &= (1- f_{\text{out}}) I_i (-u_+,0) \\
\frac{1}{N} \sum_{i=1}^{N} \left( \frac{1}{f_i (1-f_i)} \right) I_i \left( g_i z \sqrt{f_i (1-f_i)} + \frac{(\kappa x-z)}{w_i \sqrt{f_i (1-f_i)}} \delta_{l,1,i} \sqrt{\frac{2(\kappa x-z)Q}{w_i}} \delta_{l,0} \right) &= w_i Q^i \\
\frac{1}{N} \sum_{i=1}^{N} \left( \frac{f_i g_i}{f_i (1-f_i)} \right) I_i \left( g_i z \sqrt{f_i (1-f_i)} + \frac{(\kappa x-z)}{w_i \sqrt{f_i (1-f_i)}} \delta_{l,1,i} \sqrt{\frac{2(\kappa x-z)Q}{w_i}} \delta_{l,0} \right) &= Q \\
\frac{1}{N} \sum_{i=1}^{N} \left( \frac{f_i g_i}{f_i (1-f_i)} \right) I_i \left( g_i z \sqrt{f_i (1-f_i)} + \frac{(\kappa x-z)}{w_i \sqrt{f_i (1-f_i)}} \delta_{l,1,i} \sqrt{\frac{2(\kappa x-z)Q}{w_i}} \delta_{l,0} \right) &= \frac{2Q^2 \kappa^2}{(u_+ + u_-)^2} \\
Q &= 2(\kappa x-z) f_{\text{out}} I_0 (-u_-,0) + (1-f_{\text{out}}) I_0 (-u_+,0) x \\
u_+ + u_- &\geq 0; \quad x \geq 0; \quad (\kappa x-z) \delta_{l,0} \geq 0
\end{align*}

(C.14)

In comparison, the corresponding equations for the unconstrained case, written in the notation of Eq. (C.14), have the form (Chapeton et al., 2012):
Unconstrained case:

\[ f_{out} I_1 (-u_-,0) = (1 - f_{out}) I_1 (-u_+,0) \]

\[ z = \kappa x \]

\[
\frac{1}{N} \sum_{j=1}^{N} \frac{f_j g_j}{\sqrt{f_j (1 - f_j)}} I_1 \left( \frac{f_j g_j z}{\sqrt{f_j (1 - f_j)}}, 0 \right) = Q
\]

\[
\frac{1}{N} \sum_{j=1}^{N} I_2 \left( \frac{f_j g_j z}{\sqrt{f_j (1 - f_j)}}, 0 \right) = \frac{2Q^2 \kappa^2}{(u_+ + u_-)^2}
\]

\[ u_+ + u_- \geq 0; \quad x \geq 0 \]

(C.15)

The following special functions were used in Eqs. (C.14) and (C.15):

\[ I_n(a,b) = \int_0^a \frac{dy}{\sqrt{\pi}} e^{-(y+b+a)^2} \left( y + b + (y+b)^2 - b^2 \right)^n \]

\[ I_0(a,b) = \frac{1}{2} \left( 1 - \text{erf} \left( a + b \right) \right) \]

\[ I_1(a,0) = \frac{1}{\sqrt{\pi}} e^{-a^2} - a \left( 1 - \text{erf} \left( a \right) \right) \]

\[ I_2(a,0) = -\frac{2a}{\sqrt{\pi}} e^{-a^2} + (2a^2 + 1) \left( 1 - \text{erf} \left( a \right) \right) \]

(C.16)

The solution of Eqs. (C.14) defines the critical capacity in terms of the model parameters,

\[ \alpha_c \{ \{ g_j \}, \{ f_j \}, f_{out}, \kappa, w_1 \} = 2x^2 \left( f_{out} I_2 (-u_-,0) + (1 - f_{out}) I_2 (-u_+,0) \right) \]

\[ \frac{1}{f_{out} I_1 (-u_-,0) + (1 - f_{out}) I_1 (-u_+,0)} \]

(C.17)

Eqs. (C.14- C.17) were solved with custom MATLAB code to produce the results in the chapter.
Distribution of input weights at critical capacity

The probability density function for input weights can be written in general form as:

\[
p_i(J) = \left\langle \prod_{j=1}^{N} dJ_j \delta(J_i - J) \prod_{\mu=1}^{m} \theta \left( (2y^{\mu} - 1) \left( \frac{1}{N} \sum_{j=1}^{N} J_j X_j^{\mu} - 1 \right) - \frac{\mathcal{K}}{\sqrt{N}} \right) \prod_{j=1}^{N} \theta(J_j g_j) \delta \left( \frac{1}{N} \sum_{j=1}^{N} |J_j|' - w_l \right) \right\rangle_{X^{\mu}, y^{\mu}}
\]

(C.18)

Eq. (C.18) can be cast in a form that closely resembles Eq. (C.7), allowing one to use the results from the previous sections. To this end, we introduce \( n \) replicas and take the limit of \( n \to 0 \) after averaging over the associations,

\[
p_i(J) = \lim_{n \to 0} \left\langle \prod_{a,j=1}^{N,n} dJ_j^a \delta(J_i^a - J) \prod_{\mu,a=1}^{m,n} \theta \left( (2y^{\mu} - 1) \left( \frac{1}{N} \sum_{j=1}^{N} J_j^a X_j^{\mu} - 1 \right) - \frac{\mathcal{K}}{\sqrt{N}} \right) \prod_{j,a=1}^{N,n} \theta(J_j^a g_j) \prod_{a=1}^{n} \delta \left( \frac{1}{N} \sum_{j=1}^{N} |J_j^a|' - w_l \right) \right\rangle_{X^{\mu}, y^{\mu}}
\]

(C.19)

Following the steps described in Eqs. (C.7- C.12) we arrive at:

\[
p_i(J) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} dx e^{-x^2} \frac{e^{\frac{1}{2} \sqrt{\mathcal{J}^2 + 2\delta(\sqrt{\mathcal{J}} - f, g, z)_{g,J-f,J}}}}{\sqrt{\int dJ' e^{\frac{1}{2} \sqrt{\mathcal{J}^2 + 2\delta(\sqrt{\mathcal{J}} - f, g, z)_{g,J-f,J}}}}}
\]

(C.20)
At critical capacity, Eq. (C.20) can be simplified, leading to the probability density functions of connection weights for the various input types. A characteristic feature of these distributions is that they contain finite fractions of zero-weight connections:

\[ l = 0,1 \text{ norm constrained cases:} \]

\[
p_i(J) = \theta(g_i J) \frac{1}{\sqrt{2\pi \Sigma_i}} \left(1 - \frac{(\kappa x - z) Q \Sigma_i^2}{w_i J^2} \delta_{l,0} \right)_+ e^{\frac{-J}{\sqrt{2\Sigma_i}} \sqrt{\frac{J}{1-f_i}} \frac{\left((\kappa x - z) Q \Sigma_i^2\right)^2}{w_i}} + \]

\[
\begin{align*}
I_0 &= -g_z \sqrt{\frac{f_i}{1-f_i}} - \frac{\left((\kappa x - z) Q \Sigma_i^2\right)^2}{w_i} \delta_{l,1} - \frac{2(\kappa x - z) Q \Sigma_i^2}{w_i} \delta_{l,0} \delta(J) \\
\Sigma_i &= \frac{\sqrt{2}}{\sqrt{f_i(1-f_i)Q}}
\end{align*}
\]

(C.21)

Parameters \( \Sigma_i \) in this expression describe the widths of the input weight distributions. Plus sign in the subscript of the first equation in (C.21) denotes the positive part function. Note that in the \( l_0 \) case this function leads to gaps in the connection weight distributions. In this case, the postsynaptic neuron at critical capacity does not have weak connections, i.e. connection weights smaller than \( \Sigma_i \sqrt{(\kappa x - z) Q / w_0} \) in magnitude. We would like to mention that for a single class of sign-unconstrained but \( l_0 \) norm constrained inputs, a gap in the distribution of connection weights was previously reported by Bouten et al. (Bouten et al., 1990).

The corresponding probability density functions in the unconstrained case, written in the notation of Eq. (C.21), have the form (Chapeton et al., 2012):
Unconstrained case:

\[ p_i(J) = \theta(g_i J) \frac{1}{\sqrt{2\pi \Sigma_i}} e^{-\left( \frac{J - \frac{1}{\Sigma_i} f_i}{\sqrt{1-f_i}} \right)} + I_0 \left( -g_i z \sqrt{\frac{f_i}{1-f_i}}, 0 \right) \delta(J) \]

(C.22)

**Connection probabilities and distributions of non-zero connection weights**

Connection probabilities and uPSP amplitudes are routinely measured in electrophysiological experiments. Connection probabilities in the model, \( P_{\text{con}} \), are defined as the fractions of non-zero weight inputs. These fractions can be obtained from Eqs. (C.21) and (C.22):

\[ l = 0,1 \text{ norm constrained cases:} \]

\[ P_{l\text{con}} = I_0 \left( g_i z \sqrt{\frac{f_i}{1-f_i}} + \frac{(\kappa x - z)}{w_i \sqrt{f_i(1-f_i)}} \delta_{l,1} + \sqrt{\frac{2(\kappa x - z) Q}{w_i}} \delta_{l,0}, 0 \right) \]

Unconstrained case:

\[ P_{l\text{con}} = I_0 \left( g_i z \sqrt{\frac{f_i}{1-f_i}}, 0 \right) \]

(C.23)

Probability density functions of non-zero input weights in the model, which correspond to experimental distributions of uPSP amplitudes, are derived from Eqs. (C.21) and (C.22) as well:
\[
 l = 0, 1 \text{ norm constrained cases:}
\]
\[
p_i(J) = \frac{\left(1 - \left(\frac{\kappa x - z}{\mathcal{Q} \Sigma^2}{J^2}\delta_{1,0}\right)\right)_+ e^{-\left(\frac{J}{\sqrt{2\Sigma} \sqrt{\frac{f_i}{1-f_i}} \frac{\kappa x - z}{w_i \sqrt{f_i(1-f_i)}} (\mathcal{Q} \delta_{1,0} + \delta_{0,0})})^2\right)}}{\sqrt{2\pi \Sigma} I_0 \left( g_i z \sqrt{f_i(1-f_i)} \right)}
\]

Unconstrained case:
\[
p_i(J) = \frac{e^{-\left(\frac{J}{\sqrt{2\Sigma} \sqrt{\frac{f_i}{1-f_i}}}ight)^2}}{\sqrt{2\pi \Sigma} I_0 \left( g_i z \sqrt{f_i(1-f_i)} \right)}
\]

(C.24)

We note that the connection weight distributions for the unconstrained and \(l_1\) norm constrained cases are Gaussian functions truncated at zero. The \(l_0\) norm distribution is significantly different; it is non-Gaussian and contains a gap near zero.

**Numerical solutions with methods of convex optimization**

Here we describe the numerical algorithms used to validate the solutions to the unconstrained and \(l_1\) norm constrained problems, Eq. (C.3). These simulations were used to generate the results of Figure 4.2A, C, D, F. Both problems are convex, and hence, can be solved within the standard constrained optimization framework. To this end, we turn the inequalities of Eq. (C.3) into soft constraints to make the problem feasible, and add a small regularizing potential to make the solution unique:
This problem can be solved in the primal-dual Lagrangian framework (Boyd and Vandenberghe, 2004). We construct the primal Lagrange function by combining the objective function of Eq. (C.25) with the constraints weighted by Lagrange multipliers:

\[
\min_{s,j_j} \left( s + \varepsilon \sum_{j=1}^{N} J_j^2 \right); \quad \varepsilon \geq 0 \\
- \frac{(2y^\mu - 1)}{N} \sum_{j=1}^{N} J_j X_j^\mu + (2y^\mu - 1) + \frac{\kappa}{\sqrt{N}} \leq s, \quad \mu = 1, \ldots, m \\
g_j J_j \leq s, \quad j = 1, \ldots, N \\
\frac{1}{N} \sum_{j=1}^{N} |J_j| - w_i = 0, \quad i = 1 
\]

(C.25)

The dual Lagrangian and the optimal values of connection weights are calculated as follows:

\[
L_p(s, J_j; \alpha_\mu, \beta_j, \gamma) = s + \varepsilon \sum_{j=1}^{N} J_j^2 + \sum_{\mu=1}^{m} \alpha_\mu \left( - \frac{(2y^\mu - 1)}{N} \sum_{j=1}^{N} J_j X_j^\mu + (2y^\mu - 1) + \frac{\kappa}{\sqrt{N}} - s \right) - \sum_{j=1}^{N} \beta_j (g_j J_j + s) + \gamma \delta_{l,1} \left( \frac{1}{N} \sum_{j=1}^{N} |J_j| - w_i \right) \\
\alpha_\mu \geq 0; \quad \beta_j \geq 0 
\]

(C.26)

\[
L_d(\alpha_\mu, \beta_j, \gamma) = \min_{s,J_j} L_p(s, J_j; \alpha_\mu, \beta_j, \gamma) = \sum_{\mu=1}^{m} \alpha_\mu \left( 2y^\mu - 1 \right) + \frac{\kappa}{\sqrt{N}} - \gamma w_i \delta_{l,1} - \frac{1}{4\varepsilon N} \sum_{j=1}^{N} \left( \sum_{\mu=1}^{m} \alpha_\mu \left( 2y^\mu - 1 \right) X_j^\mu + \beta_j g_j - \gamma g_j \delta_{l,1} \right)^2 + \begin{cases} 0, & \sum_{\mu=1}^{m} \alpha_\mu + \sum_{j=1}^{N} \beta_j = 1 \\
-\infty, & \text{else} \end{cases} \\
J_j = \frac{1}{2\varepsilon} \left( \sum_{\mu=1}^{m} \alpha_\mu \left( 2y^\mu - 1 \right) X_j^\mu + \beta_j g_j - \gamma g_j \delta_{l,1} \right) 
\]
This leads to a relatively simple associated dual problem:

\[
\begin{align*}
\max_{\alpha_\mu, \beta_j, \gamma} & \quad L_d(\alpha_\mu, \beta_j, \gamma) \\
\alpha_\mu & \geq 0; \quad \beta_j \geq 0 \\
\alpha_\mu \left( -\frac{(2y^\mu - 1)}{N} \sum_{j=1}^{N} J_j X_j^\mu + (2y^\mu - 1) + \frac{\kappa}{\sqrt{N}} - s \right) & = 0 \\
\beta_j \left( g_j J_j + s \right) & = 0
\end{align*}
\]

Maximization over the Lagrange multiplier \( \gamma \) can be carried out explicitly,

\[
\begin{align*}
\max_{\alpha_\mu, \beta_j} & \quad \left( \sum_{\mu=1}^{m} \alpha_\mu \left( 2y^\mu - 1 \right) X_j^\mu + \beta_j \right) \\
\alpha_\mu & \geq 0; \quad \beta_j \geq 0 \\
\alpha_\mu \left( -\frac{(2y^\mu - 1)}{N} \sum_{j=1}^{N} J_j X_j^\mu + (2y^\mu - 1) + \frac{\kappa}{\sqrt{N}} - s \right) & = 0 \\
\beta_j \left( g_j J_j + s \right) & = 0 \\
\sum_{\mu=1}^{m} \alpha_\mu + \sum_{j=1}^{N} \beta_j & = 1
\end{align*}
\]

The resulting optimization problem was solved by using a custom MATLAB algorithm, generating the numerical results of Figure 4.2. The connection weights were determined based on the optimum values of the Lagrange multipliers \( \alpha \) and \( \beta \) according to Eq. (C.27).
The critical capacity resulting from these simulations is in good agreement with the theoretical calculations (Figure 4.2A, C). Numerical simulations also reproduce the overall shapes of theoretical distributions of connection weights (Figure 4.2D, F). However, a small deviation in distribution width was observed in the unconstrained model. We attribute this discrepancy to the effect of a finite number of inputs \((N_{inh} = 75 \text{ and } N_{exc} = 425)\). Consistent with this interpretation, the deviation is larger for inhibitory inputs, while it is absent in the \(l_1\) case.

**Numerical solutions with perceptron-type learning rules**

In addition to the above described theoretical solutions and numerical solutions based on convex optimization, the three models considered in this study were also solved numerically with modified perceptron learning rules (see Figure 4.2B, E and Figure 4.5 of the main text). This was necessary for two reasons. First, such numerical solutions establish biological plausibility of the steady-state learning hypothesis, which among other things assumes that neurons can reach the state of critical capacity. Second, since the \(l_0\) norm constrained problem is non-convex, and hence, cannot be solved with convex optimization, we had to find an alternative numerical strategy for this case.

Such numerical simulations in the unconstrained case utilized perceptron learning rule (Engel and Broeck, 2001) with the addition of sign constraints on the weights of excitatory and inhibitory connections. This algorithm was previously described in (Chapeton et al., 2012). To incorporate the \(l_1\) norm constraint, we uniformly rescaled connection weights after every perceptron learning step. In the \(l_0\) norm constraint case, which is a constraint on the number of non-zero weight connections, we monitored this number after every perceptron learning step. If at any point the number of non-zero weight connections exceeded the value allowed by the constraint, a single weakest connection (excitatory or inhibitory) was set to zero. These three modified perceptron learning rules can be summarized as follows:
\[
J_k = J_k + \frac{\beta}{\sqrt{N}} \left( 2y^\mu - 1 \right) X_k^\mu, \quad k = 1, \ldots, N
\]

\[
J_k = J_k \theta(J_k g_k)
\]

\[
l = 0 \text{ case: } J_{\arg \min_j |J_j|} = 0, \quad \text{if } \frac{1}{N} \sum_{j=1}^N \theta(|J_j|) > w_0
\]

\[
l = 1 \text{ case: } J_k = \frac{w_i}{\frac{1}{N} \sum_{j=1}^N |J_j|} J_k
\]

(C.30)

These expressions describe the update of input weights, $J_k$, in response to the presentation of a single association $\mu$, which has not been learned by the neuron. Parameter $\beta$ defines the learning rate. This parameter was set to 0.05 in all numerical simulations. Associations were selected randomly for learning, and for each association the above update rule was applied until the association was learned. Numerical algorithms in all three cases were allowed to continue until the solution satisfying all the constraints was found, or until the maximum number of iterations ($10^6$) was reached. There is no proof of convergence for these procedures, but the results underestimate the theoretical critical capacity by no more than 5%-15% (see Figure 4.2B).
References


Bourne, J.N., and Harris, K.M. (2011). Coordination of size and number of excitatory and inhibitory synapses results in a balanced structural plasticity along mature hippocampal CA1 dendrites during LTP. Hippocampus 21, 354-373.


