

Plasticity and Learning Algorithms in Models of the Single Neuron

Thesis for the degree of *Doctor of Philosophy*
by
Toviah Moldwin

Submitted to the Senate of the Hebrew University
of Jerusalem
September 2023

This work was carried out under the supervision of Prof. Idan Segev.

Contents

Abstract	4
Acknowledgments	5
Preface	7
Introduction.....	10
Neurons: the fundamental building blocks of the brain.....	10
Thinking about neurons mathematically	12
The linear neuron model	14
Supervised classification tasks and the perceptron learning algorithm	15
Evaluating classifiers: Generalization and Capacity	17
Important differences between perceptrons and biological neurons	18
References	23
Chapter 1: Perceptron Learning and Classification in a Modeled Cortical Pyramidal Cell	27
Chapter 2: The gradient clusteron: A model neuron that learns to solve classification tasks via dendritic nonlinearities, structural plasticity, and gradient descent	40
Chapter 3: Asymmetric Voltage Attenuation in Dendrites Can Enable Hierarchical Heterosynaptic Plasticity	70
Chapter 4: A Generalized Framework for the Calcium Control Hypothesis Describes Weight-Dependent Synaptic Plasticity.....	86
Chapter 5: The Calciton: A Simple Neuron Model That Implements Many Learning Rules via the Calcium Control Hypothesis	117
Discussion and Conclusions.....	143

Abstract

The connectionist model of brain function asserts that the brain can best be understood as a computational network of neurons, which individually function as nodes that each perform a simple mathematical operation. The brain can learn to store new information by modifying the connection strengths, or weights, between those neurons. The remarkable success of artificial neural networks and deep learning--which are based on connectionist principles--at solving difficult computational tasks lends credence to the idea that the brain may be operating on similar principles. But how similar are real biological neurons to the simple linear units used in artificial neural networks, both in terms of their input-output function and in terms of how they learn to solve new tasks? This dissertation addresses the above question from a variety of perspectives.

We begin by asking whether biological neurons can implement the perceptron learning algorithm, one of the simplest learning algorithms for artificial neurons. Using a detailed biophysical cable model of a layer 5 cortical pyramidal neuron, it turns out that the answer is “yes” (Moldwin & Segev, 2018).

Having explored whether biological neurons can behave like artificial neurons, we then ask the question in reverse: can we come up with a simple artificial neuron model that captures some of the interesting nonlinear phenomena observed in biology, and then use that model to solve a machine learning task? To this end, we developed the gradient clusteron, a model neuron uses location-dependent dendritic nonlinearities to produce a stimulus-dependent output. We analytically derived a gradient descent learning rule for the Gradient Clusteron and showed that it can indeed learn to solve classification tasks, such as MNIST or the XOR problem, by inducing synapses to attract or repel each other on the dendrite (Moldwin et al., 2021).

Continuing the direction of understanding how the spatial structure of dendrites can influence plasticity and learning, we also looked at heterosynaptic plasticity in cable models of dendrites, proposing that the electrical asymmetry inherent to dendritic trees can allow for a hierarchical supervisory scheme in a dendrite using NMDA spikes (Moldwin et al., 2022).

We then turn to the biological mechanisms underlying synaptic plasticity via the calcium control hypothesis (CCH), with states that the direction and magnitude of plasticity depends on the calcium concentration at the synapse. Building on previous work, we propose a simplified mathematical model for understanding the calcium control hypothesis, known as the FPLR (fixed point – learning rate) rule (Moldwin, Azran, et al., 2023). We then used this rule to model some exciting new results pertaining to behavioral time-scale plasticity (BTSP). We also propose a simple neuron model, known as the Calciton, that uses the calcium-control hypothesis to implement a wide range of learning and plasticity results (Moldwin et al., 2024).

This dissertation is based on 3 published articles (Moldwin et al., 2021; Moldwin, Kalmenson, et al., 2023; Moldwin & Segev, 2020) and 2 additional pre-publication articles which can be found on bioRxiv (Moldwin, Azran, et al., 2023; Moldwin et al., 2024).

Acknowledgments

The story of my PhD was not just an intellectual odyssey; it was a physical journey as well. I was born and raised in the United States of America; I had not seriously considered moving to Israel until I was accepted to the Edmond and Lily Safra Center for Brain Sciences at the Hebrew University of Jerusalem. There are many people who helped make this journey possible, both scientifically and in terms of helping me navigate life in a new country.

I would first like to thank my parents and family, for supporting me in many ways over the course of my doctorate. While some parents might have had reservations about their child moving halfway across the world, my parents were fully supportive throughout the entire process of my doctorate, and I am very appreciative of that.

Menachem Kalmenson and Li Azran both contributed to this work in significant ways and were coauthors on many of the projects in this collection. Without them, this dissertation would not exist, at least not in its present form. It was a pleasure to work with both of them; I admire their work ethic, their commitment, and their tolerance for my occasional micromanagement.

My supervisor, Idan Segev, has been an indispensable resource and source of support throughout the research process. Idan helped me not just with the scientific aspect of this work – especially issues of neuronal biophysics – but also with every aspect of navigating the academic landscape, from making aesthetically pleasing figures to the etiquette of responding to reviewers. What I most appreciate is the freedom that Idan gave me to pursue questions I was interested in and the resources he gave me (especially sending Menachem and Li my way!) to succeed at exploring those questions.

I appreciate the friendship and support I have received from all members of the lab, past and present. The lab has served as a second home for me, and my ~7 years I spent in the lab has been a positive experience mainly due to them. Because many people have passed through the doors of the lab over my time there, I will not list all of them exhaustively for fear of leaving someone out. However, I do want to extend an individual thanks to some of the “old guard” lab members who spent many hours assisting me when I was new in the lab and still learning the ropes. David Beniaguev, Oren Amsalem, Michael Doron, and Guy Eyal were always available for a discussion and to help me untangle some issue with NEURON, machine learning, or neuroscience generally. I also appreciate the work that they did in organizing conference trips; we always managed to enjoy our time together even if the conference wasn’t that great.

Each paper in this dissertation has its own acknowledgement section for people who contributed to those projects, and I will not repeat each of those acknowledgements here. However, I believe that Eyal Gal and Itamar Landau deserve special recognition for their contributions to the development of the G-clusteron algorithm. Menachem and I had been trying to improve the original clusteron algorithm for months, without making much headway. One day, on a whim, I decided to ask Eyal if he had any ideas. After spending a few hours together on the whiteboard, Eyal had a stroke of insight that the problem of moving synapses on a dendrite was similar to a homework problem that he once had to do in statistical physics, which involved simulating particles that exert forces on each other. The following day, Menachem and I were sitting in the conference room trying to work out this solution mathematically, and Itamar Landau (from Haim Sompolinsky’s lab) walked into the room. We explained to Itamar what we were trying to do, and Itamar suggested that we use gradient descent. With Itamar’s help to formulate the gradient descent derivation, the gradient clusteron algorithm was born.

And after some simplifications to the equation, we saw that Eyal's insight had been correct, the gradient descent solution had resulted in a rule where synapses were attracted and repelled from each other!

In addition to my scientific colleagues at ELSC, I also want to thank the administrative staff, who kept things running (mostly) smoothly so I could focus on the science. I especially want to thank Michal Leci, who joined our lab as administrative staff at some point in the middle of my doctorate. Michal was there to help me on a regular basis, especially in my role organizing our weekly lab meetings.

I also want to thank my many friends in the Jerusalem area, including the past and present members of my rock band, *Synfire Chain*. Outside of science, music has occupied a central role in my life in Jerusalem, and I appreciate the band members coming out to rehearse (and occasionally perform and record) on a regular basis for all these years.

Finally, I want to thank all the individuals and families who hosted me for Shabbat and Holiday meals over my time here in Israel. Shabbat is an essential part of life in Israel; without the seventh day of respite with friends and family, the other six days of labor during the week would have no purpose. The people who took me in for Shabbat meals are far too numerous to list individually, but they should know that their hospitality has not gone unnoticed.

Preface

Since the earliest days of computing, computer scientists and programmers have speculated about the relationship between computers and the human mind. In the early 19th century, Lady Ada Lovelace, arguably the first computer programmer and the originator of the computer algorithm, commented that “The Analytical Engine [the early computer that she worked with] has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform. It can follow analysis; but it has no power of anticipating any analytical relations or truths.”

A century later, in 1950, Alan Turing, the “father of modern computing”, wrote an article entitled "Computing Machinery and Intelligence" for the journal *Mind*, where he posed the question ‘can machines think?’ and originated the idea of the Turing Test, an important thought experiment in the philosophy of mind that attempts to more precisely formulate the distinction between humans and machines. John von Neumann, another outstanding, pioneering computer scientist of the same era as Turing, wrote *The Computer and the Brain* (Von Neumann, 1958), a book that analyzed the differences in the computational mechanisms between the computer and the brain.

Meanwhile, on the neuroscience side of things, Warren McCulloch and Walter Pitts published a paper in 1943 entitled “The Logical Calculus Immanent in Nervous Activity” (McCulloch & Pitts, 1943), which attempted to describe the inner-workings of the brain in a mathematical and computational fashion. This work led to the development of the perceptron, a simple, abstract mathematical model of a neuron that could learn to solve classification tasks, like distinguishing between images of handwritten letters or digits. In subsequent work, researchers have shown that by combining many perceptron-like units into multiple-layered “artificial neural networks” (ANNs), it is possible to create extraordinarily powerful computational systems that can learn to solve problems like recognizing speech, playing chess, or even driving a car at a performance level approaching (and sometimes exceeding) human abilities.

The remarkable success of ANNs at solving tasks thought to be only solvable by humans has led theoretical neuroscientists to propose that at least some processes in biological brains can best be modeled and understood as artificial neural networks. It is not obvious, however, that neurons in the brain really compute and learn like the abstract nodes in artificial neural networks. Biology is messy and complicated, and artificial neural networks – even those that contain large numbers of nodes and learnable parameters – can be fully specified by fairly simple mathematical formalisms (although the emergent dynamics in ANNs can also end up being complex, chaotic, and inscrutable).

The role of this dissertation is to bridge the gap between the simple artificial neurons used in ANNs (which we call McCulloch and Pitts neurons, or M&P neurons, although this takes some liberty with the original formalism from their 1943 paper) and real biological neurons. Specifically, we address the question of whether *learning* in biological neurons occurs in the same manner as in M&P neurons, or whether real biological neurons might have features that distinguish them computationally from M&P neurons, requiring biological neurons to implement learning algorithms in a manner that differs from M&P neurons. These differences could potentially make biological neurons inferior to M&P neurons in their learning and computational ability, but there is also the possibility that biology may surprise us with novel approaches to solving learning problems, enabling us to design superior, brain-inspired neural learning algorithms.

The dissertation is composed of five chapters (in addition to the introduction and conclusion) written as independent papers. Each of these chapters has its own reference and acknowledgements section.

The research methodology of this dissertation is entirely computational and theoretical. To explore the differences between biological and artificial neurons, we perform simulations of neurons which are “more similar” to biological neurons than ANNs. As is often the case when doing computational modeling, the level of abstraction we use in our simulations depends on the question we are trying to ask. Some of the simulations in this dissertation use highly detailed biophysical models based on cable equations and constrained by experimental data, other simulations use artificial neurons that are quite similar to M&P neurons with a few extra biologically-inspired tweaks, while yet other simulations are somewhere in between. The guiding principle in choosing the level of abstraction for these analyses was to be “as simple as possible, but not simpler”. In other words, we tried to choose the simplest model that would be able to address the question we were interested in.

One major difference between biological neurons and M&P neurons pertains to the issue of nonlinearities. M&P neurons sum their input linearly – that is to say, if an M&P neuron has two presynaptic inputs, each with a strength of 2, if the two inputs are activated simultaneously, the output of the M&P neuron (before applying the output threshold nonlinearity) will be 4, because $2+2=4$. Real biological neurons, on the other hand, can sometimes integrate their inputs (denoted here by \oplus) in a sublinear manner, resulting in $2\oplus 2=3$, or in a superlinear manner, resulting in $2\oplus 2=5$. The first two chapters of this dissertation deal with these phenomena.

In the first chapter, we implement perceptron learning and classification in a detailed biophysical model of a layer 5 cortical pyramidal cell, which we call the *biophysical perceptron*. Despite the many complexities of the detailed cell model, the biophysical perceptron can basically implement the perceptron learning algorithm without much trouble. We find, however, that the sublinear integration of inputs can somewhat hamper learning and classification capacity when the inputs are placed on the apical tuft. This chapter was published in Frontiers of Computational Neuroscience (Moldwin & Segev, 2020).

In the second chapter, we explore how the superlinear integration of nearby synapses on the dendrites of biological neurons can be exploited to create a novel learning algorithm, called the *gradient clusteron* or *G-clusteron*, where synapses on a simple model dendrite learn to form spatial clusters of correlated inputs by attracting or repelling each other. In addition to being able to solve the classic MNIST image classification task with high accuracy, the G-clusteron can also solve the XOR problem, which famously cannot be solved by a linear M&P perceptron. Although not all aspects of the G-clusteron are biological, it serves as proof of principle that the nonlinear quirks biological neurons can serve as inspiration for novel learning algorithms. This work was published in PLOS computational biology(Moldwin et al., 2021).

While both the biophysical perceptron and G-clusteron utilize model neurons that take into account more aspects of biology than M&P neurons, the learning algorithms themselves were either simply copied from the extant algorithm for M&P neurons (in the case of the biophysical perceptron) or analytically derived to solve a certain task (in the case of the G-clusteron). Neither the biophysical perceptron nor the G-clusteron learning algorithms themselves were based on our knowledge of how plasticity happens mechanistically in real neurons. The next three chapters are therefore devoted to models which explicitly utilize one of the leading contenders for our

mechanistic understanding of neural learning: the calcium control hypothesis for synaptic plasticity.

The calcium control hypothesis states that synapses in the brain either increase in strength (potentiate) or decrease in strength (depress) depending on the calcium concentration at the synapse. The fourth chapter of the dissertation uses the calcium control hypothesis to explore how the induction of plasticity at one synapse (called homosynaptic plasticity) can result in plasticity at other synapses, a phenomenon known as *heterosynaptic plasticity*. Using both ball-and-stick and branching dendrite cable models, we explore the hypothesis that heterosynaptic plasticity occurs due to voltage-gated calcium channels (VGCCs) being activated by dendritic depolarization. We propose that VGCC-dependent heterosynaptic plasticity combined with dendritic cable properties can lead to a *hierarchical supervision* effect, where strong synaptic inputs on a dendritic branch can yield heterosynaptic plasticity at its daughter and granddaughter branches. We also explore how clusters at different dendritic locations might interact with each other to create spatiotemporally sensitive synergistic homosynaptic and heterosynaptic effects (Moldwin, Kalmenson, et al., 2023).

The fourth chapter of this dissertation reviews previous mathematical formalizations of the calcium-based plasticity idea and emends them for the sake of mathematical straightforwardness and flexibility. The resulting equations yield a *fixed point - learning rate* framework wherein the calcium concentration at the synapse defines a process which asymptotically increases or decreases the synaptic weight toward a particular strength at a particular rate. We also explain how protein-synthesis dependent late-phase stabilization of synaptic weights fits into this framework (Moldwin, Azran, et al., 2023).

Finally, the fifth chapter of the dissertation discusses the *calcitron*, an M&P-like neuron which uses the calcium control hypothesis (in our new formulation) to implement a wide variety of both supervised and unsupervised learning rules. We show that to implement these different algorithms, neurons need to balance the amount of calcium from global signals (like backpropagating action potentials) with the local calcium influx at each individual synapse (Moldwin et al., 2024).

Introduction

Neuroscientists often think of the brain as a computational system – we receive raw data about our environment via our sensory organs, which then needs to be processed in order build an efficient and useful representation of the world. We also need to be able to recall memories, synthesize new ideas from extant information, control our motor behavior, and so on. In principle, all of these tasks can be formally framed as computational goals, and any process by which the brain accomplishes these tasks can be formalized as an algorithm. One of the primary jobs of the computational neuroscientist is to ascertain what these goals and algorithms are and to connect the algorithmic level description to the physical processes empirically observed to occur in the brain.

Neurons: the fundamental building blocks of the brain

The brain is a difficult object to understand, to put it mildly. It is not obvious how one would even begin to understand the inner workings of this mysterious biological black box. We can start by looking at the most fundamental biological unit within the brain, the brain cell, or neuron. The human brain contains around 86 billion neurons (Azevedo et al., 2009). From anatomical studies, we know that a typical neuron in the mammalian brain might receive tens of thousands of synaptic inputs from axons of other neurons impinging on its dendrites (Eyal et al., 2018) and makes tens of thousands of output connections with other neurons (Wittner et al., 2007). There are many different types of neurons, some of which are localized to specific brain regions, such as the purkinje neurons in the cerebellum.

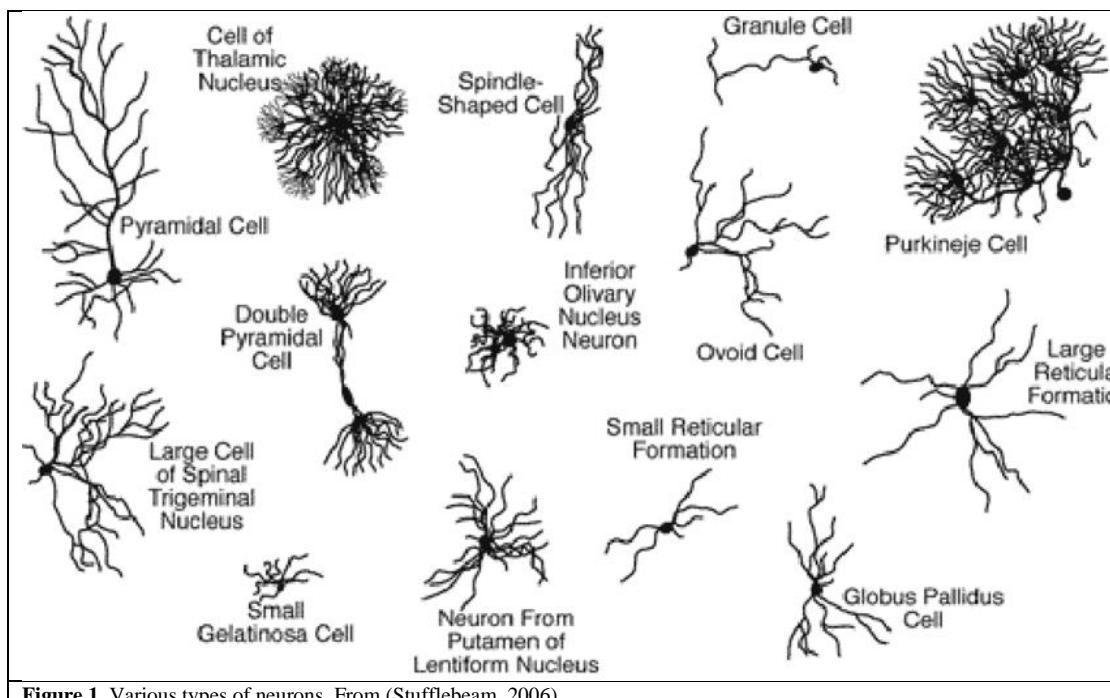


Figure 1. Various types of neurons. From (Stufflebeam, 2006).

A neuron has three basic parts: the *soma* (cell body), the *dendrites*, and an *axon* (Figure 2A). The soma contains the nucleus, which contains the cell's DNA. The dendrites are branches that extend from the neuron that receive input from other neurons. The axon is a long, thin projection from the neuron that sends the neuron's output to other neurons. At the end of the axon is the *axon terminal* which contains

boutons – basically ‘bumps’ at the end of the axon which contain chemicals called *neurotransmitters* (Fig. 2A). The axonal bouton can make contact with the dendrite of another neuron in order to communicate with it. The location at which this occurs is called a *synapse*.

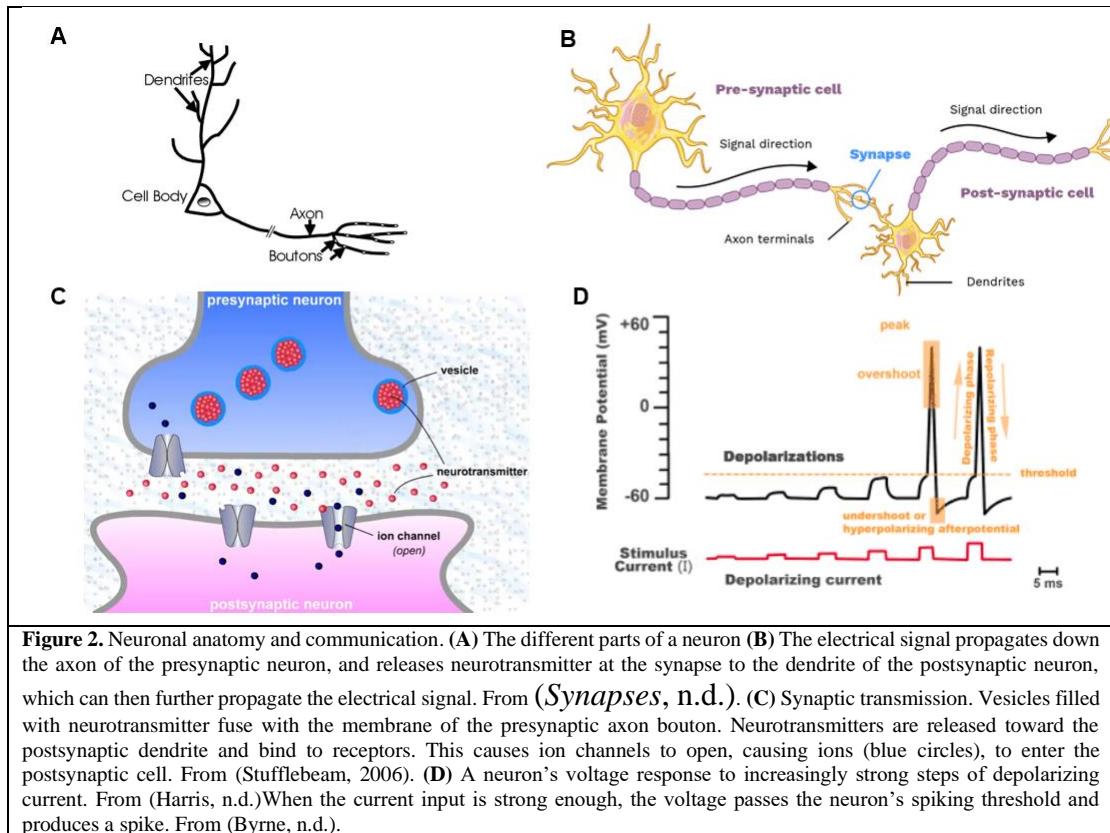
When two neurons communicate, the neuron sending information, called the *presynaptic neuron*, sends an electrical signal called a *spike* or *action potential*, down its axon. This electrical signal travels to the bouton and triggers the release of neurotransmitters from *vesicles*, which are little packages in the bouton that contain the neurotransmitters. The neurotransmitters are released into the synapse and bind to *ion channels* on the neuron receiving the information, otherwise known as the *postsynaptic neuron*.

When neurotransmitters bind to an ion channel, the ion channel opens, allowing a flow of ions into the cell, creating an electrical current. Depending on the type of ion channel that is opened, the sign of the electrical current that flows into the postsynaptic neuron can either be positive (excitatory) or negative (inhibitory). *Glutamate* is a common excitatory neurotransmitter which binds to *AMPA* or *NMDA* receptors, causing positively-charged ions to flow into the postsynaptic neuron. *GABA* is a common inhibitory neurotransmitter which binds to *GABA* receptors and causes negatively-charged ions to flow into the postsynaptic neuron (Fig. 2C).

In the absence of input, a neuron has a *resting potential* of around -60 to -90 mV (millivolts) relative to the outside of the neuron. If excitatory current enters the neuron, the neuron is said to *depolarize*, because the voltage difference between the inside and the outside of the cell becomes smaller. If inhibitory current enters the neuron, the neuron is *hyperpolarized*, because the difference between the inside and the outside of the neuron becomes greater.

Once a neuron is sufficiently depolarized such that its voltage is around -40 mV, the voltage crosses a *spike threshold*. Once this voltage threshold is reached, a large number of *voltage gated ion channels* are simultaneously opened, allowing a huge influx of excitatory current into the neuron, causing the neuron’s voltage to rapidly depolarize above 0 mV (Fig. 2D). This rapid depolarization is the spike (action potential) that we described earlier. The spike is generated in the axon initial segment, near the soma, and travels down the axon to communicate with downstream postsynaptic neurons.

In summary, neurons receive input on their dendrite from other presynaptic neurons. This input is in the form of neurotransmitters, which open ion channels on the postsynaptic neuron, allowing electrical current to flow into the postsynaptic dendrite. If the postsynaptic neuron receives enough excitatory current due to its presynaptic input, it can fire a spike. This spike is then travels down the axon of the postsynaptic neuron and enables the signal to propagate to other neurons. As such, a neuron can be thought of as an input-output device which receives input spikes from presynaptic neurons and produces output spikes as a consequence of receiving sufficient excitatory input.



Thinking about neurons mathematically

If we knew nothing about how neurons worked other than the fact that they receive many inputs which are somehow integrated to produce a single output, we could still write down a generic equation that looks like this:

$$\hat{y} = f(\mathbf{x}, \boldsymbol{\theta}) \quad (1.1)$$

where \hat{y} is the output of the neuron, \mathbf{x} is a vector of the neuron's presynaptic input spikes, $\boldsymbol{\theta}$ is a vector of the neuron's internal parameters (e.g., synaptic weights, locations, etc.), and f is whatever mysterious function the neuron uses to integrate the inputs together with its internal parameters to produce its output.

A couple of points about the above formulation. Firstly, although we have not explicitly mentioned the timing of the presynaptic inputs and output, we can easily modify Eq. (1.1) to temporally integrate information by discretizing time and defining a matrix \mathbf{X} where each row contains the activity history of a particular synapse up until the current time t , in which case we would write:

$$\hat{y}(t) = f(\mathbf{X}, \boldsymbol{\theta}) \quad (1.2)$$

However, for simplicity, this dissertation will mainly assume that we can neglect the timing aspect by binning time into non-overlapping windows and treat inputs as either “basically simultaneous” (in which case the neuron integrates the inputs together as if they were simultaneous) or “basically non-simultaneous” (in which case the neuron doesn’t integrate the inputs together). This is a common assumption used in connectionist models, especially for machine learning contexts. It should be noted,

though, that there are important computational consequences to precise spike timing. (Beniaguev et al., 2021, 2022; Gütig et al., 2016; Gütig & Sompolinsky, 2006, 2009; Memmesheimer et al., 2014).

A second important point about Eq. (1.1) is that we assume that f is a deterministic function of \mathbf{x} and $\boldsymbol{\theta}$. In other words, for a given input vector \mathbf{x} and set of parameters $\boldsymbol{\theta}$, we can reliably predict what the neuron's output will be. Another way of framing this assumption is that *unless we change the neuron's internal parameters, the neuron will always give the same output response to the same input*. This is a useful feature for neurons to have. If neurons responded in a substantially differently manner every time they received the input, it would likely be difficult for the brain to reliably recall memories, produce consistent motor behaviors, and so on.

At the same time, although we want the brain to be reliable, we don't want neurons to *always* produce the same output for the same input, because that precludes the possibility of learning new things. If the brain is to be useful for doing anything that isn't innately programmed into it from birth, there needs to be some process by which a neuron can change its output over time to the same input, i.e. $\hat{y}^t(\mathbf{x}) \neq \hat{y}^{t+1}(\mathbf{x})$, where \hat{y}^t and \hat{y}^{t+1} are the outputs of the same neuron at two subsequent time points. In order to accomplish this, according to Eq. 1.1, it is necessary to modify the internal parameters of the neuron. Formally, we can state this principle as:

$$\hat{y}^t(\mathbf{x}) \neq \hat{y}^{t+1}(\mathbf{x}) \rightarrow \boldsymbol{\theta}^t \neq \boldsymbol{\theta}^{t+1} \quad (1.3)$$

The process of changing a neuron's internal parameters to change its output for a given input is called *plasticity*.

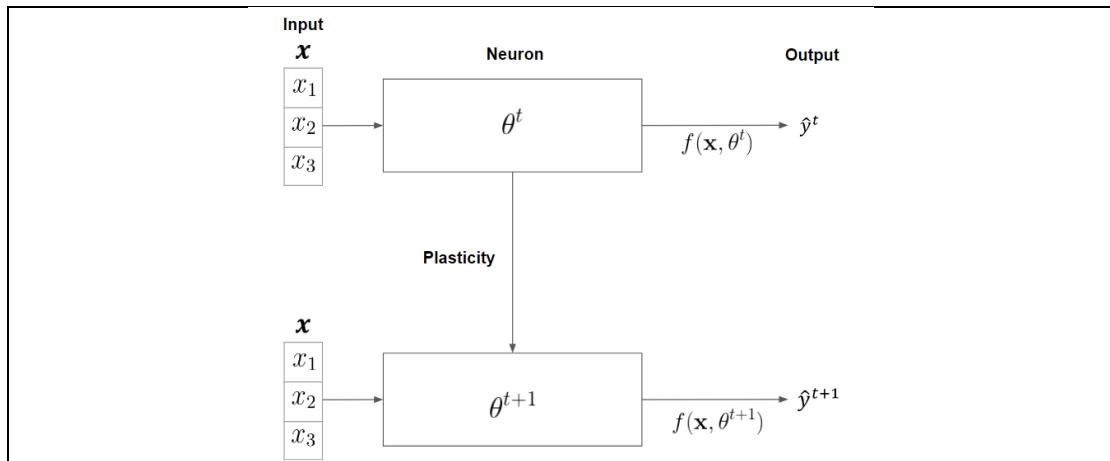


Figure 3. A neuron at time t with parameters $\boldsymbol{\theta}^t$ receives an input vector \mathbf{x} , and produces an output \hat{y}^t by performing some operation f , which is a function of the inputs and internal parameters. Plasticity is the process by which the neuron changes its internal parameters to produce a different output.

Of course, most plastic changes that could conceivably occur in a neuron would not necessarily be particularly useful for anything. In order for plasticity to result in learning storage, we need to have a *learning rule*, which is a (usually iterative) procedure for a neuron to apply plasticity in a smart way such that it gets better over time at some task (like recognizing your grandmother, given that \mathbf{x} is a vector of visual inputs, i.e. pixels). Formally, this usually involves defining a cost or loss function, denoted by L , which is small when the neuron is performing well (i.e. the neuron fires if and only if you see your grandmother) and large otherwise. A useful learning rule is a rule which substantially reduces the loss of the neuron (or the entire

network, but we will focus on the single neuron) over the relevant set of input vectors after iterating the rule for a sufficient number (T) of time steps, i.e.

$$L(f(\{\mathbf{x}\}, \boldsymbol{\theta}^{t=T})) \ll L(f(\{\mathbf{x}\}, \boldsymbol{\theta}^{t=0})). \quad (1.4)$$

Now that we have outlined the concepts of neural computation, plasticity, and learning in a general sense, we will discuss a specific neuron model with its accompanying learning algorithm: the perceptron.

The threshold-linear neuron model

One of the most classic models of the single neuron, used almost exclusively in machine learning applications, is the linear neuron. This neuron is often referred to (including in this dissertation) as a McCulloch and Pitts (M&P) neuron, after the two scientists who first formalized the idea of linear neurons, although their original formulation was somewhat different than the more modern version we will discuss here.

In the linear neuron, each input x_i is multiplied by an associated weight, w_i . If the weighted sum of these inputs exceeds a threshold, T (sometimes called b for bias), the neuron produces a spike ($\hat{y} = 1$). If the weighted sum does not exceed T , the neuron produces no spike (here $\hat{y} = 0$, sometimes the convention $\hat{y} = -1$ is used, in which case the equations below will be slightly different).

$$\hat{y} = H\left(\sum_{i=1}^N w_i x_i - T\right) \quad (2.1)$$

where $H(x)$ is the Heaviside function (other functions are sometimes used instead of the Heaviside function as the “transfer function” for the neural output, such as sigmoid functions, rectified linear functions, and so on, but for parsimony we’ll stick with the Heaviside). If we think back to our generic formulation of a neural input-output function from Eq. 1.1, the set of parameters $\boldsymbol{\theta}$ in a linear neuron are comprised of the synaptic weight vector \mathbf{w} as well as the spike threshold T (although it is sometimes possible to treat T as just another weight, as we will discuss below).

The biological intuition for this model of the neuron is that when a presynaptic axon makes a synapse on a postsynaptic neuron, the presynaptic neuron releases neurotransmitters, which bind to postsynaptic receptors. This causes ion channels to open in the postsynaptic neuron, bringing current into the cell. The amount of current that comes into the cell depends on a number of parameters, such as the number of presynaptic neurotransmitter vesicles, the number of ion channels in the postsynaptic spine, and so on. We can abstract all of these variables into a single value, the weight w_i , which is determines how much current comes into the cell when the presynaptic input x_i is activated (we will often assume x_i is binary, i.e. either on or off, but x_i can also be real-valued and thus indicates the presynaptic firing rate). The current that enters the cell from each synapse depolarizes the neuron – in other words, it brings the neuron’s voltage closer to the spiking threshold T . When the neuron is sufficiently depolarized, a postsynaptic spike is elicited, in line with what we know from early experiments about the relationship between neural depolarization and spiking activity.

A common trick used to simplify equation 2.1 is to treat the spiking threshold T as just another weighted input by defining a new input $x_0 = 1$, and then setting the associated weight $w_0 = -T$. This results in the equivalent formulation:

$$\hat{y} = H\left(\sum_{i=0}^N w_i x_i\right) \quad (2.2)$$

There is subtlety here which will become important later: by treating T as just another weight, we allow plasticity algorithms to modify the spike threshold the same way we modify other synaptic weights. Although it is mathematically convenient, this simplification can create complications when trying to map the linear neuron back to a biological setting. It is sometimes justified by assuming that a constantly active “strong synapse” or aggregate background activity enables a neuron to adjust the difference between its resting potential and the spike threshold to arbitrary values, but this is not a trivial claim. In any event, we will use Eq. 2.2 to continue to explore the linear neuron model.

It is possible to interpret Eq. 2.2 geometrically. When $\sum_{i=0}^N w_i x_i > 0$, the neuron produces a spike, when $\sum_{i=0}^N w_i x_i < 0$, the neuron doesn’t produce a spike. This means that $\sum_{i=0}^N w_i x_i = 0$ is a hyperplane that divides the input space (i.e. the set of possible input vectors \mathbf{x}) into a “spike” region and a “no spike” region. If we change the synaptic weights via some plasticity algorithm, it is possible to change the orientation of this hyperplane, which means that the “spike” and “no spike” regions will also change. This feature is why linear neurons can be considered “linear classifiers”; because they classify the space of possible inputs into the “spike” and “no spike” regions by means of a linear boundary (i.e. a hyperplane) (Fig. 4).

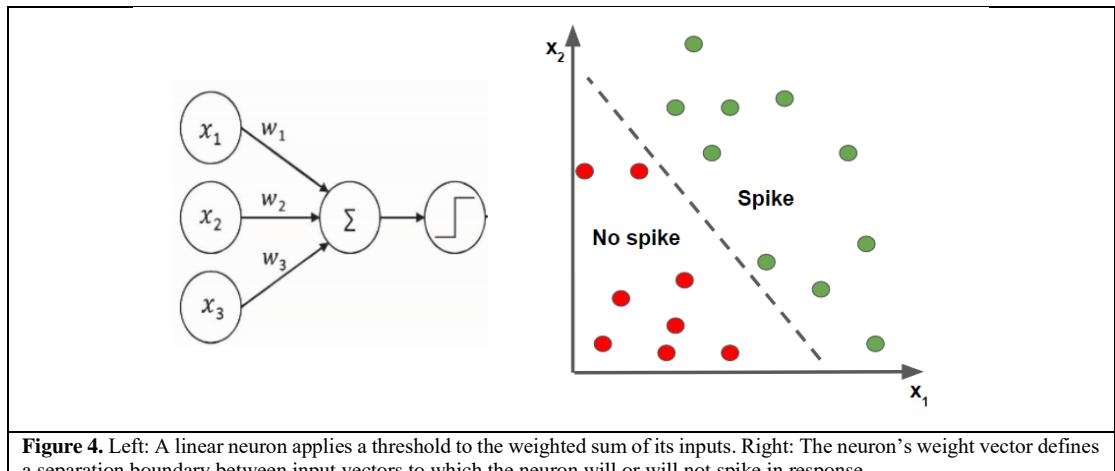


Figure 4. Left: A linear neuron applies a threshold to the weighted sum of its inputs. Right: The neuron’s weight vector defines a separation boundary between input vectors to which the neuron will or will not spike in response.

Supervised classification tasks and the perceptron learning algorithm

Because linear neurons are linear classifiers, we can use them to solve classification problems. Many tasks that the brain faces can be reframed as classification problems. For example, we might want our neuron to be able to distinguish between images of different letters of the alphabet, or decide whether to move a particular muscle in response to a sensory cue.

A common way to solve classification problems is via “supervised learning” – we provide the classification system with “labeled” input patterns that tell the system

which class the input pattern belongs to (in the case of the neuron, whether or not the neuron should spike). Formally, we can say that every training input vector \mathbf{x}^μ is associated with a target label $y^\mu \in \{0,1\}$, where μ is an index of the training examples. (This is why we use the notation of \hat{y} for the neuron's output; the neuron's output be seen as a statistical estimator—often denoted with a pointy hat--of the true label y). If appropriately trained on enough labeled input patterns, the classifier can also generalize to unseen inputs which are sufficiently similar to the training examples of their class such that they lie on the same side of the separation boundary.

The standard way to train a linear neuron is via the perceptron learning algorithm. The basic idea of the learning algorithm is that we sequentially present input patterns \mathbf{x}^μ to the neuron and compare the neuron's output \hat{y}^μ to the target output y^μ . whenever the neuron makes a mistake, we increase or decrease the weights in a manner proportional to the input. The change in the weight vector at each time step is called $\Delta\mathbf{w}$, such that $\mathbf{w}^{t+1} = \mathbf{w}^t + \Delta\mathbf{w}$. If the neuron was supposed to fire for a given input vector \mathbf{x}^μ (i.e. $(y^\mu = 1)$, but it didn't fire ($\hat{y}^\mu = 0$), we increase the weight proportionally to the input, i.e. $\Delta\mathbf{w} = \eta\mathbf{x}$, where η is the “learning rate”, which tells the neuron how aggressively to update. By adding the input vector to the weight vector, we're increasing the strength of the weights associated with that input pattern, making it more likely the neuron will produce a spike the next time it sees that pattern. Conversely, if the neuron *wasn't* supposed to fire in response to \mathbf{x}^μ ($y^\mu = 0$), but it *did* fire ($\hat{y}^\mu = 1$) we *decrease* the weights proportionally to the inputs, i.e. $\Delta\mathbf{w} = -\eta\mathbf{x}$, such that the neuron will be less likely to fire in response to that pattern. If the neuron responded properly ($y^\mu=\hat{y}^\mu$), there's no need to update the weights, so $\Delta\mathbf{w} = 0$. The perceptron learning rule can be represented by the following equation:

$$\mathbf{w}^{t+1} = \mathbf{w}^t + \Delta\mathbf{w} = \mathbf{w}^t + \eta(\hat{y} - y)\mathbf{x} \quad 3.1$$

The rule can also be expressed in a table (Table 1):

	$y = 0$	$y = 1$
$\hat{y} = 0$	0	$\eta\mathbf{x}$
$\hat{y} = 1$	$-\eta\mathbf{x}$	0

Table 1: Weight update Δw in the standard perceptron learning rule.

Eventually, over time, the perceptron learning algorithm will attempt to minimize the number of errors on the training set by appropriately setting the weight vector such that the separating hyperplane will be a border between the input patterns with different labels. However, there are certain classification tasks that are impossible for a perceptron to solve, because they are *linearly inseparable*, which means that no hyperplane can perfectly separate between the input classes (Fig. 5).

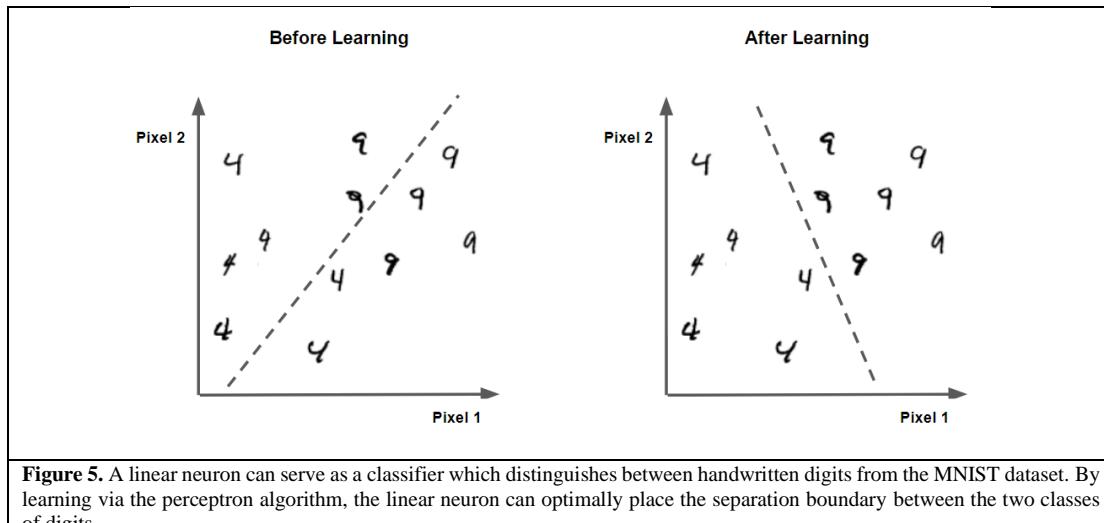


Figure 5. A linear neuron can serve as a classifier which distinguishes between handwritten digits from the MNIST dataset. By learning via the perceptron algorithm, the linear neuron can optimally place the separation boundary between the two classes of digits.

A classic example of a linearly inseparable task is the *exclusive or* (XOR) problem. The XOR function takes two binary bits of input and returns a 1 if they are different and a 0 if they are the same. $XOR([1,0])$ and $XOR([0,1])$ produce a 1, $XOR([1,1])$ and $XOR([0,0])$ produce a 0. If we represent the four binary input vectors geometrically as points on a plane, it is apparent that there is no way to draw a line that will separate the two sets of points (Fig. 6). This is a limitation of linear classifiers which can be overcome by nonlinear classifiers, which are able to draw more elaborate nonlinear boundaries between classes. We will discuss one such classifier in the context of the third chapter in the context of the *gradient clusteron*.

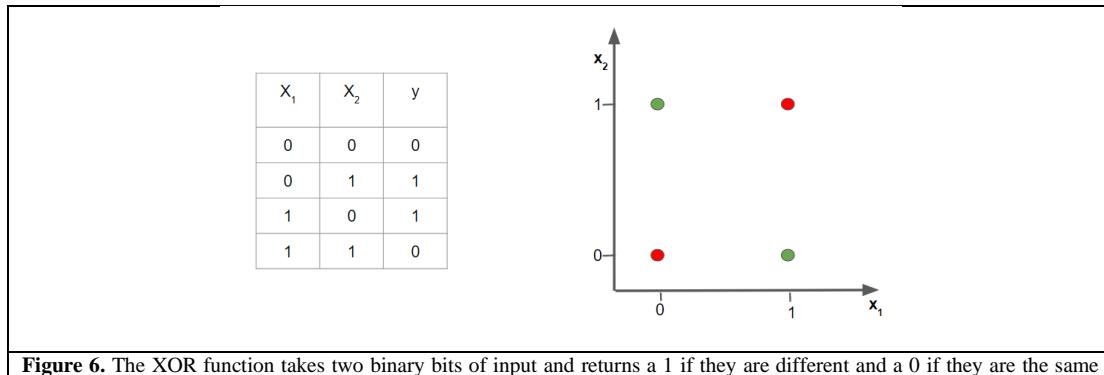


Figure 6. The XOR function takes two binary bits of input and returns a 1 if they are different and a 0 if they are the same (left). When plotted on a plane (right), it is apparent that no linear separation boundary can produce the XOR function.

It is also possible to draw nonlinear separation boundaries by using a network that contains multiple layers of perceptrons, called a *multilayer perceptron* or *deep neural net*. Deep neural nets are a remarkably potent tool for solving a wide variety of tasks, giving rise to what is sometimes referred to as the *deep learning* revolution. This dissertation will focus on the single neuron perspective, but many of the insights discussed here are potentially scalable to the context of deep networks as well.

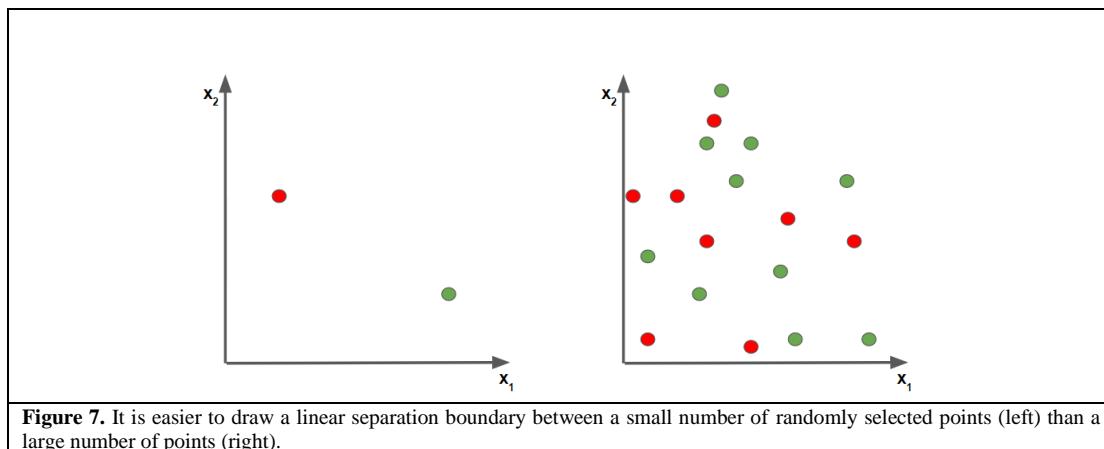
Evaluating classifiers: Generalization and Capacity

It is often useful to know how “good” a particular classification algorithm is. A common method of evaluating classifiers among machine learning practitioners is just to test the classifier’s performance on some benchmark task. A common (easy but canonical) benchmark is the MNIST classification task, where a classifier must learn to discriminate between images of handwritten digits. Typically we are interested in

generalization performance—in other words, the classifier will be trained with a labeled dataset containing, for example, 50,000 images of the digits 0-9, and then following the training phase we present a “test set” to the classifier of unlabeled images. The classifier’s performance on this test set (in terms of correctly identified labels) is used as a metric of how good the classifier is.

An alternative, more mathematically general way to evaluate a classifier is to determine its classification “capacity”. The idea of capacity is that instead of using a particular dataset as a benchmark and evaluating generalization performance, we instead are interested a more theoretical question of “how flexible are the separation boundaries that this classifier can implement?”. There are sometimes analytical ways to answer this question, but often it is easiest to approximate the answer empirically via simulations. This is done by randomly sampling patterns from some probability distribution (for example, by randomly generating vectors \mathbf{x} where each dimension x_i is sampled from a standard normal distribution) and arbitrarily assigning half the input vectors to the positive class and half to the negative class.

As we present more and more of these randomly generated vectors to our classifier, the classifier will have a harder time distinguishing between the two arbitrarily defined classes, because it will have to draw increasingly convoluted separation boundaries to distinguish between the two classes. There will tend to be a “critical capacity”, i.e. a certain number of random patterns beyond which the classifier will begin to lose the ability to correctly classify all patterns. This critical capacity is a function of the number of synapses, i.e. the dimensionality of the input patterns. Comparing the critical capacity between different classifiers (for a given number of synapses) can be an instructive method to compare their relative computational capability. In the case of a standard linear neuron, the capacity is given as $P = 2N$, i.e. the number of patterns is equal to twice the number of synapses. Theoretical work has also been done exploring the capacity of more complex neuron models (Zador & Pearlmutter, 1996).



Important differences between perceptrons and biological neurons

While perceptrons are a convenient abstraction of biological neurons, real biological neurons are incredibly complex organic devices. There are many differences between biological neurons and linear neurons that may have crucial consequences for computation. I will list some of these differences below, especially those that are relevant to this dissertation, although this is by no means an exhaustive list of the ways in which biological neurons differ from linear neurons.

Sign-constrained synapses

In real neurons, usually only a single kind of neurotransmitter is released at the axon terminal. In other words, a synapse is always either excitatory or inhibitory, it cannot “switch signs” via plasticity. This is at odds with the perceptron learning rule, which can change the sign of the synaptic weight from positive to negative or vice versa during learning. Although a modified version of the perceptron rule can be used with signed-constrained weights, this constraint can limit the capacity of a perceptron, see (Amit et al., 1999; Chapeton et al., 2012).

Temporal integration

Perceptrons integrate their information instantaneously. At every step of the perceptron algorithm, an input vector is presented to the neuron and the neuron produces an output depending solely on that input vector and the weight vector at that time. In contrast, real neurons in biology can integrate information temporally over short periods of time. Because the cell membrane in neurons acts as a capacitor, depolarization induced by current injection into the cell will decay exponentially at a rate determined by the membrane time constant τ_m . This gives real neurons a “memory” of inputs that occurred in the recent past. Inspired by this capacitance-based memory, the *Tempotron* is a model neuron with a learning algorithm designed to enable neurons to distinguish between temporal patterns of synaptic input (Gütig & Sompolsky, 2006, 2009). It is also possible for neurons to learn to specify the exact timing of their output spikes as well (Memmesheimer et al., 2014).

Dendritic attenuation and temporal filtering

Perceptrons are “point neurons”; that is, they have no spatially extended dendritic morphology. Biological neurons, in contrast, have elaborate dendritic “trees” on which they receive their synaptic inputs. From neuronal cable theory, we know that the depolarization induced by current injected at a distal location on the dendritic tree (i.e. far away from the soma) will attenuate as it reaches the soma. This means that a synapse placed far away from the soma will contribute significantly less to the peak somatic depolarization relative to the same synapse when placed proximal to the soma (Figure 1A-B) (Rall, 1959, 1964, 1967; Rall & Rinzel, 1973).

In addition to the attenuation of the peak voltage induced by distal synapses, the voltage signal from dendritic inputs becomes progressively more temporally “smeared” the further away the input is from the soma (Figure 1C) (Rall, 1959, 1964, 1967). This temporal smearing effect can be beneficial for recognizing spatiotemporal patterns of input (Beniaguev et al., 2022).

Although much of the interest in dendritic voltage attenuation has emphasized its impact on somatic output, attenuation from one dendritic branch to another potentially has important implications for *heterosynaptic plasticity*, or when plasticity induction at one synapse can also induce plasticity at other synapses. We discuss this at length in the sixth chapter.

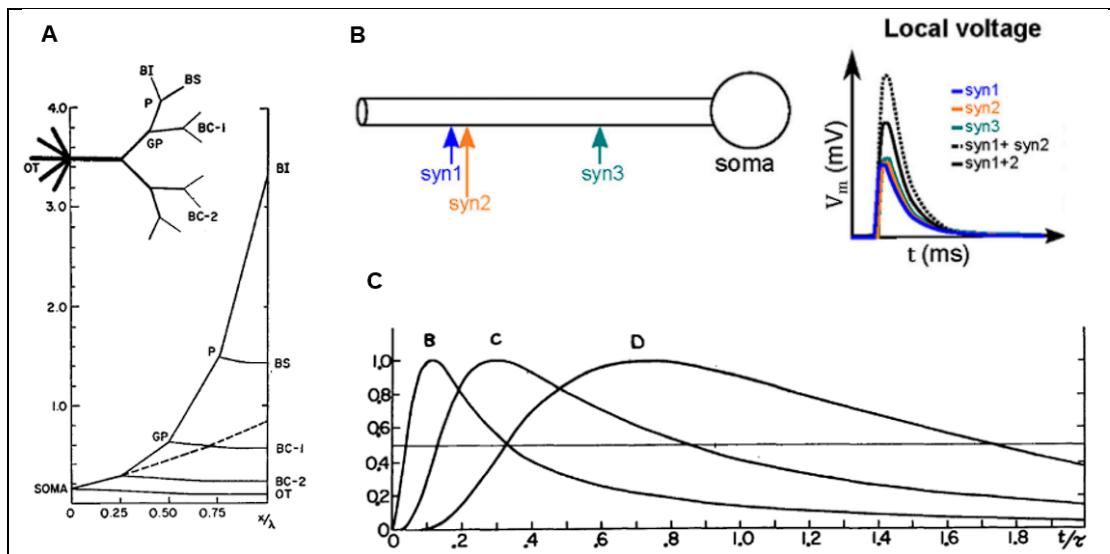


Figure 8. Effect of dendritic cable structure on voltage attenuation and temporal change in synaptic potential. (A): Voltage attenuation for a steady state current input from the input branch (BI) to the soma and other branches in a branching passive neuron model. Note the pronounced asymmetry between the attenuation from the parent branch (P) to the sister branch (BS) relative to the “grandparent” branch (GP). Adapted from (Rall & Rinzel, 1973). (B) Three synapses are placed on a ball and stick cable model. Locally measured depolarization from synapse 1 and synapse 2 sum sublinearly. Adapted from (Tran-Van-Minh et al., 2015). (C): Temporal smearing effect in an unbranched cable model of synaptic inputs that are proximal (‘B’) medial (‘C’) or distal (‘D’) relative to the soma. Voltage traces are normalized to peak voltage. Adapted from (Rall, 1967).

Sublinear integration of conductance-based synapses

Biological synapses are activated when presynaptically-released neurotransmitter binds to a postsynaptic receptor, opening ion channels, which in turn enables ionic current to enter the cell. Ion channels are thus *conductances*, or “doorways” that allow current to enter the neuron. The amount of current I_{syn} that comes through an ionic channel of a given conductance G is given by the equation $I_{syn} = G_{syn}(V - E_{syn})$ where V is the local voltage and E_{syn} is the reversal potential for that conductance. A consequence of this is that as the voltage increases, the difference between the membrane potential and reversal, $V - E_{syn}$, will decrease, bringing in less current with the same amount of conductance. If the cell depolarizes to the reversal potential E_{syn} , even an arbitrarily large conductance would not further depolarize the cell. (As an analogy, if people are trying to enter a room, the room will fill up faster if more doors are opened. But once the room is full, people won’t be able to enter the room irrespective of how many doors you open.)

This means that there is a maximal value, or “cap” to the magnitude of depolarization that a synapse can induce. Normally this is not a major concern, because the reversal potential E is higher than the spike threshold, and it would generally be unnecessary for synapses, even in aggregate, to depolarize the neuron much beyond the spike threshold (at least for slow-spiking neurons, neurons which spike at rates proportional to their synaptic input may be affected more by this).

There are some caveats here, however. If we are dealing with a neuron with elongated dendrites, if input comes to a distally located synapse, the local voltage (at the synapse) might reach the reversal potential, but due to voltage attenuation, the peak voltage observed at the soma might only be a few millivolts. Moreover, if many other synapses are co-active, the *marginal* contribution of each excitatory synapse to the overall depolarization is smaller, because activity from other excitatory synapses bring the voltage closer to the reversal potential. This means that current from multiple

synaptic inputs sum *sublinearly*, such that activated multiple synapses simultaneously will result in a smaller depolarization than one would expect from linearly summing the depolarization obtained when activating those synapses independently. We will discuss how this impacts a biological neuron's classification capacity in the first chapter.

Superlinear active dendritic mechanisms

In addition to the sublinear integration of synaptic conductances described above, synapses can also operate in a *superlinear* fashion, such that activated multiple synapses simultaneously will result in a *larger* depolarization than one would expect from linearly summing the depolarization obtained when activating those synapses independently. This can happen due to *active mechanisms*, i.e. voltage-sensitive ion channels. Voltage sensitivity can enable superlinear integration of synaptic because although a single activated synapse might not be sufficient to activate the voltage-dependent ion channels, multiple activated synapses are.

One active mechanism of particular interest is the NMDA channel, which is both ligand- and voltage-gated, due to magnesium blocking the NMDA channel pore. The NMDA channel can be opened when glutamate binds to the NMDA receptor, but it opens even more (i.e. has a larger conductance) when the dendrite is sufficiently depolarized (Jahr et al., 1990; Jahr & Stevens, 1990).

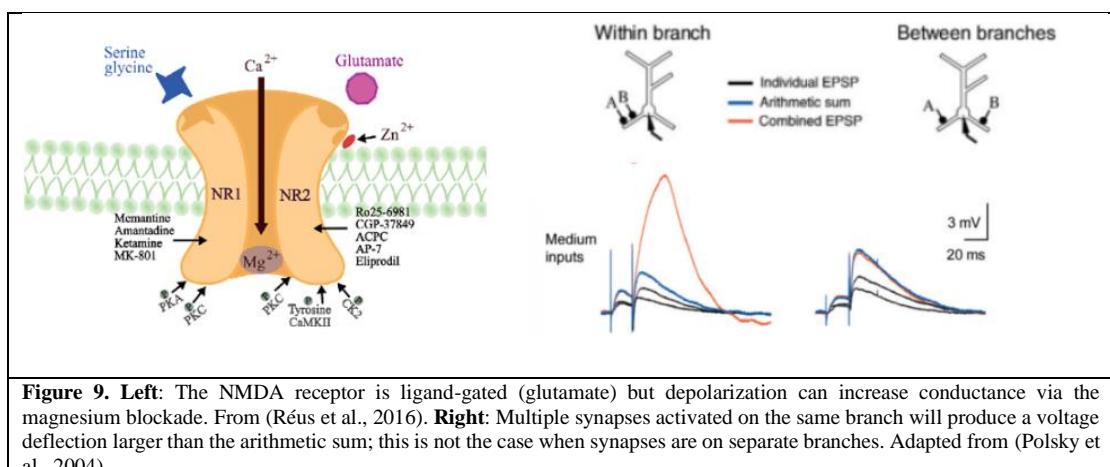


Figure 9. Left: The NMDA receptor is ligand-gated (glutamate) but depolarization can increase conductance via the magnesium blockade. From (Réus et al., 2016). **Right:** Multiple synapses activated on the same branch will produce a voltage deflection larger than the arithmetic sum; this is not the case when synapses are on separate branches. Adapted from (Polsky et al., 2004).

This enables coactivated synapses to interact synergistically if they depolarize the dendrite sufficiently to repel the magnesium block (Polsky et al., 2004a). Because voltage attenuates along the dendrite, it may be ideal for co-activated synapses to be *clustered* on the dendrite in order to activate this superlinear synergistic mechanisms (Mel, 1991). To take advantage of clustering in the context of learning, it is necessary to optimize the *locations* at which synapses are placed via a process known as *structural plasticity*. We explore the computational consequences of structural plasticity in the second chapter in the context of the gradient clusteron.

Biological plasticity protocols and mechanisms

The perceptron learning algorithm (and its multilayer extension, the backpropagation algorithm) was developed to solve classification tasks. Plasticity in biology, however, does not necessarily operate the same way as the perceptron learning rule. From

experimental evidence, we know of a number of different protocols that can induce plastic changes at synapses, such as frequency-dependent plasticity (which depends on the frequency of the pre-synaptic input) and spike timing-dependent plasticity, or STDP (which depends on the relative timing of the pre- and post-synaptic input).

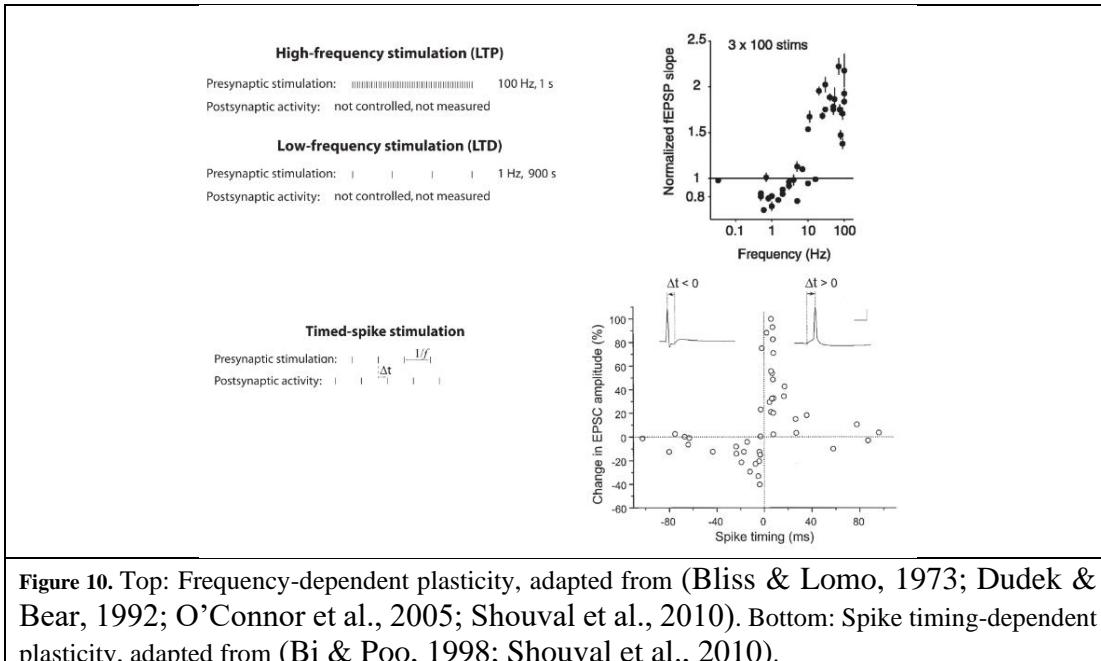


Figure 10. Top: Frequency-dependent plasticity, adapted from (Bliss & Lomo, 1973; Dudek & Bear, 1992; O'Connor et al., 2005; Shouval et al., 2010). Bottom: Spike timing-dependent plasticity, adapted from (Bi & Poo, 1998; Shouval et al., 2010).

An early model, known as the BCM plasticity rule, tried to show how plasticity dependent on the presynaptic and postsynaptic neuronal activity (Bienenstock et al., 1982). Later experimental and modeling work revealed that many plasticity rules are the result of an underlying principle, known as the *calcium control hypothesis*, which states that plastic changes of synaptic strength are determined by the calcium concentration at the synapse (Graupner & Brunel, 2012; J. Lisman, 1989; J. E. Lisman, 2001; J. Lisman & Goldring, 1988; Shouval et al., 2002, 2010). We discuss a novel mathematical formulation (the fixed point – learning rate rule, or FPLR rule) of the calcium control hypothesis in Chapter 4, and we explore computational consequences of the calcium control hypothesis for neural learning in Chapter 5.

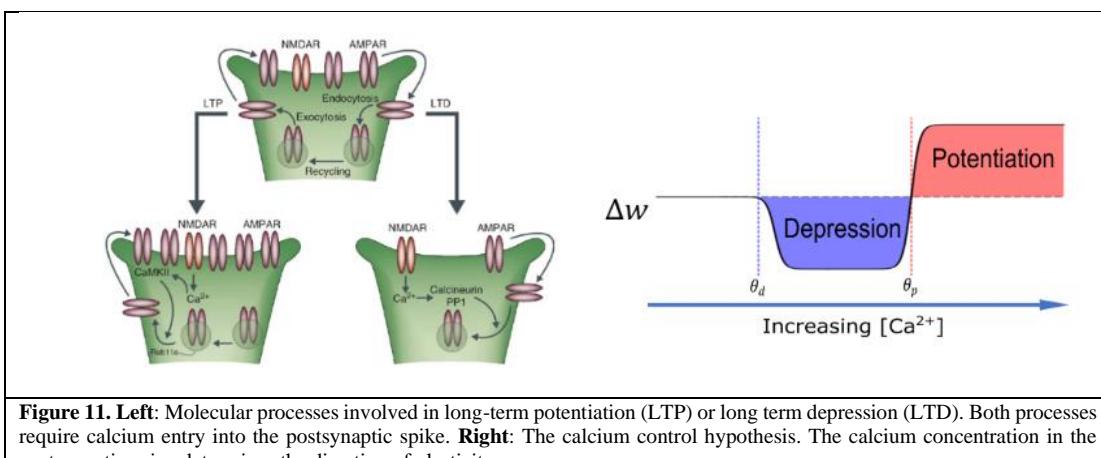


Figure 11. Left: Molecular processes involved in long-term potentiation (LTP) or long term depression (LTD). Both processes require calcium entry into the postsynaptic spike. Right: The calcium control hypothesis. The calcium concentration in the postsynaptic spine determines the direction of plasticity.

Aims of the work

The following chapters, written as independent papers, are intended to address some of the above distinctions between biological neurons and perceptrons. Our main focus will be the question “how might plasticity and learning differ between biological neurons and perceptrons?” Our exploration will use computer simulations and analytical arguments to answer this question.

References

- Amit, D. J., Wong, K. Y. M., & Campbell, C. (1999). Perceptron learning with sign-constrained weights. *Journal of Physics A: Mathematical and General*, 22(12), 2039–2045. <https://doi.org/10.1088/0305-4470/22/12/009>
- Azevedo, F. A. C., Carvalho, L. R. B., Grinberg, L. T., Farfel, J. M., Ferretti, R. E. L., Leite, R. E. P., Filho, W. J., Lent, R., & Herculano-Houzel, S. (2009). Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *The Journal of Comparative Neurology*, 513(5), 532–541. <https://doi.org/10.1002/CNE.21974>
- Beniaguev, D., Segev, I., & London, M. (2021). Single cortical neurons as deep artificial neural networks. *Neuron*, 109(17), 2727-2739.e3. <https://doi.org/10.1016/J.NEURON.2021.07.002>
- Beniaguev, D., Shapira, S., Segev, I., & London, M. (2022). Multiple Synaptic Contacts combined with Dendritic Filtering enhance Spatio-Temporal Pattern Recognition capabilities of Single Neurons. *BioRxiv*, 2022.01.28.478132. <https://doi.org/10.1101/2022.01.28.478132>
- Bi, G., & Poo, M. (1998). Synaptic Modifications in Cultured Hippocampal Neurons: Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type. *Journal of Neuroscience*, 18(24), 10464–10472. <https://doi.org/10.1523/JNEUROSCI.18-24-10464.1998>
- Bienenstock, EL., Cooper, LN., & Munro, PW. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *Journal of Neuroscience*, 2(1), 32–48. <https://doi.org/10.1523/jneurosci.02-01-00032.1982>
- Bliss, T., & Lomo, T. (1973). Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path. *The Journal of Physiology*, 232(2), 331–356. <https://doi.org/10.1113/jphysiol.1973.sp010273>
- Byrne, J. H. (n.d.). *Introduction to Neurons and Neuronal Networks*. Retrieved September 6, 2023, from <https://nba.uth.tmc.edu/neuroscience/m/s1/introduction.html>
- Chapeton, J., Fares, T., LaSota, D., & Stepanyants, A. (2012). Efficient associative memory storage in cortical circuits of inhibitory and excitatory neurons. *Proceedings of the National Academy of Sciences of the United States of America*, 109(51), E3614-22. <https://doi.org/10.1073/pnas.1211467109>

- Dudek, S. M., & Bear, M. F. (1992). Homosynaptic long-term depression in area CA1 of hippocampus and effects of N-methyl-D-aspartate receptor blockade. *Proceedings of the National Academy of Sciences*, 89(10), 4363–4367. <https://doi.org/10.1073/pnas.89.10.4363>
- Eyal, G., Verhoog, M. B., Testa-Silva, G., Deitcher, Y., Benavides-Piccione, R., DeFelipe, J., de Kock, C. P. J., Mansvelder, H. D., & Segev, I. (2018). Human Cortical Pyramidal Neurons: From Spines to Spikes via Models. *Frontiers in Cellular Neuroscience*, 12(June), 1–24. <https://doi.org/10.3389/fncel.2018.00181>
- Graupner, M., & Brunel, N. (2012). Calcium-based plasticity model explains sensitivity of synaptic changes to spike pattern, rate, and dendritic location. *Proceedings of the National Academy of Sciences of the United States of America*, 109(10), 3991–3996. <https://doi.org/10.1073/pnas.1109359109>
- Gütig, R., Gutig, R., Gütig, R., & Gutig, R. (2016). Spiking neurons can discover predictive features by aggregate-label learning. *Science*, 351(6277), aab4113–aab4113. <https://doi.org/10.1126/science.aab4113>
- Gütig, R., & Sompolinsky, H. (2006). The tempotron: a neuron that learns spike timing-based decisions. *Nature Neuroscience*, 9(3), 420–428. <https://doi.org/10.1038/nn1643>
- Gütig, R., & Sompolinsky, H. (2009). Time-Warp–Invariant Neuronal Processing. *PLoS Biology*, 7(7), e1000141. <https://doi.org/10.1371/journal.pbio.1000141>
- Harris, K. (n.d.). Axons. In *Kristen Harris*. Retrieved September 5, 2023, from <https://synapseweb.clm.utexas.edu/axons>
- Jahr, C., & Stevens, C. (1990). A quantitative description of NMDA receptor-channel kinetic behavior. *Journal of Neuroscience*, 10(6), 1830–1837. <https://doi.org/10.1523/JNEUROSCI.10-06-01830.1990>
- Jahr, C., Stevens, C. C. F., Jahr', C. E., Stevens, C. C. F., Jahr, C., & Stevens, C. C. F. (1990). Voltage dependence of NMDA-activated macroscopic conductances predicted by single-channel kinetics. *Journal of Neuroscience*, 10(9), 3178–3182. <https://doi.org/10.1523/JNEUROSCI.10-09-03178.1990>
- Lisman, J. (1989). A mechanism for the Hebb and the anti-Hebb processes underlying learning and memory. *Proceedings of the National Academy of Sciences*, 86(23), 9574–9578. <https://doi.org/10.1073/pnas.86.23.9574>
- Lisman, J. E. (2001). Three Ca²⁺ levels affect plasticity differently: the LTP zone, the LTD zone and no man's land. *The Journal of Physiology*, 532(2), 285–285. <https://doi.org/10.1111/J.1469-7793.2001.0285F.X>
- Lisman, J., & Goldring, M. (1988). Evaluation of a model of long-term memory based on the properties of the Ca²⁺/calmodulin-dependent protein kinase. *Journal de Physiologie*, 83(3), 187–197. <https://europepmc.org/article/med/2856110>
- McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133. <https://doi.org/10.1007/BF02478259>
- Mel, B. W. (1991). The clusteron: Toward a simple abstraction for a complex neuron. *Nips*, 35–42.

- Memmesheimer, R. M., Rubin, R., Ölveczky, B. P., & Sompolinsky, H. (2014). Learning Precisely Timed Spikes. *Neuron*, 82(4), 925–938. <https://doi.org/10.1016/j.neuron.2014.03.026>
- Moldwin, T., Azran, L. S., & Segev, I. (2023). A Generalized Framework for the Calcium Control Hypothesis Describes Weight-Dependent Synaptic Changes in Behavioral Time Scale Plasticity. *BioRxiv*, 2023.07.13.548837. <https://doi.org/10.1101/2023.07.13.548837>
- Moldwin, T., Azran, L. S., & Segev, I. (2024). The Calcitron: A Simple Neuron Model That Implements Many Learning Rules via the Calcium Control Hypothesis. *BioRxiv*, 2024.01.16.575890. <https://doi.org/10.1101/2024.01.16.575890>
- Moldwin, T., Kalmenson, M., & Segev, I. (2021). The gradient clusteron: A model neuron that learns to solve classification tasks via dendritic nonlinearities, structural plasticity, and gradient descent. *PLOS Computational Biology*, 17(5), e1009015. <https://doi.org/10.1371/journal.pcbi.1009015>
- Moldwin, T., Kalmenson, M., & Segev, I. (2022). Asymmetric voltage attenuation in dendrites enables hierarchical heterosynaptic plasticity. *BioRxiv*, 2022.07.07.499166. <https://doi.org/10.1101/2022.07.07.499166>
- Moldwin, T., Kalmenson, M., & Segev, I. (2023). Asymmetric Voltage Attenuation in Dendrites Can Enable Hierarchical Heterosynaptic Plasticity. *ENeuro*, 10(7), ENEURO.0014-23.2023. <https://doi.org/10.1523/ENEURO.0014-23.2023>
- Moldwin, T., & Segev, I. (2018). Perceptron learning and classification in a modeled cortical pyramidal cell. *BioRxiv*, 14(April), 464826. <https://doi.org/10.1101/464826>
- Moldwin, T., & Segev, I. (2020). Perceptron Learning and Classification in a Modeled Cortical Pyramidal Cell. *Frontiers in Computational Neuroscience*, 14(April), 1–13. <https://doi.org/10.3389/fncom.2020.00033>
- O'Connor, D. H., Wittenberg, G. M., & Wang, S. S.-H. (2005). Dissection of Bidirectional Synaptic Plasticity Into Saturable Unidirectional Processes. *Journal of Neurophysiology*, 94(2), 1565–1573. <https://doi.org/10.1152/jn.00047.2005>
- Polksy, A., Mel, B. W., & Schiller, J. (2004). Computational subunits in thin dendrites of pyramidal cells. *Nature Neuroscience* 2004 7:6, 7(6), 621–627. <https://doi.org/10.1038/nn1253>
- Rall, W. (1959). Branching dendritic trees and motoneuron membrane resistivity. *Experimental Neurology*, 1(5), 491–527. [https://doi.org/10.1016/0014-4886\(59\)90046-9](https://doi.org/10.1016/0014-4886(59)90046-9)
- Rall, W. (1964). Theoretical Significance of Dendritic Trees for Neuronal Input-Output Relations (1964). In *Neural Theory and Modeling*. Stanford University Press (pp. 73–79). The MIT Press. <https://doi.org/10.7551/MITPRESS/6743.003.0015>
- Rall, W. (1967). Distinguishing theoretical synaptic potentials computed for different soma-dendritic distributions of synaptic input. *Journal of Neurophysiology*, 30(5), 1138–1168. <https://doi.org/10.1152/jn.1967.30.5.1138>

- Rall, W., & Rinzel, J. (1973). Branch input resistance and steady attenuation for input to one branch of a dendritic neuron model. *Biophysical Journal*, 13(7), 648–688. [https://doi.org/10.1016/S0006-3495\(73\)86014-X](https://doi.org/10.1016/S0006-3495(73)86014-X)
- Réus, G. Z., Abelaira, H. M., Tuon, T., Titus, S. E., Ignácio, Z. M., Rodrigues, A. L. S., & Quevedo, J. (2016). Glutamatergic NMDA Receptor as Therapeutic Target for Depression. *Advances in Protein Chemistry and Structural Biology*, 103, 169–202. <https://doi.org/10.1016/BS.APCSB.2015.10.003>
- Shouval, H. Z., Bear, M. F., & Cooper, L. N. (2002). A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *Proceedings of the National Academy of Sciences*, 99(16), 10831–10836. <https://doi.org/10.1073/pnas.152343099>
- Shouval, H. Z., Wang, S. S. H.-H., & Wittenberg, G. M. (2010). Spike timing dependent plasticity: A consequence of more fundamental learning rules. *Frontiers in Computational Neuroscience*, 4(1), 19. <https://doi.org/10.3389/fncom.2010.00019>
- Stufflebeam, R. (2006). *Neurons, Synapses, Action Potentials, and Neurotransmission*. https://mind.ilstu.edu/curriculum/neurons_intro/neurons_intro.html
- Synapses. (n.d.). Labster Theory. Retrieved September 6, 2023, from <https://theory.labster.com/synapses/>
- Tran-Van-Minh, A., Cazé, R. D., Abrahamsson, T., Cathala, L., Gutkin, B. S., & DiGregorio, D. a. (2015). Contribution of sublinear and supralinear dendritic integration to neuronal computations. *Frontiers in Cellular Neuroscience*, 9(March), 67. <https://doi.org/10.3389/fncel.2015.00067>
- Von Neumann, J. (1958). *The computer & the brain*. 83.
- Wittner, L., Henze, D. A., Záborszky, L., & Buzsáki, G. (2007). Three-dimensional reconstruction of the axon arbor of a CA3 pyramidal cell recorded and filled in vivo. *Brain Structure & Function*, 212(1), 75. <https://doi.org/10.1007/S00429-007-0148-Y>
- Zador, A. M., & Pearlmuter, B. A. (1996). VC dimension of an Integrate and Fire Neuron Model. 8(3), 611–624. <https://doi.org/10.1145/238061.238064>



Perceptron Learning and Classification in a Modeled Cortical Pyramidal Cell

Toviah Moldwin^{1*} and Idan Segev^{1,2}

¹ Edmond and Lily Safra Center for Brain Sciences, The Hebrew University of Jerusalem, Jerusalem, Israel, ² Department of Neurobiology, The Hebrew University of Jerusalem, Jerusalem, Israel

The perceptron learning algorithm and its multiple-layer extension, the backpropagation algorithm, are the foundations of the present-day machine learning revolution. However, these algorithms utilize a highly simplified mathematical abstraction of a neuron; it is not clear to what extent real biophysical neurons with morphologically-extended non-linear dendritic trees and conductance-based synapses can realize perceptron-like learning. Here we implemented the perceptron learning algorithm in a realistic biophysical model of a layer 5 cortical pyramidal cell with a full complement of non-linear dendritic channels. We tested this biophysical perceptron (BP) on a classification task, where it needed to correctly binarily classify 100, 1,000, or 2,000 patterns, and a generalization task, where it was required to discriminate between two “noisy” patterns. We show that the BP performs these tasks with an accuracy comparable to that of the original perceptron, though the classification capacity of the apical tuft is somewhat limited. We concluded that cortical pyramidal neurons can act as powerful classification devices.

OPEN ACCESS

Edited by:

Yilei Zhang,

Nanyang Technological
University, Singapore

Reviewed by:

Salvador Dura-Bernal,

SUNY Downstate Medical Center,
United States

Harel Z. Shouval,

University of Texas Health Science
Center at Houston, United States

*Correspondence:

Toviah Moldwin

toviah.moldwin@mail.huji.ac.il

Received: 24 September 2019

Accepted: 25 March 2020

Published: 24 April 2020

Citation:

Moldwin T and Segev I (2020)

Perceptron Learning and Classification
in a Modeled Cortical Pyramidal Cell.

Front. Comput. Neurosci. 14:33.

doi: 10.3389/fncom.2020.00033

INTRODUCTION

There has been a long-standing debate within the neuroscience community about the existence of “grandmother neurons”—individual cells that code for high-level concepts such as a person’s grandmother. Recent experimental evidence, however, has indicated that there are units that are selective to specific high-level inputs. In particular (Quiroga et al., 2005) found cells in the human medial temporal lobe (MTL) that fire in response to images of a particular celebrity, such as Jennifer Aniston or Halle Berry. One remarkable aspect of this finding is that different images of the same celebrity would elicit a response in these neurons even if the subject of the image was facing a different direction, wearing different clothes, or under different lighting conditions. In other words, the specificity of these MTL cells is invariant to certain transformations of the sensory stimulus. Regardless of whether this finding is evidence for grandmother cells or merely for sparse coding (Quiroga et al., 2008), it is apparent that individual neurons can be highly selective for a particular pattern of sensory input and also possess a certain level of generalization ability, or “tolerance,” to differences in the input that do not change the essence of the sensory scene.

From a physiological standpoint, achieving a high degree of accuracy on a recognition task is a daunting challenge for a single neuron. To put this in concrete terms, a pyramidal neuron may receive around 30,000 excitatory synapses (Megías et al., 2001). As a first approximation, at any given moment, each of this neuron’s presynaptic inputs can either be active or inactive,

yielding $2^{30,000}$ possible binary patterns. If the presynaptic inputs contain information about low-level sensory stimuli (such as pixels or orientation filters) and the postsynaptic neuron needs to respond only to images of Jennifer Aniston, for example, there must be some physiological decision procedure by which the neuron “chooses” which of those $2^{30,000}$ patterns are sufficiently close to the binary representation of Jennifer Aniston to warrant firing a spike as output.

There are several ways that a neuron can selectively respond to different input patterns. The most well-known method is to adjust synaptic “weights” such that only input patterns which activate a sufficient number of highly-weighted synapses will cause the cell to fire. It is this principle which serves as the basis of the perceptron learning rule (Rosenblatt, 1958) which is, in turn, the foundation for the artificial neural networks (ANNs) that are commonly used today in machine learning and deep networks (Rumelhart et al., 1986; Krizhevsky et al., 2012).

The *perceptron* is a learning algorithm that utilizes a mathematical abstraction of a neuron which applies a threshold activation function to the weighted sum of its input (**Figure 1A**). This abstraction is known as the McCulloch and Pitts (M&P) neuron (McCulloch and Pitts, 1943). The non-linear output of the neuron plays the role of a classifier by producing a positive output (a spike, +1) in response to some input patterns and a negative output (no spike, -1) in response to other patterns. The perceptron is trained in a supervised manner wherein it receives training patterns which are labeled as belonging to either the positive or the negative category. The perceptron output is calculated for each pattern, and if the perceptron output for a particular pattern does not match the label, the perceptron’s weights are updated such that its output will be closer to the correct output for that example in the future.

While the remarkable efficacy of networks of M&P neurons has demonstrated for various learning tasks, few attempts have been made to replicate the perceptron learning algorithm in a detailed biophysical neuron model with a full morphology and active dendrites with conductance-based synapses. It thus remains to be determined whether real cells in the brain, with all their biological complexity, can integrate and classify their inputs in a perceptron-like manner.

In this study, we used the perceptron learning algorithm to teach a detailed realistic biophysical model of a layer 5 pyramidal cell with a wide variety of active dendritic channels (Hay et al., 2011) to solve two kinds of classification problems: a classification task, where the neuron must correctly classify (by either spiking or not) a predefined set of “positive” and “negative” input patterns, and a generalization task, in which the neuron has to discriminate between two patterns that are corrupted by noise in the form of bit flips (i.e., where active synaptic inputs are switched to inactive and vice versa). We explored the ability of real neurons with extended non-linear dendritic trees and conductance-based excitatory synapses to perform classification tasks of the sort commonly solved by artificial neurons (see section Discussion for a treatment of why only excitatory synapses were used). We found that the performance of the biophysical perceptron (BP) is close to that of its artificial M&P counterpart.

RESULTS

Classification Task

To implement the perceptron learning algorithm in a modeled layer 5 thick tufted pyramidal cell (L5PC) we distributed excitatory conductance-based AMPA and NMDA synapses on the detailed model developed by Hay et al. (2011). We created input patterns consisting of 1,000 excitatory synapses, 200 of which were active in any given pattern. We varied the total number of patterns (P) presented to the modeled neuron in order to determine its classification capacity (**Figure 1B**). We tested conditions of $P = 100$, $P = 1,000$, and $P = 2,000$. These binary patterns were evenly divided into a “positive” (+1) group (for which the modeled neuron should produce at least one spike) and a “negative” (-1) group (for which the modeled neuron should not produce a spike). To achieve perfect accuracy, the neuron would have to correctly fire in response to all the patterns in the positive group and not fire in response to all the patterns in the negative group. Note that, initially, there is no reason for the neuron to perform at better than chance level, because all the patterns contain the same number of active synapses.

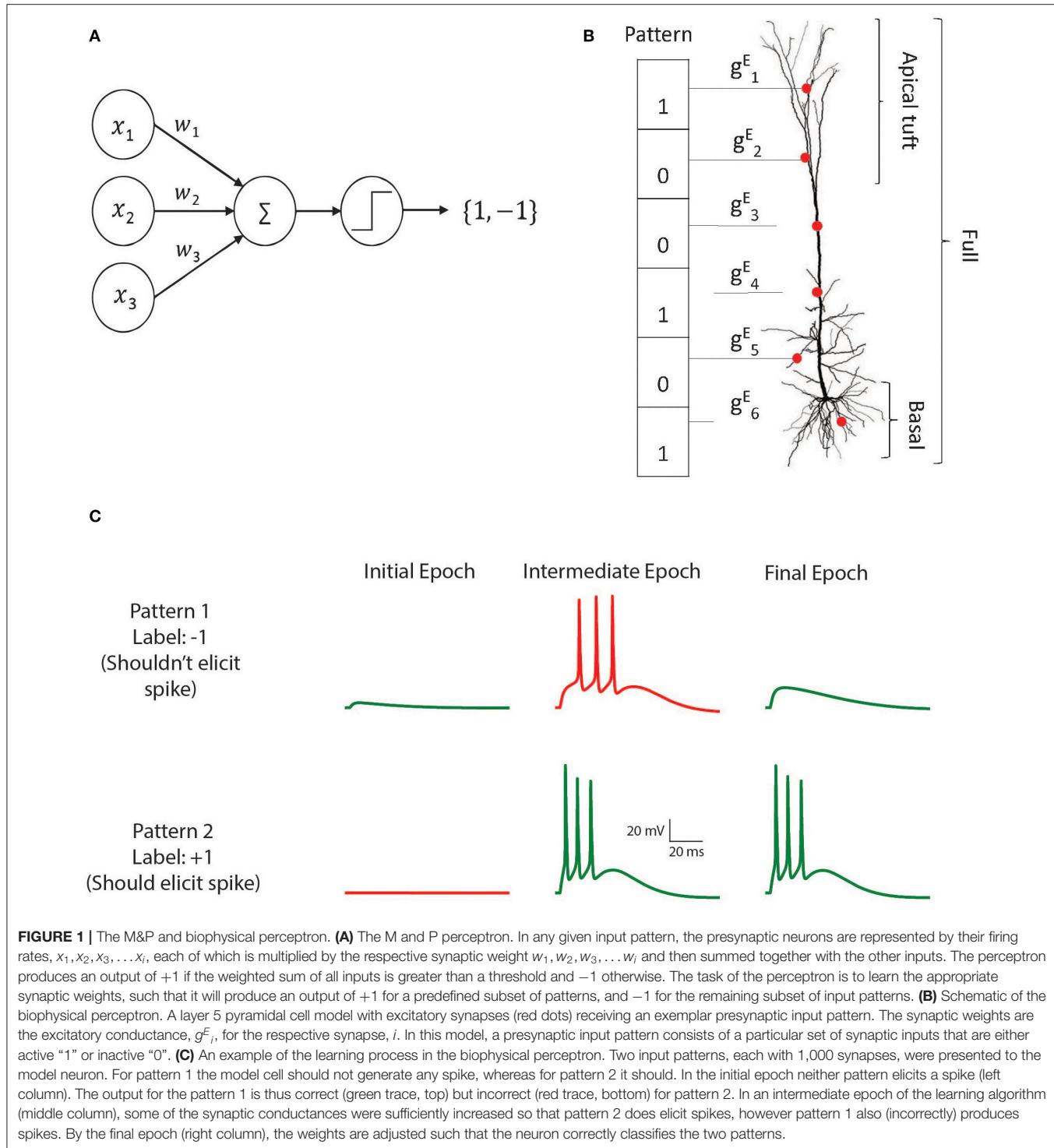
We then used the perceptron learning algorithm (see section Materials and Methods) to modify the synaptic weights such that the cell could correctly classify all the patterns (**Figure 1C**). This procedure was repeated in conditions in which synapses were placed over the whole dendritic tree, only on the apical tuft, only on the basal tree, or only on the soma in order to determine how the location of the synapses affects the cell’s ability to classify patterns using the perceptron learning rule (see section Discussion for the biological significance of input patterns on different parts of the dendritic tree). We also tested the algorithm with current-based synapses rather than of conductance-based synapses, to examine whether conductance-based synapses have any advantages or disadvantages with respect to the cell’s performance as a classifier.

Figure 2 shows the learning curves (**Figure 2A**) and classification accuracy (**Figure 2B**) for each of the above conditions. In all cases the cell is able to improve its performance relative to chance, indicating that the complexity of biophysical cells does not preclude perceptron learning despite the fact that the learning algorithm was devised for a much simpler abstraction of a cell.

We compared the classification accuracy for each condition in the biophysical model to an equivalent M&P perceptron with excitatory weights (see section Materials and Methods). When all synapses are placed on the soma or the proximal basal tree of the biophysical perceptron, the classification accuracy of the biophysical perceptron is near to that of the M&P perceptron.

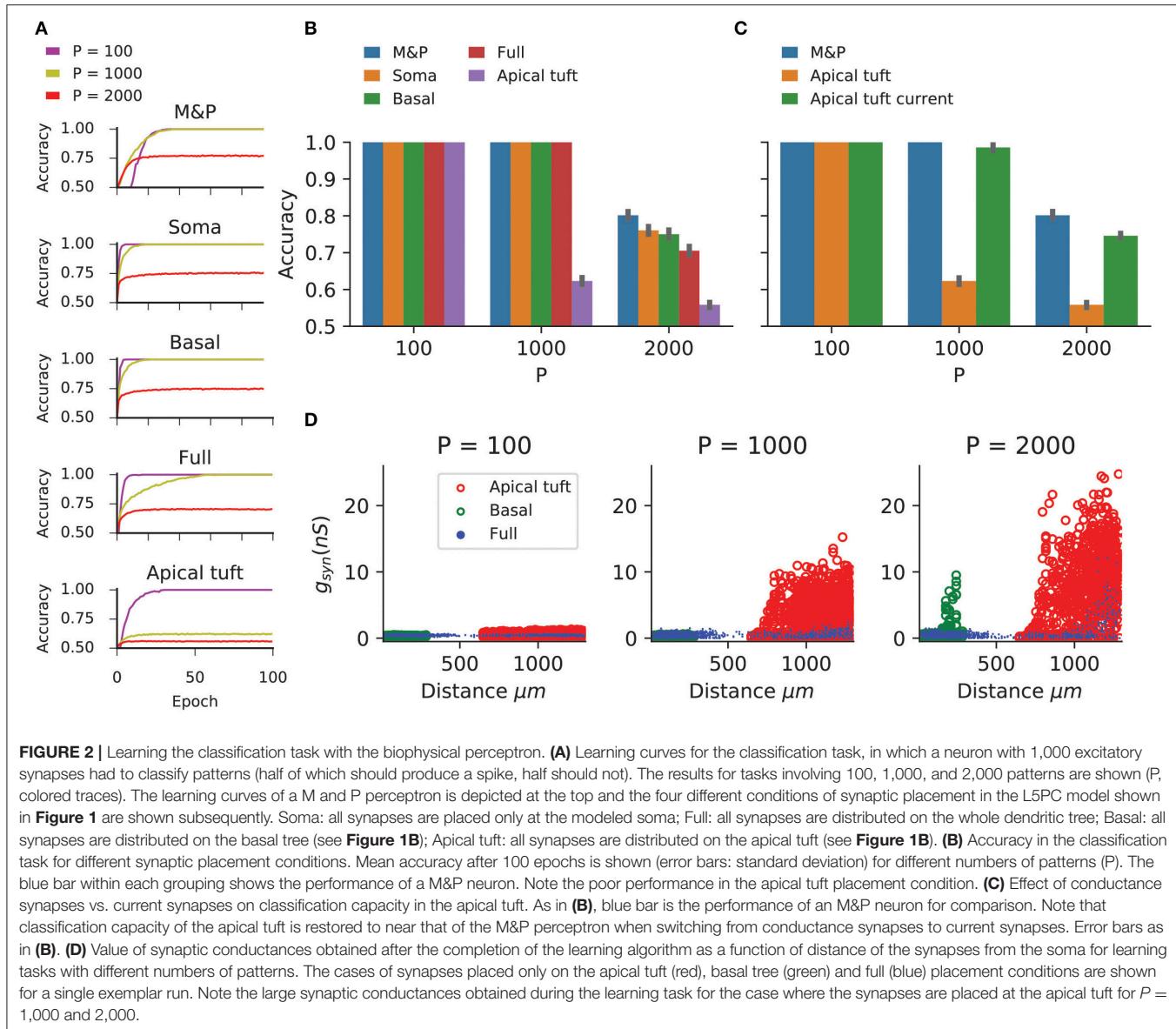
As expected from the theoretical literature (Chapeton et al., 2012), the accuracy in each condition decreases with the number of patterns that the neuron must learn. This can be seen in **Figure 2B**, where the classification accuracy degrades in each condition as we move from $P = 100$ to $P = 1,000$ and from $P = 1,000$ to $P = 2,000$.

In all synaptic placement conditions, the M&P perceptron and the BP performed with perfect accuracy on the “easy” task with $P = 100$. In conditions where the synapses were placed



only on the soma or only on the basal tree, the performance of the BP is comparable to that of the M&P neuron for $P = 1,000$ (M&P: 100%, basal: 100%, soma: 100%) and for $P = 2,000$ (M&P: 77%, basal: 75%, soma: 76%). In the condition where synapses were placed uniformly over the full tree, the discrepancies were somewhat larger for $P = 2,000$ (M&P: 77%, full: 70.5%).

However, when the synapses are all placed on the apical tuft of the biophysical cell, the classification accuracy of the biophysical perceptron decreases dramatically, even in the presence of supra-linear boosting mechanisms such as NMDA receptors and active Ca^{2+} membrane ion channels. For $P = 1,000$, the M&P neuron achieves 100% classification accuracy, whereas if



the synapses are all placed on the apical tuft, the neuron only achieves 62% accuracy. In the condition with $P = 2,000$, the M&P neuron achieves 77% classification accuracy whereas the BP achieves only 55.8% classification accuracy, barely better than chance level. However, by switching from conductance-based synapses to current-based synapses in the apical tuft condition, it was possible to regain almost all of the “loss” in the classification accuracy (In the $P = 1,000$ condition, from 62% with conductance synapses to 98.5% with current synapses, in the $P = 2,000$ condition, from 55.8% with conductance synapses to 74.5% with current synapses) (**Figure 2C**).

We argue that the reason for the discrepancy in classification accuracy for the biophysical perceptron between the conditions wherein synapses are placed on the apical tuft, as opposed to the soma or basal dendrites, is due to the passive filtering properties of the neuronal cable and the saturation effect of conductance

synapses. Specifically, the attenuation of voltage along the length of cable from apical tuft dendrites to the spike initiation zone means that the *effective weight* of that synapse—namely the magnitude of the resultant somatic EPSP—is greatly reduced. This phenomenon has been observed previously (Rall, 1967; Stuart and Spruston, 1998), but it has been argued (Häusser, 2001; Rumsey and Abbott, 2006) that the cell might be able to overcome this drop in voltage by simply increasing the strength (i.e., conductance) of distal synapses. We demonstrated, however, that this is not the case. We show (**Figure 2D**) that the perceptron learning algorithm will, on its own, increase the weights of apical tuft synapses far beyond the biologically plausible range of 0.2–1.3 nS (Sarid et al., 2007; Eyal et al., 2018) in attempting to correctly classify all the patterns. Still, the classification accuracy of the apical tuft biophysical perceptron remains quite poor [see, however Gidon and Segev (2009)]

who show that the opposite phenomena will occur with a standard STDP rule, resulting in smaller synaptic conductances for distal synapses].

We claim that “democratization” via disproportionately increasing distal synaptic conductances does not solve the classification accuracy problem for synapses located on the apical tuft because effective synaptic weights are bounded by the synaptic reversal potential in the distal dendrites, even if one were to increase synaptic conductances to arbitrarily high values. As such, the maximal effective synaptic weight (MESW)—defined as the peak somatic EPSP voltage when a given dendritic location approaches the synaptic reversal potential (**Figure 3A**)—is equivalent to the synaptic driving force multiplied by the attenuation factor from that dendritic location to the soma. (Note: This is true in the passive case, dendritic non-linearities can affect the MESW values. Our calculations of MESWs in this study are based on simulations of the model with all non-linearities present, as shown **Figure 3A**). The MESWs for distal synapses are thus smaller than those for proximal synapses (**Figure 3B**).

Importantly, the marginal effect of each synapse in the presence of background dendritic activity (as in our case, where we activated 200 synapses simultaneously) differs from the MESW (measured when the synapse acts in isolation). For example, a single synapse brought to its reversal potential can interact supralinearly with other synapses via activating NMDA-conductance, strengthening the effect of the other synapses (Polsky et al., 2004). Alternatively, if there is a substantial amount of background activity in the dendrite, the voltage in the dendrite near the location of a given synapse may already be close to the synaptic reversal potential. In this case, the marginal effect of activating that synapse, even with an arbitrarily large conductance, can be substantially below its MESW (**Figure 3C**). Indeed, when the apical tuft is sufficiently active to bring the soma near the spike threshold, this sublinear interaction between synapses dominates, and the background-adjusted maximum effective weight of each synapse (bgMESW) tends to be below the MESW of that synapse activated in isolation (**Figure 3D**, see section Materials and Methods).

From the standpoint of learning theory, the “cap” on the effective weights of distal apical synapses restricts the parameter space of the biophysical perceptron, reducing its capacity. When a perceptron learns to classify between two sets of patterns, it creates a linear separation boundary—i.e., a hyperplane—which separates the patterns in an N -dimensional space, where N is the number of synaptic inputs in each pattern. The separation boundary learned by the perceptron is defined by the hyperplane orthogonal to the vector comprising the perceptron’s weights. When the weights of the perceptron are unconstrained, the perceptron can implement any possible hyperplane in the N -dimensional space. However, when the weights are constrained—for example by the MESWs of the apical tuft of L5PCs—the perceptron can no longer learn every conceivable linear separation boundary, reducing the ability of the perceptron to discriminate between large numbers of patterns [Note: because we use only excitatory synapses, the weight space in all synaptic placement conditions is already substantially constrained to positive values even before imposing MESWs, see

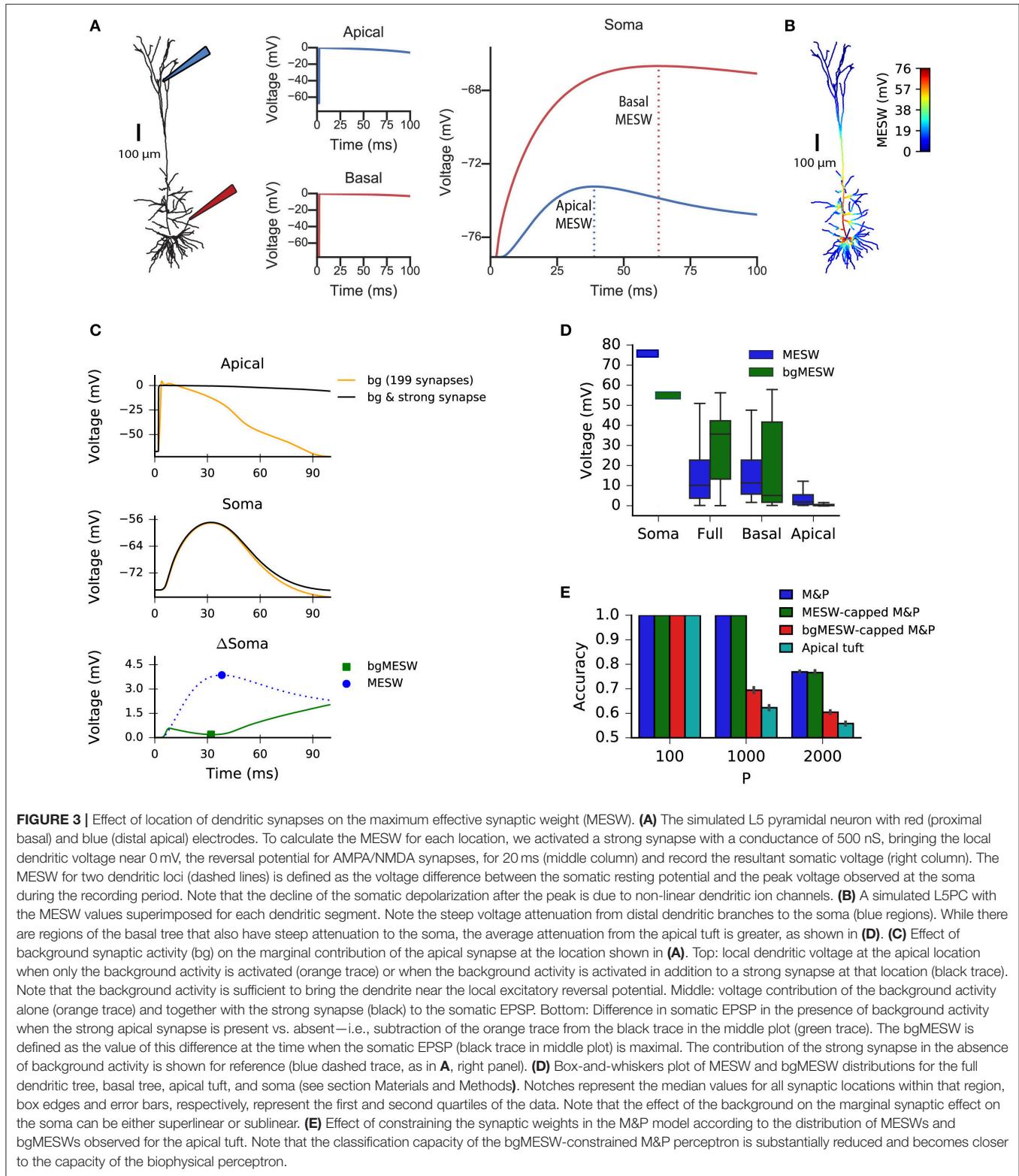
Chapeton et al. (2012) for a full treatment]. To demonstrate this effect, we calculated the MESW for each synapse in the apical tuft and then imposed this distribution of MESWs onto an M&P perceptron (see section Materials and Methods). Interestingly, the MESW caps on the synaptic weights of the M&P neuron did not hamper its classification performance on our task. However, when we used caps based on the marginal effect of each synapse in the presence of the other synaptic activity in the patterns—the bgMESWs—the weight-capped M&P perceptron produced a reduced classification capacity in a manner similar to the biophysical perceptron when synapses were restricted to the apical tuft (**Figure 3E**).

The fact that switching the apical synapses from conductance-based to current-based substantially improves classification accuracy supports the notion that voltage saturation due to synaptic reversal potential is responsible for the reduced performance of the apical tuft synapses (**Figure 2C**). It should be emphasized that the limited capacity of the apical tuft is *not* because apical synapses cannot induce the neuron to fire, as the neuron with only apical synapses performs with perfect accuracy when it only needs to classify 100 patterns, indicating that 200 active synapses on the apical tuft are fully capable of generating a somatic spike. It is thus evident that the reduced classification capacity of the apical patterns is due to the restriction of the weight space needed to properly discriminate between positive and negative patterns, not because the apical tuft input is insufficiently strong to create a somatic spike.

Generalization Task

To explore whether the apical tuft is always at a disadvantage when it comes to pattern classification, we also tested the biophysical perceptron on a generalization task. Instead of classifying a large set of fixed patterns, in the generalization task the neuron was presented with “noisy” patterns drawn from one of two underlying fixed patterns. In this task, noise was added to the underlying pattern by performing “bit flips,” i.e., flipping an active synapse to an inactive synapse or vice versa (**Figure 4A**). We tested both the biophysical perceptron (with different synaptic placement conditions, as in the classification task) and the positive-weighted M&P neuron on their ability to classify these noisy patterns in conditions with varying levels of difficulty, as determined by the number of bit flips. The goal of the task was that the neuron should fire in response to noisy patterns generated by the first underlying pattern, but not fire in response to noisy patterns generated by the second underlying pattern (**Figure 4B**).

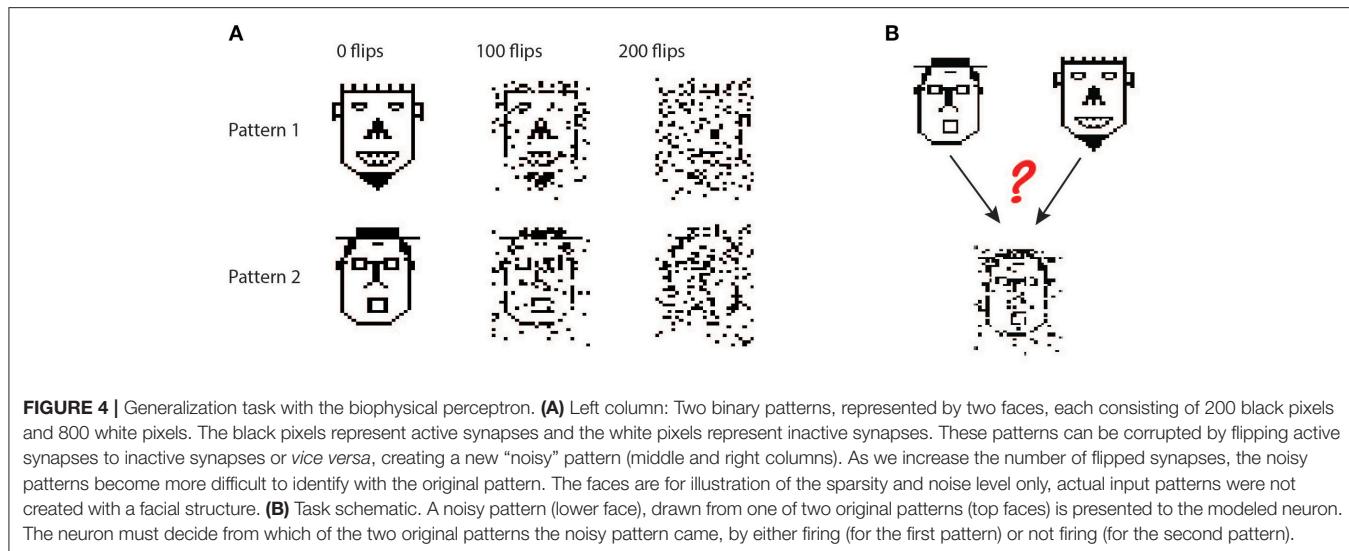
In this task, we observe that in all conditions the BP performs similarly to the M&P perceptron. We do not observe any substantial diminution in classification performance between the apical tuft and the soma, as we do in the classification task (**Figures 5A,B**). In the condition with 100 bit flips, the difference in accuracy between the apical tree and the soma were small (M&P: 85% soma: 85%, apical tuft: 81.8%). The same is true for the more difficult task with 200 bit flips (M&P: 72%, soma: 71.8%, apical tuft: 67.4%). Changing the conductance synapses to current synapses did not substantially affect these results (**Figure 5C**). Moreover, capping the weights of the M&P neuron with the bgMESWs from the apical tuft, as



we did in the classification task, did not considerably worsen the M&P perceptron's performance (**Figure 5D**). We also note that, while some of the synaptic weights of the apical tuft did increase beyond the biological range during learning in the

biophysical perceptron (**Figure 5E**), the effect is much smaller in the classification task (**Figure 2D**).

The discrepancy between the apical tuft and soma may be smaller in the generalization task than in the classification task



because the difficulty in the classification task is fundamentally about finding the correct hyperplane that will separate between the two classes of patterns. As we increase the number of patterns in each of the classes, we require more flexibility in the weight space of the neuron to ensure that all the positive and negative patterns end up on opposite sides of the separating hyperplane. This flexibility is impeded by the bgMESWs of the apical tuft. By contrast, the generalization problem only contains two canonical “patterns.” The difficulty in learning the generalization task with a large amount of noise (in terms of bit flips) does not stem from the challenge of precisely defining a separation boundary. Rather, solving the generalization task is hard because, even if we had an optimal separation boundary, the noise in the input entails that some of the noisy patterns would still necessarily be misclassified.

MATERIALS AND METHODS

Cell Model

We utilized a detailed biophysical model of a cortical layer 5b thick-tufted rat pyramidal cell written in NEURON with a Python wrapper (Carnevale and Hines, 1997; Hines et al., 2009). The parameters of the model, which includes numerous active mechanisms, are described in Hay et al. (2011). The mechanisms used in this model were: Im, Ca_LVAst, Ca_HVA, CaDynamics_E2, SKv3_1, SK_E2, K_Tst, K_Pst, Nap_Et2, NaTa_t, NaTs2_t. The model had 108 apical compartments and 83 basal compartments.

Excitatory synapses were AMPA/NMDA-based synapses as in Muller and Reimann (2011) with a dual-exponential conductance profile with a voltage-dependent magnesium gate (Jahr and Stevens, 1990; Rhodes, 2006) with a gate constant of 0.08. The AMPA conductance had a rise time (τ) of 0.2 ms and a decay time of 1.7 ms while the NMDA conductance had a rise time of 0.29 ms and a decay time of 43 ms. The synaptic depression and facilitation parameters set to 0. The NMDA:AMPA conductance ratio was set to 1.6:1. In both the classification and generalization experiments, we placed all 1,000 synapses in each pattern either

on the soma, basal tree, or apical tuft according to a uniform spatial distribution.

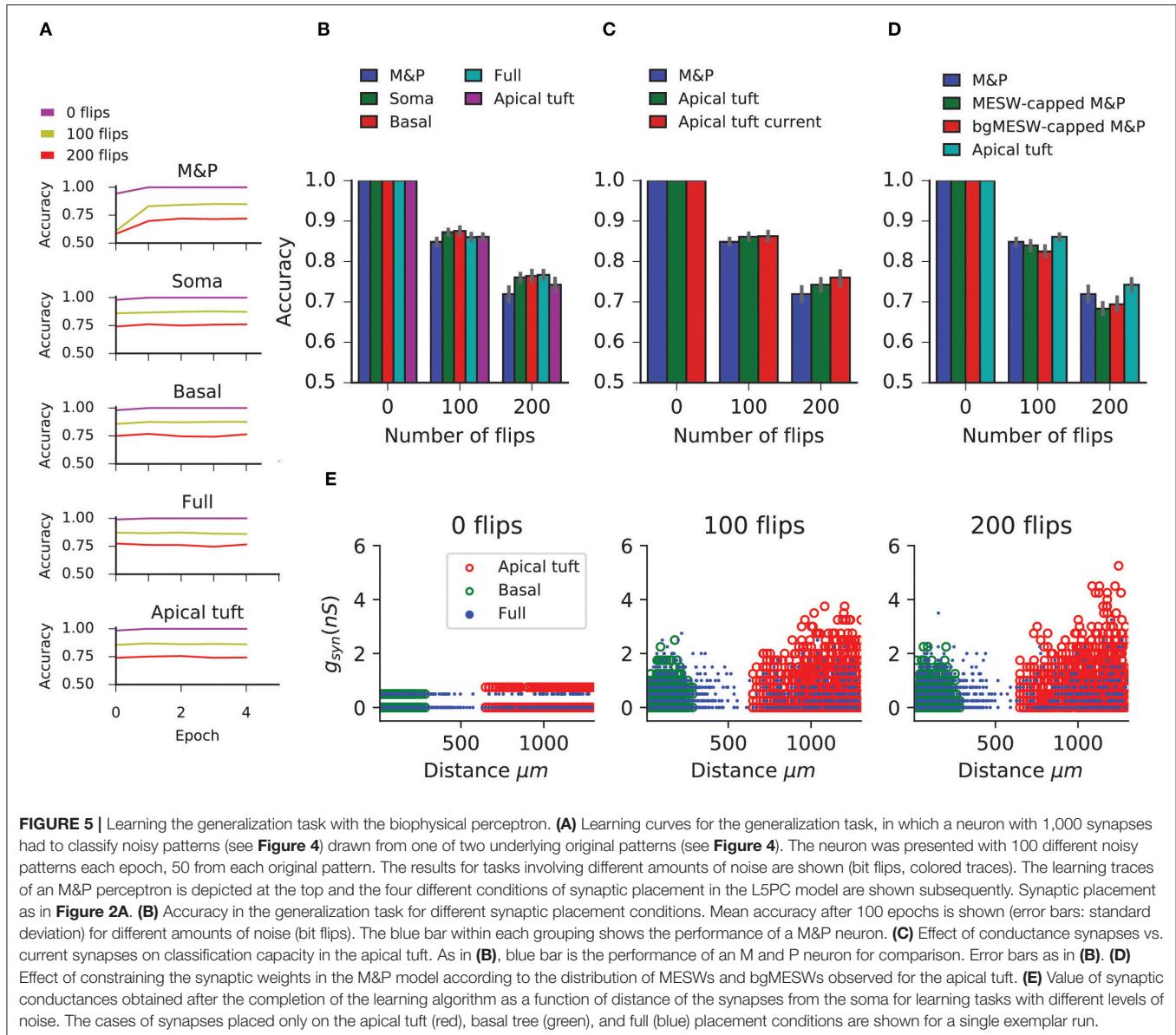
Classification Task

For the classification task, each of the P patterns was generated by randomly choosing 200 out of the 1,000 synapses to be activated. The patterns were then randomly assigned to either the positive or negative class. Patterns were presented to the cell by simultaneously stimulating the 200 active synapses with a single presynaptic spike at the beginning of the simulation. Simulations of the neuron were run with a Δt of 0.1 ms for a total of 100 ms. Patterns were considered to have been classified as “positive” if they produced at least one spike within the 100 ms time window and as “negative” if no spikes occurred.

The choice of 200 active synapses was to simulate a regime of high cortical activity. The maximal firing rate for excitatory cortical neurons is estimated to be around 20 Hz (Heimel et al., 2005; Hengen et al., 2013). Assuming the maximum firing rate per excitatory synapse, a pyramidal cell with 10,000 excitatory synapses would receive 200,000 synaptic inputs/sec or 200 inputs/ms.

We utilized an “online” version of the perceptron learning algorithm, applying the plasticity rule every time a pattern was presented to the neuron. Also, because we limited our analysis to excitatory synapses, we use the modified algorithm proposed in Amit et al. (1999) for sign-constrained synapses, which ensures that synaptic weights never become negative.

The algorithm works as follows: A presynaptic input pattern x is presented to the neuron, where x is a vector consisting of 1,000 binary inputs, each of which is labeled x_i and associated with a particular synapse on the dendritic tree with synaptic weight w_i (for conductance synapses, this is the excitatory conductance of the synapse, g_i^E). Each pattern has a target value, $y_0 \in \{1, -1\}$ associated with it, where 1 means “should spike” and -1 means “shouldn’t spike.” When the pattern is presented to the neuron via simultaneous activation of all the synapses in the pattern, the soma of the neuron will produce a voltage response. If that



voltage response contains at least one spike within 100 ms, we set the output variable $y = 1$. If the voltage response does not contain any spikes, we set $y = -1$. For each presynaptic input pattern, the plasticity rule for synapse i to update its weight w_i at time is defined as:

$$w_i \leftarrow \max(0, w_i + \eta d w_i) \quad (1)$$

where $d w_i$ is defined as:

$$d w_i = \begin{cases} 0, & y = y_0 \\ y_0 x_i, & y \neq y_0 \end{cases} \quad (2)$$

and η is the learning rate.

In other words, if the target output is the same as the actual output of the neuron, we do nothing. If the target is “should spike” and the neuron does not spike, we increase the weight of all synaptic inputs that were active in the pattern. If the target is “shouldn’t spike” and the neuron does spike, we decrease the synaptic weights of all synaptic inputs that were active in the pattern, unless that would decrease the synaptic weight below 0, in which case we reduced the weight of that synapse to 0.

The accuracy of the neuron’s output was calculated after each epoch, which consisted of a full pass of presenting each pattern (in random order) to the neuron. To ensure that accuracy improved on every epoch and reached a reasonable asymptote for all conditions, we set the learning rate η to 0.002 for the condition with AMPA/NMDA conductance synapses and an active tree, and a rate of 0.19 for the condition with current synapses. We

also used the “momentum” technique (Rumelhart et al., 1986) to improve learning speed. The average simulation time for a complete run of the learning algorithm for the classification task (i.e., 100 epochs) was several hours to 2–3 days depending on the task (more patterns required more simulation time). Results shown in **Figures 2A–D, 3D** are averaged over 10 runs of the classification task.

M&P Model (Not Constrained by Synaptic Battery)

To compare the BP to an equivalent M&P perceptron (**Figures 2A,B, 5A,B**) we used a M&P perceptron with only excitatory weights as described in Amit et al. (1999) (See Equations (1) and (2) above). A M&P neuron with no inputs would have a “bias” input value of -77.13 to mimic the resting potential of the BP and a “spiking threshold” of -53.1 to mimic the voltage spiking threshold of the biophysical neuron. The learning algorithm used a learning rate η of 0.0008 which was dynamically modified in the learning algorithm via the momentum technique (Rumelhart et al., 1986). Initial learning rates were hand-tuned according to the criteria that the learning curves should monotonically increase and reach an asymptote (verified by visual inspection). We also tested smaller learning rates than the ones listed in the M&P model to see if accuracy could be improved via smaller learning steps; however the improvement in classification obtained by doing so was negligible and thus was deemed not worth the additional computational time in the biophysical model.

MESW Calculation and MESW-Constrained M&P Model

To calculate the MESWs for the L5PC model, we added a very strong synapse (500 nS) to each dendritic segment in the neuron model bringing the segment within 2.5 mV of the synaptic reversal potential of 0 mV. The MESW for a dendritic segment is defined as the difference between the somatic resting potential and the peak depolarization obtained at the soma within 100 ms after synaptic activation (**Figure 3A**).

To create an MESW-constrained M&P model for the apical tuft, we calculated the distribution of MESWs per unit length of the dendritic membrane in the apical tuft. The median and quartile values of the MESWs for all synaptic placement conditions are shown in the box-and-whisker plot in **Figure 3D**. We then created an M&P neuron where each weight was individually given a “cap” drawn randomly from the apical tuft MESW probability distribution which would prevent the weight of that input from increasing above a certain value. In other words, if the plasticity algorithm (Equation 1) would bring w_i to be greater than cap, c_i , we would “freeze” the weight at c_i . Formally, this means that the plasticity rule in the case of an error for the MESW-capped neuron is

$$w_i \leftarrow \max(0, \min(c_i, w_i + \eta dw_i)) \quad (3)$$

where η and dw_i are as defined above in Equation (2).

bgMESW Calculation and bgMESW-Constrained M&P Model

To calculate the bgMESWs for the L5PC model, we distributed 199 “background” synapses on the neuron according to a uniform distribution per unit length of the dendritic membrane. All background synapses had the same conductance. To find the synaptic conductance required to bring the neuron near its spiking threshold, we gradually increased the synaptic conductances of all synapses by 0.05 nS steps until the neuron produced at least one spike. The largest conductance that didn’t cause the neuron to spike was used as the conductance for the near-threshold background activity. The conductances for each distribution condition were: Soma: 0.3 nS, Basal: 0.3 nS, Apical tuft: 0.52 nS, Full: 0.48 nS (values are averaged over 10 trials of this procedure to account for the randomness in the placement of the background synapses). In the presence of this background activity, we added a strong synapse to each dendritic location, as detailed in the section for the MESW calculation. To find the marginal contribution of a single strong input at each location, we subtracted the somatic EPSP obtained via the background activity from the somatic EPSP obtained when both the background activity and the strong synapse at that location are active, creating a difference curve (**Figure 3C**). The bgMESW is defined as the value of this difference curve at the time when the somatic EPSP is maximal in the condition when both the strong synapse and background activity are active.

Generalization Task

In the second task (generalization), we created two underlying patterns of $1,000$ synapses each, where 200 synapses were active, as in the classification task. These patterns were then corrupted by flipping a given number synapses (0 , 100 , or 200 , depending on the condition) and presented to the neuron. To maintain the sparsity of the patterns, half of the flipped synapses were switched from active to inactive and the other half switched from inactive to active. For example, in the condition with 100 flipped bits, 50 out of the 200 previously active synaptic inputs were flipped to inactive, and 50 out of the 800 previously inactive synaptic inputs were switched to active.

In every epoch of the learning task, we presented the neuron with 50 noisy patterns generated by the first underlying pattern and 50 noisy patterns generated by the second underlying pattern for a total of 100 patterns per epoch (the order of the presentation of patterns from the two underlying patterns was also randomized). We set the learning rate η to 0.25 for the condition with AMPA/NMDA conductance synapses and an active tree, and a rate of 10 for the condition with current synapses. Learning rates were hand-tuned as described above. Similar to the classification task, we used the online perceptron learning rule with the momentum modifier. In this task we only ran the algorithm for 5 epochs, as this was enough for the learning to achieve a plateau. Results shown in **Figures 5A–D** are averaged over 20 repetitions of the generalization task.

Simulations

Simulations were all performed using Neuron v.7.6 (Carnevale and Hines, 1997; Hines et al., 2009) running on a multi-core

cluster computer with 3,140 logical cores; the number of available cores varied depending on other jobs being run on the cluster. Each trial of the learning algorithm was run on a separate core; neither the learning or the cell simulation was multithreaded. The average simulation time for a complete run of the learning algorithm for the generalization task (i.e., 5 epochs) was several minutes.

DISCUSSION

In the simulations described above, we have demonstrated that the perceptron learning algorithm can indeed be implemented in a detailed biophysical model of L5 pyramidal cell with conductance-based synapses and active dendrites. This is despite the fact that the perceptron learning algorithm traditionally assumes a cell which integrates its inputs linearly, which is not the case for detailed biophysical neurons with a variety of non-linear active and passive properties and conductance-based synapses. That being said, the ability of a biophysical perceptron to distinguish between different patterns of excitatory synaptic input does depend on the location of the relevant synapses. Specifically, if all the synapses are located proximally to the soma, such as on the proximal basal tree, the cell has a classification capacity similar to that of the M&P perceptron. However, for activation patterns consisting of more distal synaptic inputs, such as those on the apical tuft, the classification capacity of the BP is reduced. We showed that this is due to the reduced effectiveness of distal synapses due to cable filtering and synaptic saturation in the presence of other synaptic inputs, which limits the parameter space of the learning algorithm and thus hampers classification capacity. We also demonstrated that the diminished classification capacity in the apical tuft is negligible in a generalization task. This indicates that, while the maximum effective synaptic weights of the apical tuft may be somewhat limiting for its classification capacity, they do not hamper the apical tuft's robustness to noise.

The above discussion considers that the pyramidal cell separately classifies inputs that synapse onto different regions of its dendrites (such as the apical tuft and the basal tree) and that it does not simultaneously integrate all the synaptic input impinging on the cell. This decision was motivated by a growing body of evidence that different parts of the dendritic tree may play separate roles in shaping the neuron's output. From anatomical studies, it is known that axons from different brain regions preferentially synapse onto particular regions of layer 5 pyramidal cells. For example, basal dendrites tend to receive local inputs whereas the apical tuft receives long-range cortical inputs (Crick and Asanuma, 1986; Budd, 1998; Spratling, 2002; Spruston, 2008). This has led to theories of neuronal integration for layer 5 pyramidal cells that involve a "bottom-up" stream of information entering the basal dendrites and "top-down" signals coming to the apical tuft (Siegel et al., 2000; Larkum, 2013; Manita et al., 2015). Moreover, it has recently been shown experimentally that when experiencing somatosensory stimulation, layer 5 pyramidal cells in S1 first exhibit an increase in firing rate corresponding to the bottom-up sensory input (ostensibly to the basal tree), and then, 30 ms later, receive

top-down input to the apical tuft from M2 (Manita et al., 2015). This indicates the presence of temporally segregated time windows in which the cell separately integrates input from the apical and basal tree. There is also work suggesting that plasticity rules may function differently in different regions of the cell (Gordon et al., 2006), again indicating that different regions of the cell might serve as input regions to distinct information pathways, and, as such, may have different priorities underlying the decision of when the cell will or will not fire. Taken together, the above studies strongly suggest that the apical tuft and basal dendrites can and should be studied as independent integration units.

Inhibition

Our study made several simplifications to the learning and plasticity processes found in biology. Critically, our plasticity algorithm utilized only excitatory synapses and did not consider the effect of inhibition on learning. This is not because we believe that inhibition does not play a role in learning; on the contrary, inhibitory synapses are essential both for the learning process and in defining the input-output function of the cell (Wulff et al., 2009; Kullmann et al., 2012; Müllner et al., 2015). However, by restricting ourselves to excitatory synapses, we were able to isolate important biophysical properties of excitatory synapses—namely the impact of synaptic saturation (the bgMESWs) that might have been masked in the presence of inhibition. Future work on the "biophysical perceptron" will include the role of inhibitory synapses; in this case special care must be taken to understand how inhibitory inputs interact with excitatory inputs on different locations of the cell (Gidon and Segev, 2012; Doron et al., 2017). The addition of synaptic inhibition has the potential to increase the classification capacity of the cell (Chapeton et al., 2012), and localized inhibition may allow for additional forms of compartmentalized computation at the dendritic level.

Relationship to Models of Learning in Purkinje Cells

The focus on excitatory synapses also enables our work to be directly compared to studies of excitatory perceptron-like learning done on Purkinje cells—which have been classically conceived of as perceptrons (Marr, 1969; Albus, 1971)—, such as the work of Brunel et al. (2004), Steuber et al. (2007), and Safaryan et al. (2017). These studies demonstrated that detailed models of Purkinje cells can learn to discriminate between different patterns of input from the parallel fibers (PF) via a perceptron-like usage of long-term depression (LTD), which is known to occur in PF-Purkinje synapses. Crucially, the difference between the Purkinje cell's responses to learned vs. unlearned patterns was the duration of the pause between spikes in the Purkinje cell's output subsequent to the presentation of PF input. Steuber et al. (2007) argue that this pause duration-based learning depends on the modulation of calcium concentrations inside the cell. This is different from the more direct M&P-like mechanism, used in the present study, of synapses being weighted such that only certain input patterns will reach the cell's spiking threshold.

Non-linearities and Alternative Plasticity Rules

Our focus on perceptron-like learning constitutes an additional simplification, as perceptron learning ignores how dendritic non-linearities such as local NMDA spikes (Schiller et al., 2000; Polsky et al., 2004), dendritic Na^+ spikes (Golding and Spruston, 1998; Sun et al., 2014), and dendritic Ca^{2+} spikes (Magee and Johnston, 1995; Kampa et al., 2006; Cichon and Gan, 2015) may impact learning in classification tasks. Although a variety of dendritic non-linearities are present in our L5 pyramidal cell model, we did not make explicit use of them in our plasticity rule. Indeed, some models of dendritic integration such as the Clusteron (Mel, 1991, 1992) and the two-layer model (Poirazi and Mel, 2001) treat the NMDA spike as critical for dendritic computation. In particular, these models treat clustering of nearby synapses, and “structural plasticity,” or the relocation of synaptic inputs within and between branches as crucial for learning (Trachtenberg et al., 2002; Larkum and Nevian, 2008; Losonczy et al., 2008; Kastellakis et al., 2015; Weber et al., 2016; Mel et al., 2017). The present study did not address the role of synaptic clustering in learning; a promising future direction would be to combine the weight-based learning rules used in our study with the structural plasticity algorithm as discussed in Mel (1992).

There are several other models of learning and plasticity that make use of neuronal biophysics and constitute promising opportunities for improving the learning ability of pyramidal cell models in a biologically plausible way. The calcium-based plasticity rule of Graupner and Brunel (2012) presents an exciting possibility for implementing perceptron-like learning in a more biological manner by making direct use of the experimentally observed mechanisms of plasticity in neurons. Because neurons exhibit some properties of multi-layered networks (Poirazi et al., 2003; Beniaguev et al., 2019), it would also be valuable to explore more powerful learning algorithms that make use of the dendrites as a second (or higher) layer of computation as in Schiess et al. (2016). Alternatively, it may make sense to consider a different paradigm of dendritic learning, where the dendrites attempt to “predict” the somatic output, allowing for forms of both supervised and unsupervised learning (Urbanczik and Senn, 2014). Variants of the perceptron rule, such as the three-threshold learning rule (Alemi et al., 2015) may also be valuable to explore the ability of biophysical cells to solve pattern-completion tasks.

Timing

Another crucial element that remains to be studied in detailed biophysical models is the role of the timing of both the input and output of pyramidal cells in learning and computation. Regarding input timing, some theoretical work has been done on the M&P perceptron, which has been extended in a variety of ways to take into account several components of real neurons. One such extension is the tempotron, which uses a leaky integrate and fire mechanism (Gütig and Sompolinsky, 2006) and can make use of conductance-based synapses (Gütig and Sompolinsky, 2009) to classify spatiotemporal input

patterns. Regarding output timing and firing rate, learning rules like the one from Gütig (2016) can learn to solve the temporal credit-assignment by producing different spike rates for different inputs. Similarly, the Chronotron (Florian, 2012) considers learning rules that generate precisely timed output spikes. It is not clear to what extent these particular plasticity algorithms are truly “biological,” but there is no question that temporal sequence learning is an essential feature of the brain (Aslin et al., 1998; Xu et al., 2012; Moldwin et al., 2017). The addition of a temporal dimension increases the classification capacity of the cell, as discussed in Gütig and Sompolinsky (2009).

Broader Relevance

The present study shows that, by implementing the perceptron learning rule, layer 5 cortical pyramidal cells are powerful learning and generalization units, comparable—at the very least—to the abstract M&P perceptron. Other plasticity rules, which take into account synaptic clustering, input and output timing, and interaction between the apical and basal regions of pyramidal cells will be explored in further studies in detailed biophysical models in order to determine their biological plausibility and classification capacity. Until then, our study should be viewed as a baseline for comparison of any future work implementing learning algorithms in detailed biophysical models of neurons.

DATA AVAILABILITY STATEMENT

The code used for the biophysical model (including the hoc files for the pyramidal cell model) and the M&P model, as well as the code used to generate the input patterns, can be found at <https://github.com/tmoldwin/BiophysicalPerceptron>.

AUTHOR CONTRIBUTIONS

TM and IS designed the research. TM implemented the simulation, analyzed the results, and created the figures. IS supervised the research and contributed to the development of the theoretical and biophysical aspects of the study.

FUNDING

This work was supported by the Drahi family foundation, the EU Horizon 2020 program (720270, Human Brain Project), a grant from Huawei Technologies Co., Ltd., and by a grant from the Gatsby Charitable Foundation, the ETH domain for the Blue Brain Project (BBP), and from the NIH Grant Agreement U01MH114812.

ACKNOWLEDGMENTS

We would like to thank Oren Amsalem for his assistance in many aspects of this work, particularly his help with the use of the NEURON software. We also would like to thank David

Beniaguev, Guy Eyal, and Michael Doron for their insightful discussions about machine learning and neuronal biophysics. Itamar Landau provided several useful comments to an early version of this work, inspiring important revisions. Additionally,

we appreciate Nizar Abed's support in maintaining the computer systems used to perform our simulations and data analysis. This manuscript has been released as a pre-print at BioRxiv (Moldwin and Segev, 2018).

REFERENCES

- Albus, J. S. (1971). A theory of cerebellar function. *Math. Biosci.* 10, 25–61.
- Alemi, A., Baldassi, C., Brunel, N., and Zecchina, R. (2015). A three-threshold learning rule approaches the maximal capacity of recurrent neural networks. *PLoS Comput. Biol.* 11, 1–23. doi: 10.1371/journal.pcbi.1004439
- Amit, D. J., Wong, K. Y. M., and Campbell, C. (1999). Perceptron learning with sign-constrained weights. *J. Phys. A* 22, 2039–2045. doi: 10.1088/0305-4470/22/12/009
- Aslin, R. N., Saffran, J. R., and Newport, E. L. (1998). Computation of conditional probability statistics by 8-month-old. *Infants.* 9, 321–324.
- Beniaguev, D., Segev, I., and London, M. (2019). Single cortical neurons as deep artificial neural networks. *BioRxiv* 613141. doi: 10.1101/613141
- Brunel, N., Hakim, V., Isobe, P., Nadal, J.-P., and Barbour, B. (2004). Optimal information storage and the distribution of synaptic weights: perceptron versus Purkinje cell. *Neuron* 43, 745–757. doi: 10.1016/j.neuron.2004.08.023
- Budd, J. M. L. (1998). Extrastriate feedback to primary visual cortex in primates: a quantitative analysis of connectivity. *Proc. R. Soc. B Biol. Sci.* 265, 1037–1044. doi: 10.1098/rspb.1998.0396
- Carnevale, N. T., and Hines, M. L. (1997). The neuron simulation environment. *Neural Comput.* 1209, 1–26.
- Chapeton, J., Fares, T., Lasota, D., and Stepanyants, A. (2012). Efficient associative memory storage in cortical circuits of inhibitory and excitatory neurons. *Proc. Natl. Acad. Sci. U.S. Am.* 109, E3614–E3622. doi: 10.1073/pnas.1211467109
- Cichon, J., and Gan, W.-B. (2015). Branch-specific dendritic Ca^{2+} spikes cause persistent synaptic plasticity. *Nature* 520, 180–185. doi: 10.1038/nature14251
- Crick, F., and Asanuma, C. (1986). “Certain aspects of the anatomy and physiology of the cerebral cortex,” in *Parallel Distributed Processing: Explorations in the Microstructures of Cognition. Volume 2: Psychological and Biological Models*. eds D. E. Rumelhart, J. L. McClelland, and PDP Research Group (Cambridge, MA: MIT Press), 333–371.
- Doron, M., Chindemi, G., Muller, E., Markram, H., and Segev, I., et al. (2017). Timed synaptic inhibition shapes NMDA spikes, influencing local dendritic processing and global I/O properties of cortical neurons. *Cell Rep.* 21, 1550–1561. doi: 10.1016/j.celrep.2017.10.035
- Eyal, G., Verhoog, M. B., Testa-Silva, G., Deitcher, Y., Benavides-Piccione, R., DeFelipe, J., et al. (2018). Human cortical pyramidal neurons: from spines to spikes via models. *Front. Cell. Neurosci.* 12:181. doi: 10.3389/fncel.2018.00181
- Florian, R. V. (2012). The chronotron: a neuron that learns to fire temporally precise spike patterns. *PLoS ONE* 7:e040233. doi: 10.1371/journal.pone.0040233
- Gidon, A., and Segev, I. (2009). Spike-timing-dependent synaptic plasticity and synaptic democracy in dendrites. *J. Neurophysiol.* 101, 3226–3234. doi: 10.1152/jn.91349.2008
- Gidon, A., and Segev, I. (2012). Principles governing the operation of synaptic inhibition in dendrites. *Neuron* 75, 330–341. doi: 10.1016/j.neuron.2012.05.015
- Golding, N. L., and Spruston, N. (1998). Dendritic sodium spikes are variable triggers of axonal action potentials in hippocampal CA1 pyramidal neurons. *Neuron* 21, 1189–1200. doi: 10.1016/S0896-6273(00)80635-2
- Gordon, U., Polksy, A., and Schiller, J. (2006). Plasticity compartments in basal dendrites of neocortical pyramidal neurons. *J. Neurosci.* 26, 12717–12726. doi: 10.1523/JNEUROSCI.3502-06.2006
- Grupner, M., and Brunel, N. (2012). Calcium-based plasticity model explains sensitivity of synaptic changes to spike pattern, rate, and dendritic location. *Proc. Natl. Acad. Sci. U.S.A.* 109, 3991–3996. doi: 10.1073/pnas.1109359109
- Gutig, R. (2016). Spiking neurons can discover predictive features by aggregate-label learning. *Science* 351:aab4113. doi: 10.1126/science.aab4113
- Gütig, R., and Sompolinsky, H. (2006). The tempotron: a neuron that learns spike timing-based decisions. *Nat. Neurosci.* 9, 420–428. doi: 10.1038/nn1643
- Gütig, R., and Sompolinsky, H. (2009). Time-warp-invariant neuronal processing. *PLoS Biology.* 7:e1000141. doi: 10.1371/journal.pbio.1000141
- Häusser, M. (2001). Synaptic function: dendritic democracy. *Curr. Biol.* 11, 10–12. doi: 10.1016/S0960-9822(00)00034-8
- Hay, E., Hill, S., Schürmann, F., Markram, H., and Segev, I. (2011). Models of neocortical layer 5b pyramidal cells capturing a wide range of dendritic and perisomatic active properties. *PLoS Comput. Biol.* 7:e1002107. doi: 10.1371/journal.pcbi.1002107
- Heimel, J. A., Van Hooser, S. D., and Nelson, S. B. (2005). Laminar organization of response properties in primary visual cortex of the gray squirrel (*Sciurus carolinensis*). *J. Neurophysiol.* 94, 3538–3554. doi: 10.1152/jn.00106.2005
- Hengen, K. B., Lambo, M. E., Van Hooser, S. D., Katz, D. B., and Turrigiano, G. G. (2013). Firing rate homeostasis in visual cortex of freely behaving rodents. *Neuron* 80, 335–342. doi: 10.1038/jid.2014.371
- Hines, M. L., Davison, A. P., and Muller, E. (2009). NEURON and python. *Front. Neuroinformatics* 3:1. doi: 10.3389/neuro.11.001.2009
- Jahr, C. E., and Stevens, C. F. (1990). Voltage dependence of NMDA-activated predicted by single-channel kinetics. *J. Neurosci.* 10, 3178–3182.
- Kampa, B. M., Letzkus, J. J., and Stuart, G. J. (2006). Requirement of dendritic calcium spikes for induction of spike-timing-dependent synaptic plasticity. *J. Physiol.* 574, 283–290. doi: 10.1113/jphysiol.2006.111062
- Kastellakis, G., Cai, D. J., Mednick, S. C., Silva, A. J., and Poirazi, P. (2015). Synaptic clustering within dendrites: an emerging theory of memory formation. *Prog. Neurobiol.* 126, 19–35. doi: 10.1016/j.pneurobio.2014.12.002
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). “ImageNet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, 1097–1105.
- Kullmann, D. M., Moreau, A. W., Bakiri, Y., and Nicholson, E. (2012). Plasticity of inhibition. *Neuron* 75, 951–962. doi: 10.1016/j.neuron.2012.07.030
- Larkum, M. (2013). A cellular mechanism for cortical associations: an organizing principle for the cerebral cortex. *Trends Neurosci.* 36, 141–151. doi: 10.1016/j.tins.2012.11.006
- Larkum, M. E., and Nevian, T. (2008). Synaptic clustering by dendritic signalling mechanisms. *Curr. Opin. Neurobiol.* 18, 321–331. doi: 10.1016/j.conb.2008.08.013
- Losonczy, A., Makara, J. K., and Magee, J. C. (2008). Compartmentalized dendritic plasticity and input feature storage in neurons. *Nature* 452, 436–441. doi: 10.1038/nature06725
- Magee, J. C., and Johnston, D. (1995). Characterization of single voltage-gated Na^+ and Ca^{2+} channels in apical dendrites of rat CA1 pyramidal neurons. *J. Physiol.* 481, 67–90.
- Manita, S., Suzuki, T., Homma, C., Matsumoto, T., Odagawa, M., Yamada, K., et al. (2015). A top-down cortical circuit for accurate sensory perception. *Neuron* 86, 1304–1316. doi: 10.1016/j.neuron.2015.05.006
- Marr, D. (1969). A theory of cerebellar cortex. *J. Physiol.* 202, 437–470. doi: 10.2307/1776957
- McCulloch, W. S., and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* 5, 115–133. doi: 10.1007/BF02478259
- Mégias, M., Emri, Z., Freund, T. F., and Gulyás, A. I. (2001). Total number and distribution of inhibitory and excitatory synapses on hippocampal CA1 pyramidal cells. *Neuroscience* 102, 527–540. doi: 10.1016/S0306-4522(00)00496-6
- Mel, B. W. (1991). The clusteron: toward a simple abstraction for a complex neuron. *Nips* 35–42.
- Mel, B. W. (1992). NMDA-based pattern discrimination in a modeled cortical neuron. *Neural Comput.* 4, 502–517. doi: 10.1162/neco.1992.4.4.502
- Mel, B. W., Schiller, J., and Poirazi, P. (2017). Synaptic plasticity in dendrites: complications and coping strategies. *Curr. Opin. Neurobiol.* 43, 177–186. doi: 10.1016/j.conb.2017.03.012

- Moldwin, T., Schwartz, O., and Sussman, E. S. (2017). Statistical learning of melodic patterns influences the Brain's response to wrong notes. *J. Cogn. Neurosci.* 29, 2114–2122. doi: 10.1162/jocn_a_01181
- Moldwin, T., and Segev, I. (2018). Perceptron learning and classification in a modeled cortical pyramidal cell. *bioRxiv* 464826. doi: 10.1101/464826
- Muller, E., and Reimann, M. S. R. (2011). *Modification of ProbAMPA/NMDA: 2-State model*. Available online at: https://github.com/BlueBrain/BluePyOpt/blob/master/examples/l5pc/mechanisms/ProbAMPA/NMDA_EMS.mod
- Müllner, F. E., Wierenga, C. J., and Bonhoeffer, T. (2015). Precision of inhibition: dendritic inhibition by individual GABAergic synapses on hippocampal pyramidal cells is confined in space and time. *Neuron* 87, 576–589. doi: 10.1016/j.neuron.2015.07.003
- Poirazi, P., Brannon, T., and Mel, B. W. (2003). Pyramidal neuron as two-layer neural network. *Neuron* 37, 989–999. doi: 10.1016/S0896-6273(03)00149-1
- Poirazi, P., and Mel, B. W. (2001). Impact of active dendrites and structural plasticity on the memory capacity of neural tissue. *Neuron* 29, 779–796. doi: 10.1016/S0896-6273(01)00252-5
- Polsky, A., Mel, B. W., and Schiller, J. (2004). Computational subunits in thin dendrites of pyramidal cells. *Nat. Neurosci.* 7, 621–627. doi: 10.1038/nn1253
- Quiroga, R. Q., Kreiman, G., Koch, C., and Fried, I. (2008). Sparse but not "Grandmother-cell" coding in the medial temporal lobe. *Trends Cogn. Sci.* 12, 87–91. doi: 10.1016/j.tics.2007.12.003
- Quiroga, R. Q., Reddy, L., Kreiman, G., Koch, C., and Fried, I. (2005). Invariant visual representation by single neurons in the human brain. *Nature* 435, 1102–1107. doi: 10.1038/nature03687
- Rall, W. (1967). Distinguishing theoretical synaptic potentials computed for different soma-dendritic distributions of synaptic input. *J. Neurophysiol.* 30, 1138–1168.
- Rhodes, P. (2006). The properties and implications of NMDA spikes in neocortical pyramidal cells. *J. Neurosci.* 26, 6704–6715. doi: 10.1523/JNEUROSCI.3791-05.2006
- Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.* 65, 386–408. doi: 10.1037/h0042519
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature* 323, 533–536. doi: 10.1038/323533a0
- Rumsey, C. C., and Abbott, L. F. (2006). Synaptic democracy in active dendrites. *J. Neurophysiol.* 96, 2307–2318. doi: 10.1152/jn.00149.2006
- Safaryan, K., Maex, R., Maex, N., Adams, R., and Steuber, V. (2017). Nonspecific synaptic plasticity improves the recognition of sparse patterns degraded by local noise. *Sci. Rep.* 7:46550. doi: 10.1038/srep46550
- Sarid, L., Bruno, R., Sakmann, B., Segev, I., and Feldmeyer, D. (2007). Modeling a layer 4-to-layer 2/3 module of a single column in rat neocortex: interweaving *in vitro* and *in vivo* experimental observations. *Proc. Natl. Acad. Sci. U.S.A.* 104, 16353–16358. doi: 10.1073/pnas.0707853104
- Schiess, M., Urbanczik, R., and Senn, W. (2016). Somato-dendritic synaptic plasticity and error-backpropagation in active dendrites. *PLoS Comput. Biol.* 12:e1004638. doi: 10.1371/journal.pcbi.1004638
- Schiller, J., Bruno, R., Bruno, B., Segev, I., and Feldmeyer, D. (2000). NMDA spikes in basal dendrites of cortical pyramidal neurons. *Nature* 404, 285–289. doi: 10.1038/35005094
- Siegel, M., Körding, K. P., and König, P. (2000). Integrating top-down and bottom-up sensory processing by somato-dendritic interactions. *J. Comput. Neurosci.* 8, 161–173. doi: 10.1023/A:1008973215925
- Spratling, M. W. (2002). Cortical region interactions and the functional role of apical dendrites. *Behav. Cogn. Neurosci. Rev.* 1, 219–228. doi: 10.1177/1534582302001003003
- Spruston, N. (2008). Pyramidal neurons: dendritic structure and synaptic integration. *Nat. Rev. Neurosci.* 9, 206–221. doi: 10.1038/nrn2286
- Steuber, V., Mittmann, W., Hoebeek, F. E., Silver, R. A., De Zeeuw, C. I., Häusser, M., et al. (2007). Cerebellar LTD and pattern recognition by Purkinje cells. *Neuron* 54, 121–136. doi: 10.1016/j.neuron.2007.03.015
- Stuart, G., and Spruston, N. (1998). Determinants of voltage attenuation in neocortical pyramidal neuron dendrites. *J. Neurosci.* 18, 3501–10.
- Sun, Q., Srinivas, K. V., Sotayo, A., and Siegelbaum, S. A. (2014). Dendritic Na(+) spikes enable cortical input to drive action potential output from hippocampal CA2 pyramidal neurons. *eLife* 3, 1–24. doi: 10.7554/eLife.04551
- Trachtenberg, J. T., Chen, B. E., Knott, G. W., Feng, G., Sanes, G. R., Welker, E., et al. (2002). Long-term *in vivo* imaging of experience-dependent synaptic plasticity in adult cortex. *Nature* 420, 789–794. doi: 10.1111/j.1528-1157.1986.tb03495.x
- Urbanczik, R., and Senn, W. (2014). Learning by the dendritic prediction of somatic spiking. *Neuron* 81, 521–528. doi: 10.1016/j.neuron.2013.11.030
- Weber, J. P., Ujfaluassy, B. B., and Makara, J. K. (2016). Location-dependent synaptic plasticity rules by dendritic spine cooperativity. *Nat. Commun.* 7:11380. doi: 10.1038/ncomms11380
- Wulff, P., Schonewille, M., Renzi, M., Viltno, L., Sassoè-Pognetto, M., Badura, A., et al. (2009). Synaptic inhibition of Purkinje cells mediates consolidation of vestibulo-cerebellar motor learning. *Nat. Neurosci.* 12, 1042–1049. doi: 10.1038/nn.2348
- Xu, S., Jiang, W., Poo, M., and Dan, Y. (2012). Activity recall in a visual cortical ensemble. *Nat. Neurosci.* 15, 449–455. doi: 10.1038/nn.3036

Conflict of Interest: The authors declare that this study received funding from Huawei Technologies Co., Ltd. The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article or the decision to submit it for publication.

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Moldwin and Segev. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

RESEARCH ARTICLE

The gradient clusteron: A model neuron that learns to solve classification tasks via dendritic nonlinearities, structural plasticity, and gradient descent

Toviah Moldwin¹*, Menachem Kalmenson¹, Idan Segev^{1,2}

1 Edmond and Lily Safra Center for Brain Sciences, the Hebrew University of Jerusalem, Jerusalem, Israel,
2 Department of Neurobiology, the Hebrew University of Jerusalem, Jerusalem, Israel

* These authors contributed equally to this work.

* Toviah.moldwin@mail.huji.ac.il



OPEN ACCESS

Citation: Moldwin T, Kalmenson M, Segev I (2021) The gradient clusteron: A model neuron that learns to solve classification tasks via dendritic nonlinearities, structural plasticity, and gradient descent. PLoS Comput Biol 17(5): e1009015. <https://doi.org/10.1371/journal.pcbi.1009015>

Editor: Michele Migliore, National Research Council, ITALY

Received: January 18, 2021

Accepted: April 28, 2021

Published: May 24, 2021

Copyright: © 2021 Moldwin et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: Code for the paper can be found at: <https://github.com/mkblitz/The-Gradient-Clusteron>. The MNIST dataset was obtained from <http://yann.lecun.com/exdb/mnist> (36), which we imported via Tensorflow (78). Accuracy results for individual trials of each classifier for the MNIST task can be found at <https://github.com/mkblitz/The-Gradient-Clusteron/tree/main/Classifier%20comparison%20data>.

Funding: This work was made possible through a grant from Huawei Technologies Co., Ltd. [<https://>

Abstract

Synaptic clustering on neuronal dendrites has been hypothesized to play an important role in implementing pattern recognition. Neighboring synapses on a dendritic branch can interact in a synergistic, cooperative manner via nonlinear voltage-dependent mechanisms, such as NMDA receptors. Inspired by the NMDA receptor, the single-branch clusteron learning algorithm takes advantage of location-dependent multiplicative nonlinearities to solve classification tasks by randomly shuffling the locations of “under-performing” synapses on a model dendrite during learning (“structural plasticity”), eventually resulting in synapses with correlated activity being placed next to each other on the dendrite. We propose an alternative model, the gradient clusteron, or G-clusteron, which uses an analytically-derived gradient descent rule where synapses are “attracted to” or “repelled from” each other in an input- and location-dependent manner. We demonstrate the classification ability of this algorithm by testing it on the MNIST handwritten digit dataset and show that, when using a softmax activation function, the accuracy of the G-clusteron on the all-versus-all MNIST task (~85%) approaches that of logistic regression (~93%). In addition to the location update rule, we also derive a learning rule for the synaptic weights of the G-clusteron (“functional plasticity”) and show that a G-clusteron that utilizes the weight update rule can achieve ~89% accuracy on the MNIST task. We also show that a G-clusteron with both the weight and location update rules can learn to solve the XOR problem from arbitrary initial conditions.

Author summary

Artificial neural networks (ANNs) have become among the most powerful tools in artificial intelligence and machine learning, enabling computers to solve complex tasks like image recognition. Inspired by the brain, ANNs are composed of simple neuron-like units that perform a weighted sum of their synaptic inputs. Artificial neurons can “learn” by modifying the weight of each input. Biological neurons, however, are more complex.

www.huawei.com], a fellowship from AtonRâ Partners (Genève, Switzerland) [<https://www.atonra.ch/>] to T.M., a generous gift to the Segev lab by the Patrick and Lina Drahi Foundation (PLFA) [<https://plfa.info/>], the ETH domain for the Blue Brain Project [<https://www.epfl.ch/research/domains/bluebrain/>], the Gatsby Charitable Foundation [<https://www.gatsby.org.uk/>], and the NIH Grant Agreement U01MH114812. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Synapses in a real neuron can interact nonlinearly with each other in a distance-dependent manner due to voltage-dependent mechanisms, such as the NMDA receptor. We created a model neuron, called the Gradient clusteron (G-clusteron) which can use distance-dependent nonlinearities to learn to solve classification tasks by making its synapses attract or repel each other. We tested the G-clusteron on its ability to recognize handwritten digits from the MNIST dataset and showed that it can achieve a high level of classification accuracy (~85%) just by updating its synaptic locations. We also derive a rule for updating the synaptic weights of the G-clusteron and show that having the ability to update both its synaptic weights and synaptic locations allows the G-clusteron to solve the “exclusive or” (XOR) problem, which is famously impossible for a linear neuron.

Introduction

In the discipline of machine learning, artificial neural networks (ANNs) have gained great popularity due to their impressive success in solving a wide variety of computational tasks [1]. ANNs also hold a special appeal due to their similarity to the networks of neurons in biological brains, lending hope to the possibility that ANNs might serve as a general-purpose framework for replicating human-like cognition. ANNs have also been used by computational neuroscientists to model activity dynamics, computation, and learning processes in the brain [2–6].

Despite their utility for both machine learning and neuroscience, ANNs operate at a level of abstraction that disregards many of the details of biological neural networks. In particular, the artificial neurons in ANNs integrate their inputs linearly; that is to say, each input to a neuron in an ANN is given an independent weight and the neuron applies an activation function to the linear weighted sum of its inputs.

In contrast, synapses in biological neurons display an assortment of nonlinear interactions due to the passive biophysical properties of the cell as well as the active properties of voltage-gated ion channels. One nonlinear channel of particular interest is linked to the N-methyl-D-aspartate (NMDA) receptor. The NMDA channel is both ligand-gated and voltage-gated [7–9], allowing neighboring synapses on a dendrite to interact in a cooperative, supralinear manner. When two nearby NMDA synapses are activated simultaneously, the voltage-dependence of the NMDA receptor creates a voltage perturbation (depolarization) that is greater than the linear sum of the two inputs had they been activated independently [10].

The NMDA receptor has been shown to play a crucial role in structural plasticity, or the growth and elimination of synapses between neurons [11–14]. Structural plasticity plays a crucial role in development and learning [15–19]. Structural plasticity increases the likelihood that temporally correlated synapses are placed next to each other [12, 14, 20–22], which may take advantage of the NMDA supralinearity. While some structural plasticity may be due to new presynaptic-postsynaptic pairs being formed, structural plasticity can allow neurons with extant synaptic connections between them to find the most functionally effective dendritic location to make a synapse. The commonality of multiple synaptic contacts between a single presynaptic-postsynaptic pair of cells, as shown by electron microscopy [23–26], as well as evidence that the number of multiple synapse boutons increases in an enriched environment [27], lends support to the idea that connected neurons may be sampling dendritic locations so as to optimize synaptic placement.

The synergistic coactivation of neighboring inputs on the dendrite via nonlinear NMDA receptors and the ability of a neuron to relocalize its synapses in response to correlated activity have been theorized to have a computational function. An early framework proposed that, by

placing synapses with correlated inputs next to each other, a neuron could learn to solve classification tasks. To demonstrate this capability, Bartlett Mel [28] created a simplified model of a neuron with a single dendrite called the *clusteron*. The clusteron's dendrite consisted of a sequence of discrete locations, from 1 to N , where N is the number of features in the input (in an image, N would be the number of pixels). Each input to the neuron would impinge upon a specific dendritic location. The “activation” of each synapse was defined as the direct input to that synapse multiplied by the sum of the inputs to nearby synapses within a fixed radius on the dendrite.

In order to train the clusteron to recognize a class of patterns (e.g. to identify pictures containing the digit 2), the clusteron is presented with patterns from the class to be recognized (called the positive class) and stores the average activation of each synapse per pattern. At every epoch of learning, the clusteron randomly swaps the locations of “poorly performing” synapses (i.e. synapses with a low activation relative to the other synapses) with each other on the dendrite, eventually resulting in a configuration wherein correlated inputs become spatially adjacent to each other. The adjacent correlated synapses interact nonlinearly with each other, leading to a higher overall activation for the positive class of patterns.

The clusteron model has spurred a wave of research into *synaptic clustering*, the idea that synapses may be selectively bunched together on a dendrite (structural clustering), and that synaptic clusters may receive correlated inputs which can influence a neuron’s computation (functional clustering) [29]. Recent experimental work has provided evidence for both structural and functional clustering [14, 19, 30–33] with implications for visual orientation selectivity [32] and multimodal sensory adaptation [19]. For a review of both experimental and modeling work pertaining to synaptic clustering, see [29].

In this work, we propose an alternative model for learning via dendritic nonlinearities and structural plasticity on the single dendrite, which we call the *gradient clusteron*, or G-clusteron. Unlike the original clusteron model, the G-clusteron uses a dendrite with continuous-valued locations (as opposed to the discrete locations in the clusteron) and a bell-shaped distance-dependent function to model the location-dependence of interactions between synapses. These modifications allow us to analytically derive a gradient descent-based learning rule for the synaptic locations of the inputs to the G-clusteron. We show that the G-clusteron’s location update rule can learn to solve the classic MNIST handwritten digit multiclass classification task with accuracy comparable to logistic regression (i.e. a linear classifier of the form $\hat{y} = \text{sigmoid}(\mathbf{w} \cdot \mathbf{x})$, where \hat{y} represents the predicted probability that the input pattern \mathbf{x} belongs to the positive class and \mathbf{w} is a learned weight vector). Moreover, we derive an additional plasticity rule for the synaptic weights of the G-clusteron and show that when the synaptic location update rule and the synaptic weight update rule are used simultaneously, the G-clusteron can learn to solve the exclusive or (XOR) binary classification task, a feat which cannot be accomplished by a linear classifier [34].

Results

The G-clusteron is a model neuron with a single one-dimensional “dendrite” containing synapses at various dendritic locations (Fig 1A). The *activation* of a synapse is defined as the product of its weighted input with a distance-weighted sum of the weighted input of every synapse on the dendrite (including itself).

Formally, for a given real-valued input pattern vector $\mathbf{x} = [x_1, x_2, \dots, x_N]$, the real-valued activation, a_i , of the synapse with index i is:

$$a_i = w_i x_i \sum_{j=1}^N f(l_i, l_j) w_j x_j \quad (1)$$

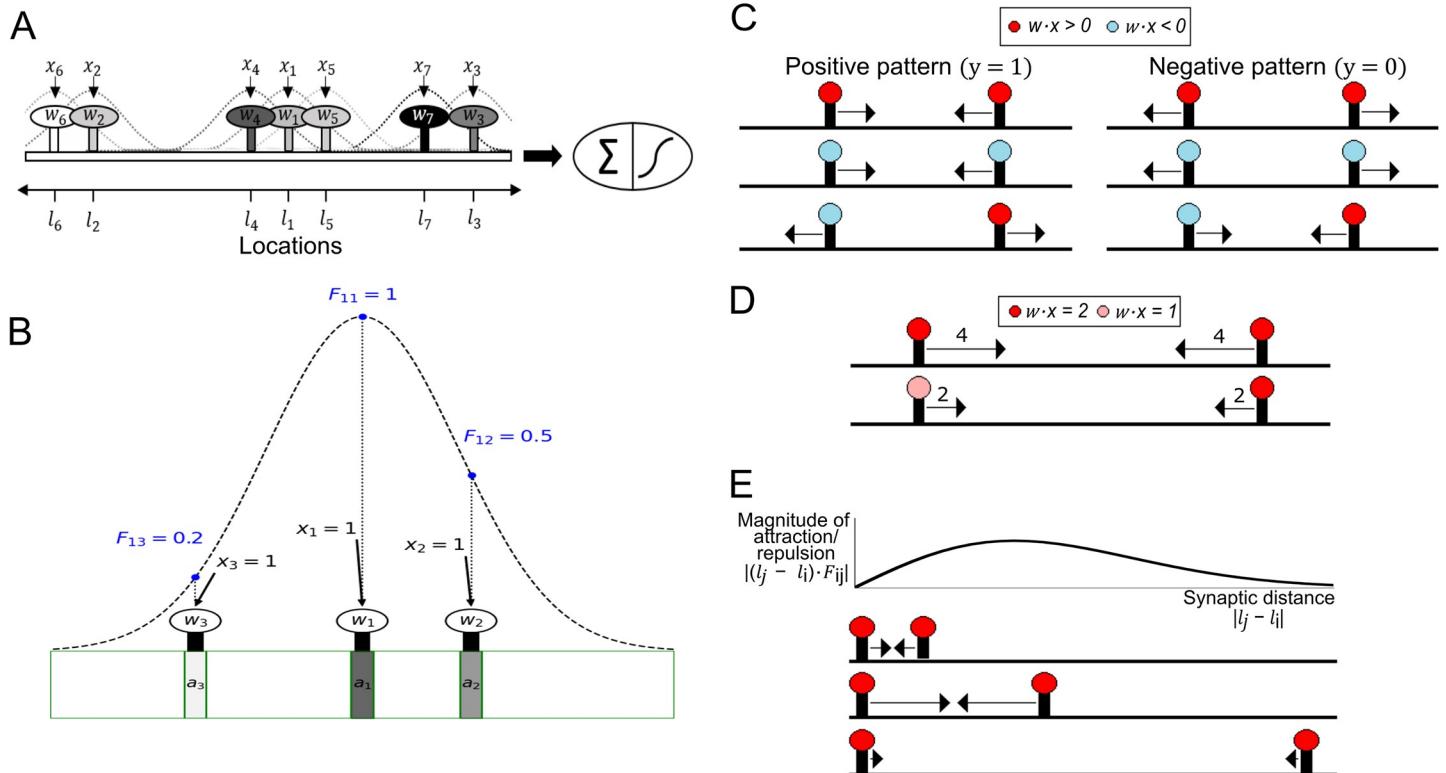


Fig 1. G-clusteron model and synaptic location update rule. (A) Schematic of the G-clusteron. Each input x_i is associated with a synapse at location l_i and weight w_i . The activation of each synapse is affected by each other synapse according to a distance-dependent factor (dashed curves). A threshold function is applied to the sum of the activations. (B) Schematic of distance-dependent interactions [Eqs 2 and 3]. The distance-dependent factor between synapse 1 and synapses 2, 3, and itself are shown as blue points along the distance-dependent function (dashed curve). Synapse 1 interacts maximally with itself, moderately with synapse 2, and slightly with synapse 3. Even though the inputs and weights for all synapses are identical, the activations (shaded rectangles, darker shades indicate larger activation) differ due to the relative locations of the synapses (e.g. synapse 1 is centrally located so it is affected by synapses 2 and 3 more than synapses 2 and 3 affect each other). (C) Schematic of the sign-dependence of the location update rule (Eq 7). Colors of the synapses indicate the sign of the weighted inputs (wx). For positive patterns (left column), two synapses attract each other if their weighted inputs have the same sign (left column, top and center) and repel each other if the weighted inputs have opposite signs (left column, bottom). In negative patterns this is reversed (right column). (D) Effect of input and weight magnitude on the magnitude of the location update rule. The magnitude of the attraction and repulsion (arrow length) is proportional to $|w_i x_i^* w_j x_j|$. (E) Top: Magnitude of the location update rule as a function of the distance between two synapses. Bottom: Three examples for the location-dependence of the update rule (aligned to the curve above). Arrow length denotes magnitude of update. This magnitude is multiplied by the weighted input factor of the learning rule from D as well as the magnitude of the error $|y - \hat{y}|$ to obtain the final magnitude.

<https://doi.org/10.1371/journal.pcbi.1009015.g001>

where l_i is the real-valued location of synapse i on the dendrite, w_i is the real-valued weight of synapse i , and $f(l_i, l_j)$ is a bell curve-shaped distance-dependent function which determines how much each synapse affects each other synapse, defined as:

$$f(l_i, l_j) = e^{-\frac{(l_i - l_j)^2}{r}} \quad (2)$$

where r is a positive-valued “radius” hyperparameter that determines the width of the curve. (From a biophysical standpoint, Eq 2 is analogous to the distance-dependence of the voltage response of a passive cable to an instantaneous current impulse [35]. In this framework, r can be thought of as the square of the cable length constant λ , where $\lambda^2 = \frac{r_m}{r_a}$, i.e. the ratio of the membrane resistance r_m (in $\Omega \cdot cm$) to the axial resistance r_a (in Ω/cm).)

Note that $f(l_i, l_j) = 1$ when $l_i = l_j$ (i.e. when synapses i and j occupy the same location) and that $f(l_i, l_j)$ approaches 0 as synapses i and j move further away from each other (Fig 1B). Also note that $f(l_i, l_j) = f(l_j, l_i)$. For convenience of notation and computation, we define a matrix F

such that

$$F_{ij} = f(l_i, l_j) \quad (3)$$

F is thus a symmetric matrix with ones on its diagonal.

For a given pattern of presynaptic inputs \mathbf{x} , the G-clusteron sums its activations together with a bias term b to produce:

$$h(\mathbf{x}) = \sum_{i=1}^N a_i - b \quad (4)$$

Because we will be using the G-clusteron as a binary classifier, we apply a sigmoidal nonlinearity g to $h(\mathbf{x})$ and produce an output \hat{y} ranging between 0 and 1:

$$\hat{y} = g(h(\mathbf{x})) \quad (5)$$

As in logistic regression, \hat{y} can be interpreted as a probability estimate for the binary label for the input pattern \mathbf{x} , with values closer to 0 representing a prediction for the label 0 (called the negative class) and values closer to 1 representing a prediction for the label 1 (called the positive class).

The G-clusteron thus differs from the original clusteron [28] in two important ways: 1) each synapse has a real-valued location on the dendrite instead of an integer-indexed location and 2) each synapse's activation function depends on the inputs of its neighbors as a gradually decreasing distance-dependent function as opposed to a hard cutoff at a fixed distance.

Defining the output of the G-clusteron in this fashion allows us to derive a gradient descent plasticity rule for the synaptic locations (assuming a logit or cross-entropy error loss function, see [Methods](#)). For each input pattern presented to the G-clusteron, we update each synaptic location according to the rule:

$$l_i \leftarrow l_i + \Delta l_i \quad (6)$$

where Δl_i is the update to the location of synapse i defined as:

$$\Delta l_i = -\eta_L (\hat{y} - y) \sum_{j=1}^N (l_j - l_i) F_{ij} w_i x_i w_j x_j \quad (7)$$

where $y \in \{0, 1\}$ is the true label (negative or positive class) for pattern \mathbf{x} , and η_L is the positive-valued learning rate for the synaptic locations. This gradient rule for each synapse can be understood as summing over “forces” that depend on pairwise interactions between that synapse and each of the other synapses on the dendrite. The interaction between two synapses depends both on the weighted inputs of the synapses ($w_i x_i$ and $w_j x_j$) and the distance between the synapses in the following manner (see [S1 Text](#) for further explanation):

1. For positive-class training patterns ($y = 1$), synapses with same-sign weighted inputs (i.e. $sign(w_i x_i) = sign(w_j x_j)$) exhibit “attraction” while synapses with opposite-sign weighted inputs (i.e. $sign(w_i x_i) = -sign(w_j x_j)$) exhibit “repulsion” ([Fig 1C](#), left).
2. For negative-class training patterns ($y = 0$), this trend is reversed: same-sign synapses are repelled, while opposite-sign synapses attract ([Fig 1C](#), right).
3. The magnitude of attraction and repulsion between two synapses is proportional to the product of the weighted inputs of the two synapses ([Fig 1D](#)).

4. The magnitude of attraction and repulsion between two synapses is distance-dependent. The attractive/repulsive force is small at very small distances, becomes larger at intermediate distances, and shrinks again at large distances ([Fig 1E](#)).

The magnitude of the update for each synapse is also scaled by the magnitude of the error term, $|\hat{y} - y|$.

[Eq 7](#) can be interpreted as a vector field along the dendrite, like force fields in physical systems of particles. If we consider a “unit synapse” such that $wx = 1$ at an arbitrary location l on the dendrite, the magnitude and direction of the “force vector” created by the plasticity rule at that location for a given input pattern x is given by:

$$\Delta l = -\eta_l (\hat{y} - y) \sum_{j=1}^N (l_j - l) f(l, l_j) w_j x_j \quad (8)$$

This interpretation of the location plasticity rule can provide some intuition as to how such an algorithm might be implemented biologically. We can imagine extracellular chemical signals being released at regions of the dendrite where there is a concentration of synaptic activity. Excitatory and inhibitory inputs would have different chemical signals associated with them. The diffusion of these chemicals in the extracellular space around the dendrite would create a location-dependent field of chemical gradients. These chemical gradients could induce presynaptic axons to form or eliminate synapses on the dendritic regions that were recently active, with differential effects for excitatory and inhibitory synapses.

Toy examples

To illustrate how the location update learning rule works, we consider several toy problems. For these problems, instead of training the G-clusteron to discriminate between two classes, we will show what happens when the G-clusteron is given a dataset where all examples are from the positive class (in other words, the G-clusteron is tasked with maximizing its output on all the input patterns, see [Methods](#) for an explanation of how the error signal $\hat{y} - y$ is implemented in these tasks) and when all examples are from the negative class (so the G-clusteron must minimize its output for all input patterns). For all the examples in this section, all synaptic weights are fixed to have the value 1 and don’t change over the course of learning; our results thus strictly depend on the synaptic locations and input values.

For the first toy problem, we create a single synaptic input vector x where the values of x are randomly distributed between -1 and 1. This input vector is repeatedly presented to the G-clusteron with a positive label ($y = 1$), such that the G-clusteron will attempt to maximize its overall activation. Because the interaction between synapses in [Eq 1](#) is multiplicative, a sensible strategy would be to minimize the distance between synapses with same-sign inputs and maximize the distance between synapses with opposite-sign inputs. This would result in two clusters on the dendrite: one with positive-valued synaptic inputs and one with negative-valued synaptic inputs. The location update rule, by causing attraction between same-sign synapses and repulsion between opposite-sign synapses, does exactly this ([Fig 2A](#) and [S1 Movie](#)).

We now show how the radius hyperparameter r (i.e. the width of the distance-dependent curve) affects the learning rule for input patterns from the positive class. In the first example, r was sufficiently large relative to the initial dispersion of synapses such that all the synapses could “feel” each other, thus enabling all synapses with the same sign to eventually aggregate into a single cluster. However, if r is reduced, synapses that start further apart from each other do not exert a strong pull on each other, so instead the learning rule operates in a more local fashion, creating several same-sign clusters ([Fig 2B](#) and [S2 Movie](#)).

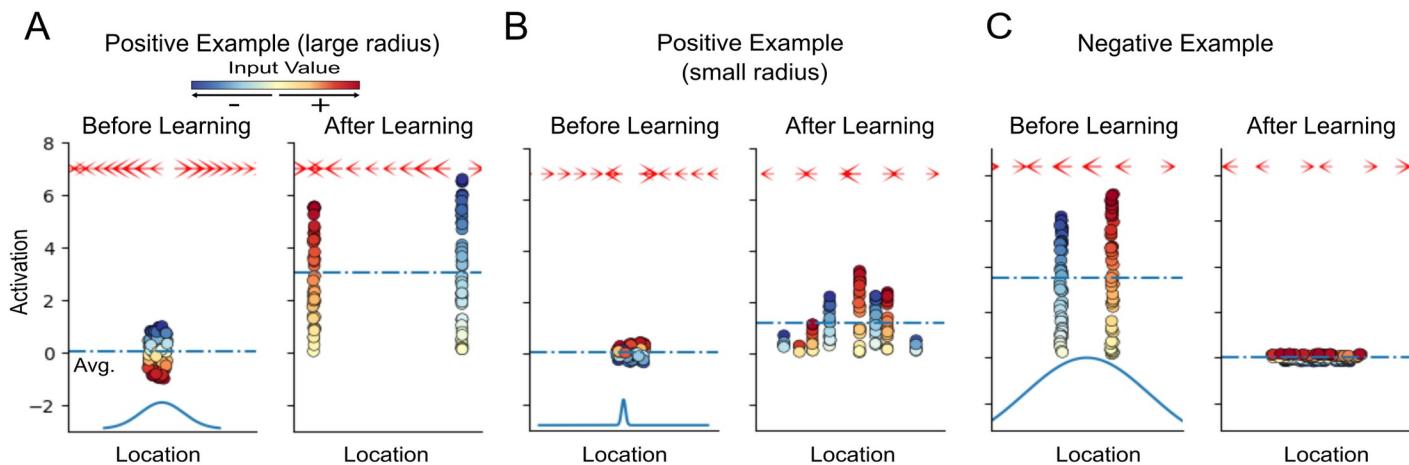


Fig 2. Dynamics of synaptic clustering due to the G-clusteron update rule. (A) Training the G-clusteron on a single positive pattern (see [S1 Movie](#)). Synaptic input values (colored circles) were drawn uniformly between -1 and +1. Position of circles along the X-axis denotes synaptic location on dendrite; their location along the Y-axis denotes synaptic activation value. In the initial epoch (left), synaptic locations are randomly initialized in close proximity. During learning, synapses with positive inputs ($x > 0$) move in the opposite direction from synapses with negative inputs ($x < 0$). In the final epoch (right) separate clusters are observed for positive and negative inputs. Dashed blue line shows the average synaptic activation, which in this case is increased by the plasticity rule. Arrows at the top of each panel denote the magnitude and direction of the “force field” created by the plasticity rule (Eq 8); by convention, the arrow at each location points in the direction that a unitary positive input ($wx = 1$) would move according to the plasticity rule. Blue curve at the bottom of the left panel shows the width of the distance-dependent function (dependent on the radius hyperparameter r) for this example. (B) Same as A with a smaller value for r (see [S2 Movie](#)). Here the plasticity rule operates on a more local scale, creating multiple smaller same-sign clusters instead of two larger clusters as in A. (C) Same as A for a negative pattern (see [S3 Movie](#)). Before learning, synapses are initialized in separate clusters for positive and negative inputs (left). Over the course of the learning, synapses are pulled toward each other until they are intermixed (right). Note that after learning the average activation level decreases.

<https://doi.org/10.1371/journal.pcbi.1009015.g002>

We next train the G-clusteron on the same input pattern, but this time with a negative label ($y = 0$), such that the G-clusteron will minimize its overall activation. Here we pre-initialize the G-clusteron in a high-activation state where synapses start in separated homogenous-signed clusters. Inversely to the first strategy, we would like opposite-sign inputs to become intermixed. Because the plasticity rule flips attraction and repulsion for input patterns from the negative class, we achieve the expected result ([Fig 2C](#) and [S3 Movie](#)).

Having demonstrated that the G-clusteron learns by aggregating and dispersing same-sign and opposite-sign synaptic inputs for individual input patterns, we now illustrate that this mechanism serves to learn the correlation structure of a dataset comprised of multiple input patterns. To this end, we create a 20-dimensional multivariate Gaussian distribution where each dimension of the Gaussian had a mean of 0 and a covariance matrix Σ such that:

$$\Sigma_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0.7 & \text{if } i \neq j \text{ and } \lceil i/5 \rceil = \lceil j/5 \rceil \\ -0.2 & \text{otherwise} \end{cases} \quad (9)$$

In other words, dimensions $x_1 \dots x_5$ were all positively correlated with each other, as were dimensions $x_6 \dots x_{10}$, $x_{11} \dots x_{15}$, and $x_{16} \dots x_{20}$, but otherwise, dimensions were slightly negatively correlated ([Fig 3A](#)). As such, each vector sampled from this distribution would exhibit similarity between the first five dimensions, second five dimensions, and so forth ([Fig 3B](#)).

As the mean of each dimension is 0, a linear neuron (without a bias) would be unable to increase its average output on this dataset, as there is no independent information in each synapse that could be given a linear weight. However, the G-clusteron can take advantage of the correlational structure of the data. Because the groups of positively correlated inputs will tend

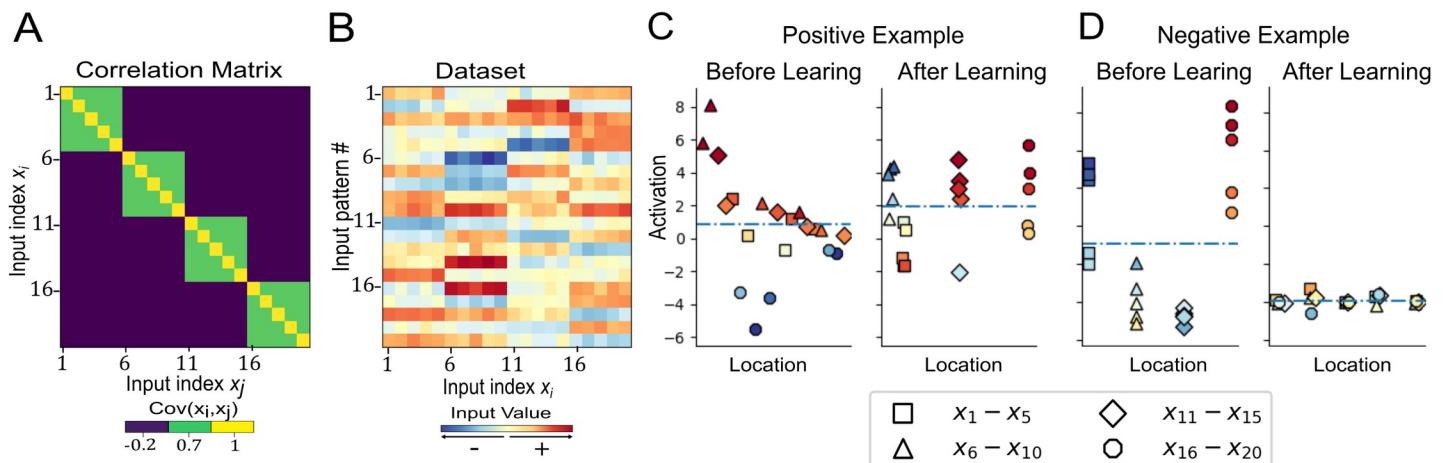


Fig 3. Dynamics of synaptic clustering for a multivariate Gaussian dataset with correlated input dimensions. (A) Covariance matrix for multivariate Gaussian example (Eq 9). This covariance matrix generates samples where dimensions $x_1 \dots x_5, x_6 \dots x_{10}$, and so on are positively correlated with each other, while all other correlations are negative. (B) Patterns drawn from the multivariate Gaussian defined with the covariance matrix in A. Each row is an input pattern presented to the G-clusteron. (C) The G-clusteron is initialized with randomized synaptic locations (left) and presented with positive-labeled input patterns as in B (see S4 Movie). Correlated inputs have the same shape; colors indicate input values for the pattern (as in B) presented in the depicted epoch of the algorithm. To increase activation, the G-clusteron groups together correlated inputs (right). (D) Same as C but for a negative example (see S5 Movie). For illustration, the G-clusteron is initialized with correlated inputs clustered together (left), as in the right panel of C. Over the course of learning the synapses eventually form clusters of negatively correlated synapses (right).

<https://doi.org/10.1371/journal.pcbi.1009015.g003>

to all be either all-positive or all-negative in any given input pattern, the learning rule will gradually move the positively correlated synapses together, creating dendritic clusters of synapses with correlated inputs. As these clusters form, the output of the G-clusteron increases (Fig 3C and S4 Movie). Inversely, the G-clusteron can learn to decrease its output on this dataset by clustering negatively correlated inputs (Fig 3D and S5 Movie). We verified that the G-clusteron appropriately increases or decreases its average activation for this task by running 100 trials of the algorithm and checking the average activation before and after learning (See S2 Text). The average activation on the positive class task before learning was 0.86, and after learning was 2.95 (average signed $d' = 1.095$). For the negative class task, the average activation before learning was 2.03, and after learning was 0.44 (average signed $d' = -1.155$).

Learning MNIST

The ability of the G-clusteron to learn the correlational structure of a dataset enables it to perform supervised classification by aggregating correlated inputs from the patterns where $y = 1$ and disaggregating correlated inputs from the patterns where $y = 0$ (Fig 4A). We trained the G-clusteron on the MNIST dataset of images of handwritten digits [36] and compared its performance to that of the original clusteron from [28] as well as to logistic regression. Before attempting the all-versus-all classification task, we test the G-clusteron separately on each digit in a one-versus-all classification paradigm (example shown in Fig 4B). For each digit from 0–9, a G-clusteron was trained on a dataset where half of the images were of that digit (positive class, label $y = 1$) and half of the images contained other digits (negative class, label $y = 0$). The G-clusteron was then tested on a holdout test set (See Methods). This procedure was repeated for the original clusteron (only positive-class training examples are used for the original clusteron, see Methods) as well as logistic regression. The learning and testing process was repeated 10 times for each classifier to ensure performance stability. The results are shown in Table 1.

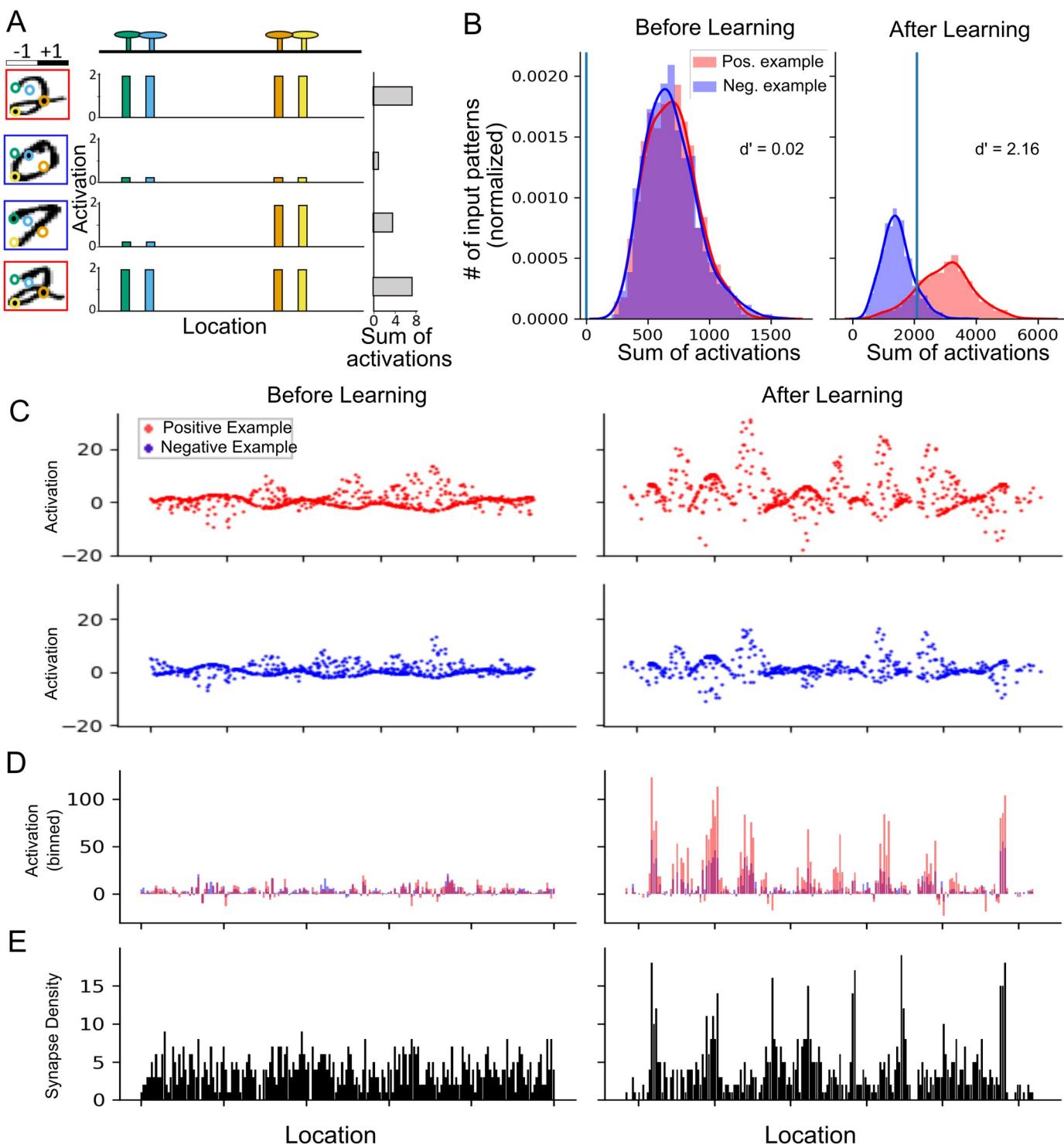


Fig 4. G-clusteron classification of handwritten digits (MNIST) and the resultant distribution of synaptic locations and activations. (A) Schematic of G-clusteron performing the one-versus-all classification task, where it must produce a larger activation for images containing the positive class 2 (digit with red border) than any other number (digit with blue border). Colored circles in the digit images indicate exemplar pixel locations mapped to exemplar synapses on the dendrite with the same color, generating two dendritic clusters (green-blue and orange-yellow, respectively). The synapses within these clusters will be maximally activated if both inputs to the cluster are the same sign (i.e. both black or both white), while opposite sign inputs will result in lower activation. (B) Exemplar histogram of sum of synaptic activations from the G-clusteron tasked with classifying the digit 2 (label $y = 1$, red) versus all other digits (label $y = 0$, blue) before learning (left) and after learning (right). Over the course of learning, the G-clusteron increases its activation on images containing a 2 relative to images displaying other digits, enabling binary classification. Blue vertical line is the value of the bias term. (C) Synaptic activations for positive (top) and negative (bottom) patterns

before (left) and after (right) learning. Note the increase in synaptic activations for positive patterns during learning. (D) Sum of the activations within small bins of dendritic length averaged over positive (red) and negative (blue) datasets before and after learning (aligned with C). Note that there are small regions (functional clusters) where the activations for positive patterns is much larger than for negative patterns. (E) Synaptic density per bin (same for positive and negative input patterns, aligned with C-D). Note that there are several high-density regions (structural clusters) that sometimes correspond to the functional clusters.

<https://doi.org/10.1371/journal.pcbi.1009015.g004>

For all digits, all three classifiers achieved a classification accuracy far above chance level of 50%. Depending on the particular digit, the clusteron achieved between 74.4% and 92.8% accuracy, the G-clusteron achieved between 80.7% and 96.8% accuracy, and logistic regression achieved between 91.2% and 98.3% accuracy. We emphasize that both the clusteron and the G-clusteron had all their weights fixed to 1 during the entire learning procedure and were only able to update their synaptic locations. Our results on the one-versus-all MNIST task thus demonstrate the remarkable efficacy of structural plasticity-based learning and suggest that the G-clusteron algorithm may be superior to the original clusteron algorithm for solving certain tasks.

The distribution of synapses and synaptic activations before and after learning MNIST (Fig 4C–4E) can be instructive in understanding how the G-clusteron operates. In many instances, over the course of learning, the synapses on the dendrite moved from an approximately uniform synaptic density to a more clustered structure, with some regions of the dendrite having a higher synaptic density than others (Fig 4E). These higher density areas occasionally had larger activations for patterns from the positive class than for patterns from the negative class (Fig 4D), suggesting that the G-clusteron may be building structural clusters to take advantage of correlated inputs in the positive class relative to the negative class. However, not all high-density clusters showed high activation, and some high-activation regions were low density. Thus, while the G-clusteron learning algorithm does produce structural clusters as a consequence of learning, there is not a guaranteed correspondence between the structural clusters and activation level in a complex task like image classification.

Having shown that the clusteron and G-clusteron exhibit satisfactory performance on the one-versus-all MNIST task, we now turn to the harder problem of all-versus-all classification. Here, we wish to train a single-layer network of 10 classifiers on the MNIST dataset (one for each digit) and have the network classify each digit correctly. We consider two standard ways to train a single-layer network on a multiclass classification problem. In the one-versus-rest (OVR) method, 10 units are independently trained on a one-versus-all paradigm, as before, and the classifier which produces the largest output (\hat{y}) for a given input pattern is declared the “winner”, and the input pattern is assigned to the positive class for that classifier (Fig 5A). In the softmax (SM) method, the units are trained simultaneously on each example and the raw outputs ($h(\mathbf{x})$) of all units are passed through a softmax function (see **Methods**), which normalizes the output of each unit by the sum of the outputs of all the units in the layer (Fig 5B). This normalized output is then used in the error term $\hat{y} - y$ when calculating the update rule. Because the softmax method allows the classifiers to communicate with each other via the output normalization, it can often lead to superior results for multiclass classification [37].

Importantly, the original single-dendrite clusteron does not utilize an error signal in its learning rule, so the original clusteron can only be trained on the multiclass task with the OVR

Table 1. Accuracies of one-versus-all on the MNIST handwritten digit dataset. Values are averaged over ten runs, with standard deviations in parentheses.

	0	1	2	3	4	5	6	7	8	9
Clusteron	92.8(0.8)	83.6(5.3)	82.5(0.5)	85.0(0.5)	77.6(1.7)	74.4(1.5)	87.5(0.8)	87.8(0.8)	80.2(1.1)	81.3(1.0)
Gradient Clusteron	96.8(0.4)	93.9(4.4)	87.6(5.3)	88.5(0.9)	91.8(1.1)	80.7(4.1)	95.1(0.4)	90.1(5.0)	82.3(4.3)	85.8(3.6)
Logistic Regression	98.2(0.0)	98.3(0.0)	94.4(0.0)	93.3(0.0)	96.7(0.0)	91.6(0.0)	96.5(0.0)	95.4(0.0)	91.2(0.0)	92.3(0.0)

<https://doi.org/10.1371/journal.pcbi.1009015.t001>

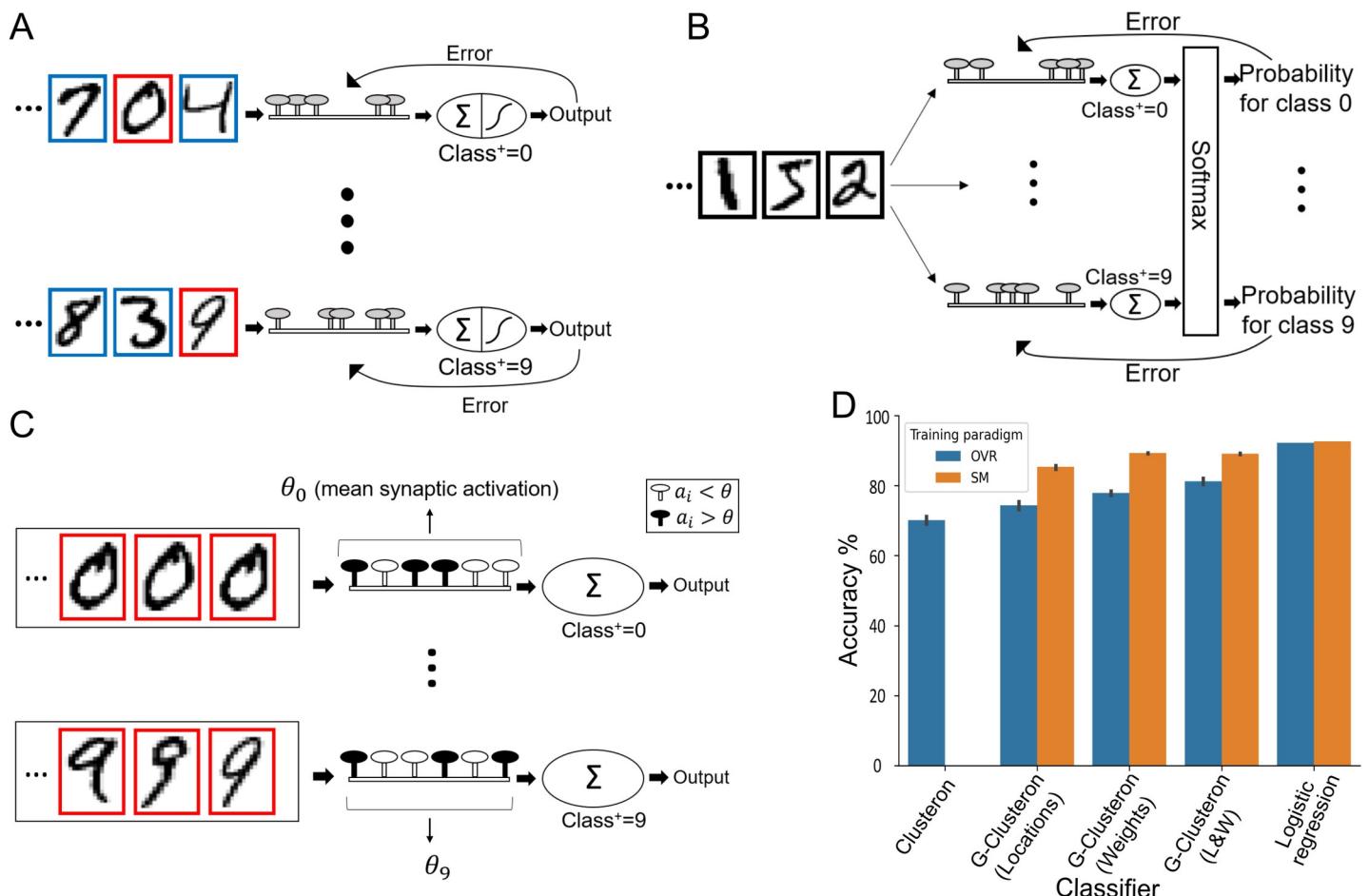


Fig 5. Multiclass classification with the G-clusteron on handwritten digits (MNIST). (A) One-versus-rest (OVR) learning scheme for the G-clusteron. Classifiers for each digit are trained independently on both positive (red border) and negative (blue border) digits. (B) Softmax (SM) classification scheme for the G-clusteron. Classifiers are trained simultaneously on each example and the sums of the synaptic activations for each classifier are fed into a softmax function which normalizes the output of each classifier by the output of all the classifiers. The error term used to update each classifier for an input digit thus has information about the output of all the classifiers for that digit. (C) OVR learning scheme for the original clusteron. On each epoch, each classifier is presented with an entire dataset consisting only of positive examples (red borders), and θ —the mean synaptic activation over the dataset—is calculated. The clusteron learns by randomly shuffling the locations of synapses whose activation was less than θ . Because each classifier only trains on positive examples and because there is no error term that can be communicated to other classifiers, the original clusteron does not lend itself to a softmax architecture as in B. (D) Mean accuracy for clusteron, G-clusteron (with either only the location update rule, only the weight update rule, or both rules), and logistic regression on the all-versus-all MNIST task using either the OVR method (blue) or SM method (orange). Error bars indicate standard deviation. Note that the original clusteron cannot use the SM method.

<https://doi.org/10.1371/journal.pcbi.1009015.g005>

paradigm (Fig 5C). Both logistic regression and the G-clusteron, however, do use an error term, making it straightforward to perform multiclass classification with both the OVR and SM methods.

To test the performance of the G-clusteron under both multiclass classification paradigms, we trained the original clusteron, the G-clusteron, and logistic regression using the OVR method, and the G-clusteron and logistic regression (but not the original clusteron) using the SM method. As with the previous task, learning for all classifiers was repeated 10 times to ensure stability.

When using the OVR paradigm, the original clusteron achieved an average accuracy of 70.1%, the G-clusteron achieves 74.3% accuracy, and logistic regression achieves 92.2% accuracy. While both the original clusteron and the G-clusteron achieve performance far better than chance level of 10%, neither of them do nearly as well as logistic regression. However,

when we train the G-clusteron using the SM method, it achieves an impressive accuracy of 85.3%, closing in on logistic regression's 92.6% accuracy in the SM scheme (Fig 5D). Although not superior to logistic regression, 85.3% accuracy on an all-versus-all MNIST classification task is notable for an algorithm that can only modify synaptic locations and not synaptic weights.

Synaptic weight update rule for the G-clusteron

In addition to the location update rule, we also derive a gradient descent rule for the weights of the G-clusteron of the form:

$$w_i \leftarrow w_i + \Delta w_i \quad (10)$$

Where Δw_i is the update to the weight of synapse i defined as (assuming a logit or cross-entropy error loss function, see [Methods](#)):

$$\Delta w_i = -\eta_W (\hat{y} - y) \frac{a_i}{w_i} \quad (11)$$

with η_W as the positive-valued learning rate for the weights. This rule can either be used on its own or in conjunction with the location update rule (Eq 6) by simultaneously updating the weights and locations using their respective rules during each epoch. To see if the weight update rule can improve the accuracy of the G-clusteron on the all-versus-all MNIST task, we trained a single-layer G-clusteron network that used only the weight update rule as well as a network that used both the location update rule and the weight update rule simultaneously, using both the OVR and SM protocols.

When only the weight update rule was used, the G-clusteron achieved an accuracy of 77.9% for OVR and 89.3% for SM, and when both the weight and location update rules were used simultaneously, the G-clusteron achieved an improved accuracy of 81.2% for OVR and a similar accuracy of 89.1% for SM. Thus, being able to manipulate synaptic weights may allow the G-clusteron to achieve slightly superior accuracy relative to a G-clusteron that can only update its synaptic locations (Fig 5D).

Although the G-clusteron with both the weight and location update rules did not perform as well as logistic regression on the MNIST task, this does not indicate that the theoretical classification capacity of the G-clusteron is inferior to that of a linear classifier. In fact, we prove (see [S3 Text](#)) that if a G-clusteron with an arbitrary localization of synapses is equipped with a single additional “bias synapse” x_0 ($x_0 = 1$ for all input patterns, this is distinct from the 0th-order bias term b), it can approximate a linear classifier to an arbitrary degree of precision by appropriately assigning the weights. The discrepancy in classification accuracy in this task between logistic regression and the G-clusteron with a weight update rule is thus likely due to the G-clusteron's failure to converge to a global minimum of the loss function rather than an inability to implement a linear separation boundary. Future optimizations to the learning algorithm might therefore allow the G-clusteron to surpass the accuracy of logistic regression on MNIST. In any event, we will now show that the G-clusteron is indeed superior to a linear classifier with respect to one particularly famous task: the XOR problem.

XOR problem

To motivate the use of the weight update rule in combination with the location update rule for the G-clusteron, we consider the XOR function. The XOR function receives two bits of binary input (x_1 and x_2) and returns a 1 if the input bits are different or a 0 if the input bits are the same. The XOR function famously cannot be implemented by linear classifiers like the

perceptron [34]. It is thus valuable to demonstrate that the G-clusteron can implement this function. We first wish to show that there are values for the parameters (i.e. the weights— w_1 and w_2 —and the locations— l_1 and l_2 —of the two synapses) of the G-clusteron that will result in a correct implementation of the XOR function. We then show that a G-clusteron must be able to update both its synaptic locations and its synaptic weights if it is to learn to solve the XOR problem from every arbitrary initialization of its parameter values.

To implement the XOR function, the parameters of the G-clusteron must satisfy both of the following inequalities (See [Methods](#)):

$$w_2^2 < -2 \cdot F_{12} \cdot w_1 w_2 \quad (12)$$

$$w_1^2 < -2 \cdot F_{12} \cdot w_1 w_2 \quad (13)$$

Note that although we originally have 4 parameters (w_1 , w_2 , l_1 , and l_2), the solution space satisfying these inequalities ([Fig 6A and 6B](#)) can be expressed in terms of only three parameters: w_1 , w_2 , and F_{12} .

When expressed in this manner, there are several things to observe about these inequalities. First, note that for $F_{12} \leq 0.5$, there is no way to assign the weights such that Eqs 12 and 13 are satisfied ([Fig 6B](#), left). This means that if the two synapses are initialized sufficiently far away from each other such that their distance-dependent factor is less than 0.5, a G-clusteron with only a weight update rule ([Eq 10](#)) would have no way to solve the XOR problem.

Moreover, the range of weights that are valid to solve the XOR problem increases as F_{12} increases from 0.5 to 1, where the solution space for the weights is maximal ([Fig 6B](#), center and right). In other words, as the synapses move closer together, there is a larger set of weights that would satisfy Eqs 12 and 13. However, even when the synapses occupy the exact same location (i.e. $F_{12} = 1$), there is still a large range of invalid weights. For example, if both weights are initialized with the same sign, a G-clusteron with only a location update rule ([Eq 6](#)) could not implement the XOR function ([Fig 6B and 6C](#)). Therefore, for the G-clusteron to solve XOR from arbitrary initial parameter values, it must be able to update both its weights and its locations.

To demonstrate this numerically, we created 3 G-clusterons: one with only a weight update rule, one with only a location update rule, and one with both update rules. For each G-clusteron, we ran 1000 trials with randomized initializations of the inputs and labels for the XOR problem. Each trial ran for 10,000 epochs or until convergence (see [Methods](#)).

For a G-clusteron with only the weight update rule, a trial was designated as “possible to converge” if some weight assignment could produce the correct XOR output given the initialization of the two synaptic locations, which can only occur if the initial F_{12} was greater than 0.5. For a G-clusteron with only the location update rule, a trial was designated as “possible to converge” if some location assignment could produce the correct XOR output given the initialization of its synaptic weights. Because the solution space for XOR grows as the synapses move closer to each other, this entails that the initial weights would have to satisfy Eqs 12 and 13 given that $F_{12} = 1$ (in other words, the weights must fall in the overlap shaded region of [Fig 6B](#), right panel). For the G-clusteron with both the weight and the location update rules, all trials could potentially converge ([Fig 6D](#)).

The G-clusteron with only the weight update rule converged for 475 trials out of 485 trials analytically possible to converge ([Fig 6D and 6E](#)), unexpectedly failing on trials where F_{12} was initialized slightly above the boundary condition of 0.5, because the solution space for the weights when F_{12} is fixed near 0.5 is small ([Fig 6B](#) middle), and it can therefore take the

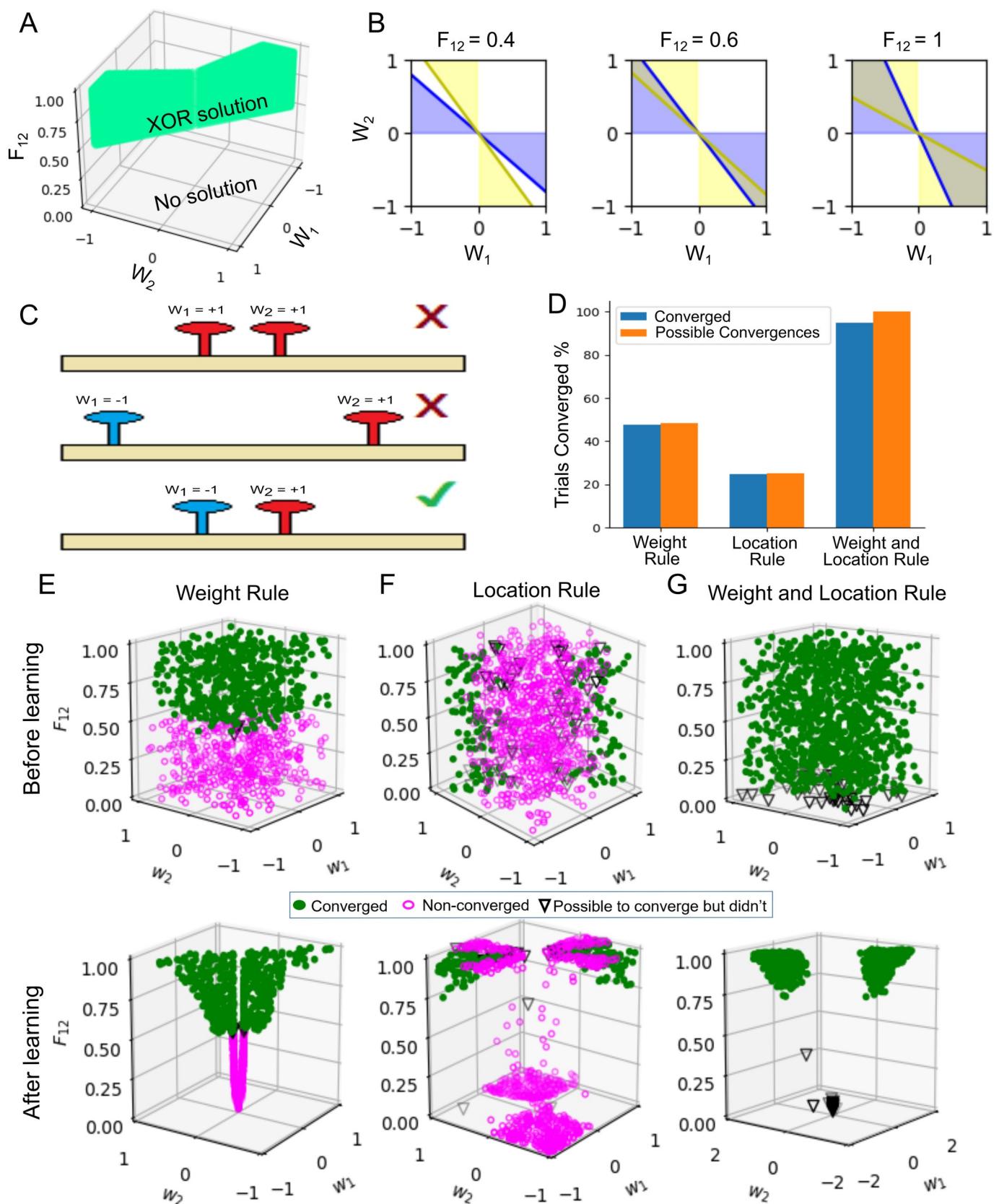


Fig 6. The G-clusteron requires the weight and location update rules to solve the XOR problem from arbitrary initial conditions. (A) Analytical solution to XOR problem. Green area represents the parameter space in terms of the two weights, w_1 and w_2 , and the distance-dependent factor, F_{12} , which would produce the correct outputs for the XOR problem. (B) Representation in weight space of the analytical solution to XOR, shown for selected values of F_{12} (i.e. slices in the F_{12} axis of A). The blue region satisfies the inequality from Eq 12, and the yellow region satisfies the inequality from Eq 13. The G-clusteron can solve the XOR problem when both inequalities are satisfied (overlap region, same as green region in A). Note that the solution space for the weights is null when $F_{12}<0.5$ and increases as F_{12} approaches 1. However, there are many weight assignments that never satisfy both inequalities. (C) Examples of correct and incorrect parameter assignments for the XOR problem. Some weight assignments never produce a correct solution for XOR regardless of the distance between the synapses (top). Moreover, if the synapses are too far away from each other, there are no weights that satisfy the XOR relation (middle). Thus, solving XOR from arbitrary initial conditions requires that the synapses be close together and the weights appropriately set (bottom). (D) Number of trials that were analytically possible to converge versus trials that successfully converged on the XOR problem for the G-clusteron trained with only the weight update rule, only the location update rule, or both rules. (E) Top: Convergence (marker shape and color, see legend) as a function of the initial parameters of 1000 randomly initialized G-clusterons with only the weight update rule. Note that only trials in which the initial value for F_{12} was greater than 0.5 converged. Bottom: Convergence as a function of the final parameters of the G-clusterons after learning. Note that converged trials fall within the region specified in A. (F) Same as E for G-clusteron using only the location update rule. Note that only trials in which the initial values for the weights were within the overlap region of B (right panel) were able to converge. (G) Same as E for G-clusteron with both the weight and location update rules. Almost all trials converge to the correct solution.

<https://doi.org/10.1371/journal.pcbi.1009015.g006>

learning algorithm an exorbitantly long time to direct the weights into this exactingly narrow range of values.

The G-clusteron with only the location update rule converged for 247 trials out of 251 trials analytically possible to converge (Fig 6D and 6F), unexpectedly failing on trials where F_{12} was initialized very close to 0, as before, and also on trials where the weights were initialized very close to the boundary of the solution space (Fig 6B, right panel). In these cases, it was evidently difficult for the algorithm to find the bias value with sufficient precision to correctly perform the classification, even though the weights and locations were analytically valid.

Only the G-clusteron with both the weight and the location update rules was able to converge on almost all of the trials (947 out of 1000 trials), unexpectedly failing only on trials where F_{12} was initialized very close to 0 (Fig 6D and 6G). These unexpected failures occurred because the magnitude of the location update rule for a given epoch scales with the value of F_{12} and the weights at that epoch (Eq 7). Thus, if F_{12} is initialized very close to 0, the synaptic locations may only move a minuscule amount each time the gradient rule is applied, depending on the values of the other parameters.

Discussion

We have shown that the G-clusteron is a robust single-neuron learning algorithm that can solve real-world classification tasks, such as MNIST handwritten digit classification. The G-clusteron's ability to solve this task merely by moving synapses on its dendrite without using synaptic weights demonstrates the computational potential of structural plasticity that makes use of distance-dependent nonlinearities.

While the original single-dendrite clusteron [28] also exhibits impressive performance when tasked with one-versus-all classification on a single digit, the lack of a supervised gradient descent-based plasticity rule makes it difficult to scale this algorithm to a multiclass classification problem, as the clusteron learning rule does not have an error signal that can be communicated between neurons within a layer via a softmax activation function.

The gradient descent plasticity rule of the G-clusteron enables effective multiclass classification and allows the use of a variety of techniques developed for classic ANNs. For example, we used the ADAM momentum-based adaptive learning method [38] to dynamically optimize our learning rate. Bringing dendritic cluster-based plasticity algorithms within the fold of gradient descent learning opens up many possibilities for the extension of the G-clusteron algorithm using the extant mathematical frameworks and robust literature for learning with gradient descent in ANNs [39].

We have also shown that incorporating a plasticity rule for the weights in a G-clusteron alongside the dendritic location plasticity rule enables the G-clusteron to solve the classic linearly inseparable XOR problem [34]. This reinforces the possibility that neuron models with dendritic nonlinearities may be more computationally powerful than the linear nodes used in ANNs [40–43].

Relationship to previous work

The original clusteron model [28] has been developed into the “two-layer model” [40, 41] in which synapses are linearly summed within dendritic branches, and each branch computes a sigmoidal nonlinearity of its input before sending its output to the soma where a final nonlinearity is applied. Unlike the original clusteron, this two-layer model does employ a supervised learning strategy that applies gradient descent to modify synaptic stability, which in turn determines how synapses are mapped to individual branches [40], however it does not directly learn exact synaptic locations via gradient descent as we do here. The two-layer model has been shown to achieve above 96% accuracy on a binary MNIST classification task (although not directly comparable to the tasks in our study, this result was superior to a linear classifier) [44]. A two-layer neuron with two branches can also implement the XOR function by, for example, having one branch implement OR, the other branch implement NAND, and the soma implement AND, each of which is a linear operation. We emphasize that the G-clusteron, as a single-branch model, is not in competition with the two-layer model in terms of classification performance. Rather, the G-clusteron is a model of very local, distance-dependent nonlinearities that may occur within a single dendritic branch. The sigmoidal output of the G-clusteron can be interpreted as a branch-level nonlinearity (as opposed to a somatic nonlinearity) and the G-clusteron can thus be incorporated as the branch-level unit in a two-layer model. This would provide an interesting avenue of future research.

In addition to the two-layer model [40, 41], there have been a variety of other attempts to model learning with dendrites and dendritic nonlinearities [45–47]. These models also tend to treat the dendritic branch (or sometimes an entire dendritic tree) as discrete loci of nonlinearity. Conceivably, both discrete, branch- or tree-dependent nonlinearities and fluid, location-dependent nonlinearities could exist simultaneously, further expanding the computational capabilities of real neurons.

Although our work here treats dendrites with NMDA receptors as nonlinear integration units, recent work [48] has shown that a detailed biophysical model of a cortical layer 5 pyramidal cell with NMDA synapses and other active mechanisms can implement the perceptron learning algorithm, implying that a neuron can behave as a linear classifier. However, this does not preclude the possibility of nonlinear integration and plasticity rules; rather it should be taken as an indication a neuron can elect to apply a simple plasticity rule that ignores the synergistic interactions between synapses and still manage to solve classification tasks as a perceptron would. Moreover, Mel [49] has shown that a clusteron-like algorithm can also be implemented in a detailed biophysical model of a cortical pyramidal cell with NMDA synapses.

The original clusteron [28] fits within the framework of subsampled quadratic classifiers, as the clusteron sums a subset of its N^2 mixed terms $x_i x_j$ [42]. The G-clusteron can be thought of as a “constrained” quadratic classifier in the sense that the equation for the G-clusteron contains all N^2 mixed terms, but the coefficients of these terms—the entries of the F matrix—are “tied together” in the sense that F must be produced from a matrix representing distances between points in one dimension, and not all matrices are valid distance matrices. The G-clusteron therefore gives us N^2 mixed terms for the price of N location parameters, but at the cost of not being able to manipulate each of the N^2 coefficients independently.

From a computational standpoint, we note that the covariance perceptron [50] also takes a correlation-based approach to solving learning tasks.

Biological plausibility

The G-clusteron takes several liberties with respect to the biological phenomena from which it is inspired. While a multiplicative activation function may be a tolerable first-order approximation of the synergistic cooperation between NMDA synapses, a sigmoidal function provides a closer fit [41]. Even to the extent that a multiplicative model does portray the synergistic relationship between excitatory NMDA synapses, such a relationship does not exist between excitatory synapses and inhibitory synapses or between inhibitory synapses and other inhibitory synapses. The multiplicative interaction with negative inputs, however, is essential to the learning protocol of the G-clusteron, so this aspect of our algorithm should be viewed as “biologically inspired” rather than an accurate depiction of what occurs in real biological cells. We do note, however, that clustering dynamics have been observed to occur between dendritic spines and inhibitory synapses [51].

Although in real biological neurons there is voltage attenuation from the dendrite to the soma [52, 53], we have chosen not to include this feature in the G-clusteron in order to emphasize the effect of local interactions, to maintain similarity to the original clusteron model [28], and to keep the model relatively simple. This choice can be justified biologically either on the basis of “synaptic democracy” [54, 55], where synaptic weights are adjusted to compensate for attenuation, or by noting that in some cases the high input resistance of distal dendritic regions may help balance out attenuation to the soma in active dendrites [56].

While some have posited that the dendritic branch may be the fundamental computational unit within a neuron [57, 58], there is reason to believe that interesting clustering dynamics occur within branches on a more local scale. Although there is no consensus definition for synaptic clustering, several studies have shown that cluster-related dynamics occur on a scale from 0–15 μm. Synapses are more likely to exhibit correlated activity if they are within 10 μm of each other [14]. Relatedly, the intercontact distance (ICD) between synapses decreases from ~10 μm to ~5 μm in owl inferior colliculus following an adaptation protocol [19]. Moreover, the relevant range for BDNF-dependent clustering seems to be ~15 μm [11], and the spread of Ras, which allows “crosstalk” between synapses in a long-term potentiation (LTP) protocol, is limited to ~10 μm [59, 60]. Heterosynaptic plasticity—structural long-term depression (LTD) occurring near spines that underwent structural LTP—also occurs on a scale of ~10 μm [21].

The mechanism of learning synaptic locations in the G-clusteron, namely the attraction and repulsion of synapses based on the activity of their presynaptic inputs, also warrants discussion from a biological standpoint. As we have described, the computation of the “forces” exerted by each synapse results in a vector field along the dendrite with regions of attraction and repulsion. Such a vector field could be implemented by the release of attractive or repulsive chemical factors in proportion to the local dendritic activation which diffuse in the extracellular space, creating a distance-dependent gradient that effectively sums the “pull” of the synapses that were active along the dendrite. The attractive factors could stimulate the growth of presynaptic axonal boutons or postsynaptic filopodia at specific regions of the dendrite, while the repulsive factors could eliminate existing synapses.

Brain-derived neurotrophic factor (BDNF) and its precursor, proBDNF, are strong candidates for the signaling mechanism underlying the sort of structural plasticity we suggest here. BDNF has been shown to be responsible for structural plasticity in development by stabilizing correlated synapses during development, while proBDNF weakens synapses that exhibit uncorrelated activity [11]. Another possible signaling agent is estradiol, which is also involved

in the formation of new spines [61] and interacts with BDNF in spine regulation pathways [62]. Astrocytes and microglia, which play an important role in spine elimination [63] may also be implicated in our model as a mechanism for intelligently rearranging synapses on the dendrite in response to local activity.

We note that the NMDA receptor is not only a mechanism for voltage nonlinearities; the receptor also plays a crucial role in the plasticity process itself by allowing Ca^{2+} ions to enter into the postsynaptic cell, initializing a cascade that can result in LTP or LTD, depending on the stimulation protocol used [64, 65]. Interestingly, the NMDA receptor is involved in structural plasticity as well, as the release of proBDNF and BDNF is NMDA-dependent [11, 12]. Moreover, there is evidence that both LTP and LTD protocols can cause structural plasticity changes, with LTP encouraging the growth of new spines and LTD resulting in spine retraction [66]. This lends support to the idea that structural and functional plasticity may be happening simultaneously, both mediated by the NMDA receptor.

Several testable experimental predictions emerge from the G-clusteron's location update rule. If presynaptic axons are indeed being coaxed into moving their synapses along chemical gradients, we might expect to see axonal boutons being destroyed and new boutons from the same axon forming a short distance away. We may also expect to see different clustering patterns on different regions of the dendrite due to different lengths constants, as in Fig 2A and 2B. Because the length constant λ increases with dendritic diameter [35, 53], we might expect that thinner regions of dendrite would have a larger number of small clusters than wider regions, which may tend to group synapses into a smaller number of large clusters. However, these effects would likely depend on the exact nature of the task that the neuron performs.

The distance-dependence of the weight update rule (Eq 11) of the G-clusteron can also be understood in a biological framework. Experimental evidence has shown that when LTP is induced in one synaptic spine, the threshold for LTP induction in nearby spines within $\sim 10\mu\text{m}$ is reduced [67], a process mediated by the intracellular diffusion of Ras [59]. The G-clusteron's weight update rule qualitatively expresses a similar phenomenon—if a synapse has a strong input such that its weight increases a large amount, nearby synapses will require less input to achieve a large weight update.

Both the weight and location update rules in our model require an error signal ($\hat{y} - y$). A prominent candidate for this error signal is the neuromodulator dopamine, which has been shown to correspond to reward prediction error [68]. Dopamine has also been demonstrated to be necessary for LTP in hippocampal CA1 [69] and is involved in learning and plasticity processes in cortex [70], adding support to the idea that dopamine-mediated error signals play an important role in supervised learning.

Future directions

The G-clusteron can be extended in a variety of directions. Because the G-clusteron is inspired by biological dendrites, we use one-dimensional synaptic locations, but our model and algorithm can be modified such that synapses interact with each other as a function of distance in a space with arbitrarily high dimensions, instead of merely moving along a line. Presynaptic axons ostensibly do not just need to localize their boutons within a branch; they may also want to decide between branches of the same neuron or among different neurons. As such, a 2D or 3D G-clusteron, where synapses are localized within the plane of a neuron's branching structure or within a 3D volume of brain tissue, also has some biological motivation. The 2D or 3D case may be more relevant during early development, when the brain is being wired up for the first time, whereas the one-dimensional case presented here may be more relevant to later stages of development or adulthood, when the brain is densely packed and neuronal processes

are likely more restricted in their motion, possibly limited to resampling locations along a dendritic branch.

Our model can also be made more biologically plausible in a number of ways, such as by using a sigmoidal nonlinearity instead of a multiplicative nonlinearity [40, 41], incorporating recent work regarding distance-dependent interactions between nonlinear synapses [71–73], or incorporating attenuation from the dendrite to the soma [52, 53]. One recent study [74] used an attenuation-based model to show how neurons could learn probability distributions via creating multiple synaptic connections at different dendritic locations. It would be valuable to see how incorporating these features into the G-clusteron model would affect both the weight and location gradient update rules.

As with any single-neuron model, the G-clusteron should also be explored in the context of a multi-layer network. The G-clusteron's gradient descent-based update rules lend themselves to the possibility of a backpropagation algorithm for a deep network of G-clusteron neurons. This can create exciting directions for a version of deep learning that incorporates synaptic weight updates together with dendritic nonlinearities and structural plasticity.

Methods

Derivation of location update rule for the G-clusteron

We wish to derive a learning rule that updates the synaptic locations on each iteration of the algorithm of the form:

$$l_i \leftarrow l_i + \Delta l_i$$

Where Δl_i is proportional to the gradient of the error with respect to the locations. For an arbitrary loss function J and nonlinearity g where we have $J(g(h(l_i)))$ we have:

$$\Delta l_i = -\eta_L \frac{\delta J}{\delta l_i}$$

By the chain rule we have:

$$\frac{\delta J}{\delta l_i} = \frac{\delta J}{\delta g} \frac{\delta g}{\delta h} \frac{\delta h}{\delta l_i}$$

The first two factors of the gradient, $\frac{\delta J}{\delta g}$ and $\frac{\delta g}{\delta h}$, are specific to the cost function and non-linearity chosen. In our case, we use the logit cross-entropy error

$$J(\theta) = -y \log(\hat{y}) - (1 - y) \log(1 - \hat{y}) \quad (14)$$

and sigmoidal nonlinearity

$$g(h) = \frac{1}{1 + e^{-h}} \quad (15)$$

We therefore have (see [75] for elaboration):

$$\frac{\delta J}{\delta g} \frac{\delta g}{\delta h} = \hat{y} - y \quad (16)$$

For the final term (see S4 Text for full derivation of $\frac{\delta h}{\delta l_i}$):

$$\frac{\delta h}{\delta l_i} = \frac{4}{r} \sum_{j=1}^N (l_j - l_i) F_{ij} w_i x_i w_j x_j$$

We thus have:

$$\frac{\delta J}{\delta l_i} = (\hat{y} - y) \frac{4}{r} \sum_{j=1}^N (l_j - l_i) F_{ij} w_i x_i w_j x_j$$

We can subsume the constant factor $\frac{4}{r}$ into the learning rate η_L , which gives:

$$\Delta l_i = -\eta_L (\hat{y} - y) \sum_{j=1}^N (l_j - l_i) F_{ij} w_i x_i w_j x_j$$

Which gives our location update rule (Eq 7). This same expression holds if we use a softmax nonlinearity:

$$\hat{y}_i = \frac{e^{h_i}}{\sum_j e^{h_j}}$$

and a cross-entropy cost function:

$$J(\theta) = - \sum_i y_i \log(\hat{y}_i)$$

as we do for the multiclass MNIST classification task (See [75]). (Here i and j refer to the class labels in a multiclass classification task).

Derivation of bias update rule

We also require a rule for the update of the bias term b . As above:

$$b \leftarrow b + \Delta b \quad (17)$$

$$\Delta b = -\eta_b \frac{\delta J}{\delta b}$$

And

$$\frac{\delta J}{\delta b} = \frac{\delta J}{\delta g} \frac{\delta g}{\delta h} \frac{\delta h}{\delta b}$$

The first two terms are as above, the final term is:

$$\frac{\delta h}{\delta b} = -1$$

so we have

$$\frac{\delta J}{\delta b} = -(\hat{y} - y)$$

And

$$\Delta b = \eta_b (\hat{y} - y) \quad (18)$$

Where η_b is the positive-valued bias learning rate. (Note the difference in sign relative to the location and weight update rules.)

Derivation of weight update rule

In addition to the location update rule, we also derive a gradient descent rule for the weights of the G-clusteron of the form:

$$w_i \leftarrow w_i + \Delta w_i$$

Where:

$$\Delta w_i = -\eta_w \frac{\delta J}{\delta w_i}$$

We have

$$\frac{\delta J}{\delta w_i} = \frac{\delta J}{\delta g} \frac{\delta g}{\delta h} \frac{\delta h}{\delta w_i}$$

As with the location update rule, we have:

$$\frac{\delta J}{\delta g} \frac{\delta g}{\delta h} = \hat{y} - y$$

Taking the derivative $\frac{\delta h}{\delta w_i}$ we obtain (see [S5 Text](#) for full derivation of $\frac{\delta h}{\delta w_i}$):

$$\frac{\delta h}{\delta w_i} = 2x_i \sum_{j=1}^N F_{ij} w_j x_j = \frac{2a_i}{w_i}$$

We can subsume the constant factor 2 into the learning rate. The weight update rule for the G-clusteron is thus:

$$\Delta w_i = -\eta_w (\hat{y} - y) \frac{a_i}{w_i}$$

Efficient computation and batch learning

For computational efficiency on the MNIST task, the output of the G-clusteron and the learning rules can be implemented in a vectorized fashion. We assume here that we will be working with a dataset comprised of multiple input patterns, with indices $p \in 1, 2, \dots, P$. We will denote input j in pattern p as $x_j^{(p)}$. To efficiently calculate each synaptic activation over an entire dataset, we define a matrix S such that

$$S_{pj} = w_j x_j^{(p)} \tag{19}$$

We also define a signed distance matrix D where:

$$D_{ij} = l_j - l_i \tag{20}$$

The distance-dependent factor matrix F can be expressed in terms of D :

$$F_{ij} = e^{-\frac{D_{ij}^2}{r}} \tag{21}$$

The activation for synapse j in pattern p , $a_j^{(p)}$, is the element A_{pj} of matrix A , which can be computed as:

$$A = S^\circ (SF) \tag{22}$$

Where \circ denotes the Hadamard product, i.e. elementwise multiplication.

For the location update rule, we utilize a batch protocol where updates are performed after observing P patterns (P here is the size of the batch, not the dataset). Here, for each pattern p in the batch we define a matrix $Q^{(p)}$ such that

$$Q_{ij}^{(p)} = w_i x_i^{(p)} w_j x_j^{(p)} \quad (23)$$

The derivative of the loss function on the entire batch with respect to location l_i is given by:

$$\frac{\delta J}{\delta l_i} = \frac{1}{P} \sum_{p=1}^P \frac{\delta J^{(p)}}{\delta l_i} = \frac{1}{P} \sum_{p=1}^P (\hat{y}^{(p)} - y^{(p)}) \frac{4}{r} \sum_{j=1}^N (l_j - l_i) f(l_i, l_j) w_i x_i^{(p)} w_j x_j^{(p)} \quad (24)$$

Using our matrix notation, we have

$$\frac{\delta J}{\delta l_i} = \frac{1}{P} \sum_{p=1}^P (\hat{y}^{(p)} - y^{(p)}) \frac{4}{r} \sum_{j=1}^N [F^\circ D^\circ Q^{(p)}]_{ij} \quad (25)$$

However, this is computationally intensive as it requires an elementwise multiplication for every input pattern in the batch ($F^\circ D$ can be precomputed for the entire batch, but $F^\circ D^\circ Q^{(p)}$ is different for every input pattern). We therefore rearrange and obtain:

$$\frac{\delta J}{\delta l_i} = \frac{4}{r} \sum_{j=1}^N \left[F^\circ D^\circ \left(\frac{1}{P} \sum_{p=1}^P (\hat{y}^{(p)} - y^{(p)}) Q^{(p)} \right) \right]_{ij} \quad (26)$$

Which averages over all input patterns in the batch first, requiring only two elementwise multiplications regardless of batch size.

For the weight update rule we have:

$$\frac{\delta J}{\delta w_i} = \frac{1}{P} \sum_{p=1}^P \frac{\delta J^{(p)}}{\delta w_i} = \frac{1}{P} \sum_{p=1}^P (\hat{y}^{(p)} - y^{(p)}) \frac{2a_i^{(p)}}{w_i} = \frac{2}{P} \sum_{p=1}^P (\hat{y}^{(p)} - y^{(p)}) \frac{A_{pi}}{w_i}$$

Which can be efficiently computed for all weights in all patterns simultaneously by multiplying each row p of A by $\hat{y}^{(p)} - y^{(p)}$ and dividing each column i of A by w_i .

Toy examples

To train the G-clusteron to continually increase or decrease its activation on a particular input pattern or dataset, we treated the activation function as a threshold function which always returned the wrong answer such that it would perform the maximal update at each epoch. For patterns with label $y = 0$, we fixed $\hat{y} = 1$ such that $\hat{y} - y = 1$, for patterns with label $y = 1$, we fixed $\hat{y} = 0$ such that $\hat{y} - y = -1$.

Learning MNIST

For the MNIST learning tasks, we used the Tensorflow [76] MNIST dataset [36], which is composed of 60,000 training examples and 10,000 test examples. The examples are split roughly evenly between the 10 digits. As a preprocessing step, we centered the data such that each image had a mean pixel value of 0.

The logistic regression implementation used as our baseline for comparison on MNIST was the default implementation in the Scikit-learn library [77], which uses the ‘LBFGS’ solver [78] and an L2 regularization penalty. Using the stochastic average gradient (‘SAG’) [79] solver and no regularization penalty achieves similar results (see repository linked in **Code and Data**

Table 2. Hyperparameters for one-versus-all learning protocols.

	Epochs	Batch size	η_B	η_W	η_L	Runtime (seconds)
Clusteron	100	~	~	~	~	810
G-clusteron	100	50	0.04	0	$5 \cdot 10^{-5}$	700
Logistic regression	Default logistic regression algorithm from the Scikit-learn library [77]. Parameters: {solver: 'lbfgs', max_iter: 100, multiclass: 'ovr'}					OVR: 67 SM: 23

<https://doi.org/10.1371/journal.pcbi.1009015.t002>

Table 3. Hyperparameters for all-versus-all learning protocols. Abbreviations: WR/LR/BR: weight/location/both rules, respectively. OVR: one-versus-rest, SM: softmax.

	Epochs	Batch size	η_B	η_W	η_L	Runtime (seconds)
Clusteron	100	~	~	~	~	827
G-clusteron (WR, OVR)	100	100	0.04	10^{-4}	0	395
G-clusteron (LR, OVR)	100	100	0.04	0	$4 \cdot 10^{-5}$	1986
G-clusteron (BR, OVR)	100	100	0.04	10^{-4}	$4 \cdot 10^{-5}$	1095
G-clusteron (WR, SM)	2000	30	10^{-5}	10^{-5}	0	317
G-clusteron (LR, SM)	2000	3	$5 \cdot 10^{-6}$	0	$5 \cdot 10^{-6}$	755
G-clusteron (BR, SM)	2000	5	10^{-5}	10^{-5}	10^{-5}	961
Logistic regression	Default logistic regression from the Scikit-learn library [77]. Parameters: {solver: 'lbfgs', max_iter: 100, multiclass: 'ovr' for OVR or 'multinomial' for SM}					OVR: 67 SM: 23

<https://doi.org/10.1371/journal.pcbi.1009015.t003>

Availability Statement. The original clusteron algorithm was implemented in Python by the authors using the description from [28].

For the one-versus-all experiments, the original clusteron was trained only on positive training examples, ~6,000 examples per digit. The G-clusteron and logistic regression classifiers were trained on a balanced dataset with ~6000 images of the digit from the positive class and ~6000 total images of other digits. The test set was evenly split between positive and negative examples. Hyperparameters and runtimes for the one-versus-all task can be found in [Table 2](#).

For the all-versus-all experiments, each of the 10 clusteron units were trained on all images from their positive class. Each G-clusteron and logistic regression classifier was trained with the entire MNIST dataset. Hyperparameters and runtimes for all-versus-all tasks are shown in [Table 3](#).

For both one-versus-all and all-versus-all (SM and OVR), the hyperparameter r of the G-clusteron was always set to 0.23. For the original clusteron, synapses interacted if they were within 10 synapses of each other. Hyperparameters were hand-tuned to maximize accuracy in all cases.

Table 4. XOR function.

Input		Output
x_1	x_2	$x_1 \text{ XOR } x_2$
0	0	0
1	0	1
0	1	1
1	1	0

<https://doi.org/10.1371/journal.pcbi.1009015.t004>

XOR problem

The XOR function is defined as ([Table 4](#)):

For the G-clusteron to solve the XOR problem, we require:

$$\max(h(0,0), h(1,1)) < \min(h(0,1), h(1,0)) \quad (27)$$

where $h(\mathbf{x})$ is the output of the G-clusteron on the input vector \mathbf{x} (In our case, the vector $[x_1, x_2]$) before applying the sigmoidal nonlinearity, as defined in [Eq 4](#).

In the case of two inputs, $h(\mathbf{x})$ can be written as:

Table 5. G-clusteron outputs for two binary inputs.

Input		Output
x_1	x_2	$h(x_1, x_2)$
0	0	$-b$
1	0	$w_1^2 - b$
0	1	$w_2^2 - b$
1	1	$w_1^2 + w_2^2 + 2 \cdot F_{12} \cdot w_1 w_2 - b$

<https://doi.org/10.1371/journal.pcbi.1009015.t005>

$$h(x_1, x_2) = w_1 x_1 (w_1 x_1 + F_{12} w_2 x_2) + w_2 x_2 (F_{12} w_1 x_1 + w_2 x_2) - b \quad (28)$$

We therefore have ([Table 5](#)):

From Eqs [27](#) and [28](#) and [Table 5](#), we require:

$$w_1^2 + w_2^2 + 2 \cdot F_{12} \cdot w_1 w_2 - b < w_1^2 - b \quad (29)$$

and

$$w_1^2 + w_2^2 + 2 \cdot F_{12} \cdot w_1 w_2 - b < w_2^2 - b \quad (30)$$

which can be simplified to Eqs [12](#) and [13](#) in **Results**. (Note that there are two other inequalities that follow from Eqs [27](#) and [28](#), namely that $w_1^2 > 0$ and $w_2^2 > 0$, which require that neither w_1 nor w_2 are equal to 0, however Eqs [29](#) and [30](#) already guarantee this.)

To test the different G-clusteron learning rules on the XOR dataset, we created a G-clusteron for each of the three learning paradigms: weight update rule only, location update rule only, and both weight and location update rules together. Each G-clusteron was run on 1000 trials with different parameter initializations for 10,000 epochs per trial.

For each trial, the weight values were chosen randomly from a uniform distribution within [-1,1], and the initial locations were chosen such that F_{12} was uniformly distributed within [0,1]. The G-clusterons were trained with a stochastic gradient descent protocol such that each epoch, one out of the four input vectors for the XOR function ([Table 4](#)) were presented to the G-clusteron, which would update its parameters according to the relevant update rule(s). For

Table 6. Hyperparameters for XOR problem.

XOR	η_B	η_W	η_L	Net runtime (seconds)
Weight rule	0.0025	0.09	0	~8100
Location rule	0.0025	0	0.05	~11,880
Both rules	0.1	0.08	0.12	~864

<https://doi.org/10.1371/journal.pcbi.1009015.t006>

computational efficiency, we stopped the learning protocol on a given trial when the algorithm converged. Convergence was defined as achieving perfect accuracy on a test set of all 4 input vectors of the XOR problem for 10 consecutive epochs. Hyperparameters and the net runtime for all 10,000 trials are shown in [Table 6](#).

For the XOR problem, the r hyperparameter was set to 1. Momentum was not used for the XOR problem. Hyperparameters were hand-tuned to maximize the number of converged trials. Runtimes differed due to different convergence probabilities for the different rules.

All scripts were written in Python and run on an Acer Aspire 5 Notebook laptop with an i7-10510U quad-core processor (1.80 GHz) and 16GB RAM, running on a Windows 10 operating system.

Supporting information

S1 Movie. (Movie for [Fig 2A](#)) Dynamics of synaptic clustering due to the G-clusteron location update rule, using a large value for the r parameter and trained on a single positive-labeled pattern. Synapse locations are initiated randomly, and then separated into groups of synapses with positive or negative inputs.

(MP4)

S2 Movie. (Movie for [Fig 2B](#)) Same as [S1 Movie](#) but using a small value for the r parameter.

(MP4)

S3 Movie. (Movie for [Fig 2C](#)) Similar to [S1 Movie](#), but trained on a negative example, with synapse locations initiated such that synapses are clustered with same-sign input values. Over the course of learning, clusters are dispersed, leading to lower overall activation.

(MP4)

S4 Movie. (Movie for [Fig 3C](#)) Dynamics of synaptic clustering for a G-clusteron trained on positive-labeled input patterns from a multivariate Gaussian dataset with multiple groups of positively correlated inputs.

(MP4)

S5 Movie. (Movie for [Fig 3D](#)) Same as [S4 Movie](#), but with negative-labeled input patterns. Clusters of negatively correlated inputs are formed over the course of learning.

(MP4)

S1 Text. Explanation of signs and directionality in the location update rule.

(PDF)

S2 Text. Results for multiple trial runs of the multivariate Gaussian task.

(PDF)

S3 Text. Proof that the G-clusteron can approximate a linear classifier by adding an additional “bias synapse” and setting the weights and bias appropriately.

(PDF)

S4 Text. Extended derivation of location update rule.

(PDF)

S5 Text. Extended derivation of weight update rule.

(PDF)

Acknowledgments

The authors would like to thank Eyal Gal and Itamar Landau for their essential contributions to the development of the G-clusteron plasticity rules. We would also like to thank David Beniaguev and Israel Nelken for their helpful suggestions and Ilenna Jones for her valuable comments on an early version of this manuscript.

Author Contributions

Conceptualization: Toviah Moldwin.

Formal analysis: Toviah Moldwin, Menachem Kalmenson.

Funding acquisition: Idan Segev.

Investigation: Toviah Moldwin, Menachem Kalmenson.

Methodology: Toviah Moldwin, Menachem Kalmenson.

Project administration: Idan Segev.

Software: Menachem Kalmenson.

Supervision: Idan Segev.

Visualization: Menachem Kalmenson.

Writing – original draft: Toviah Moldwin.

Writing – review & editing: Toviah Moldwin, Menachem Kalmenson, Idan Segev.

References

1. Lecun Y, Bengio Y, Hinton G. Deep learning. *Nature*. Nature Publishing Group; 2015. pp. 436–444. <https://doi.org/10.1038/nature14539> PMID: 26017442
2. Amit DJ. Modeling brain function: The world of attractor neural networks. Cambridge university press; 1992.
3. Memmesheimer RM, Rubin R, Ölveczky BP, Sompolinsky H. Learning Precisely Timed Spikes. *Neuron*. 2014; 82: 925–938. <https://doi.org/10.1016/j.neuron.2014.03.026> PMID: 24768299
4. Kriegeskorte N. Deep Neural Networks: A New Framework for Modeling Biological Vision and Brain Information Processing. *Annu Rev Vis Sci*. 2015; 1: 417–446. <https://doi.org/10.1146/annurev-vision-082114-035447> PMID: 28532370
5. Rumelhart D, McClelland J, Group CPR. Parallel distributed processing: explorations in the microstructure of cognition, vol. 2: psychological and biological models. 1986.
6. Saxe AM, McClelland JL, Ganguli S. A mathematical theory of semantic development in deep neural networks. *Proc Natl Acad Sci U S A*. 2019; 166: 11537–11546. <https://doi.org/10.1073/pnas.1820226116> PMID: 31101713
7. Mayer ML, Westbrook GL, Guthrie PB. Voltage-dependent block by Mg²⁺ of NMDA responses in spinal cord neurones. *Nature*. 1984. <https://doi.org/10.1038/309261a0> PMID: 6325946
8. Nowak L, Bregestovski P, Ascher P, et al. Magnesium gates glutamate-activated channels in mouse central neurones. *Nature*. 307, 462–465 (1984). <https://doi.org/10.1038/307462a0>
9. Jahr CE, Stevens CF. Voltage Dependence of NMDA-Activated Predicted by Single-Channel Kinetics. *J Neurosci*. 1990; 10: 3178–3182. <https://doi.org/10.1523/JNEUROSCI.10-09-03178.1990> PMID: 1697902
10. Polksy A, Mel BW, Schiller J. Computational subunits in thin dendrites of pyramidal cells. *Nat Neurosci*. 2004; 7: 621–7. <https://doi.org/10.1038/nn1253> PMID: 15156147
11. Niculescu D, Michaelsen-Preusse K, Güner Ü, van Dorland R, Wierenga CJ, Lohmann C. A BDNF-Mediated Push-Pull Plasticity Mechanism for Synaptic Clustering. *Cell Rep*. 2018; 24: 2063–2074. <https://doi.org/10.1016/j.celrep.2018.07.073> PMID: 30134168

12. Kleindienst T, Winnubst J, Roth-Alpermann C, Bonhoeffer T, Lohmann C. Activity-dependent clustering of functional synaptic inputs on developing hippocampal dendrites. *Neuron*. 2011; 72: 1012–1024. <https://doi.org/10.1016/j.neuron.2011.10.015> PMID: 22196336
13. Breton-Provencher V, Coté D, Saghatelyan A. Activity of the principal cells of the olfactory bulb promotes a structural dynamic on the distal dendrites of immature adult-born granule cells via activation of NMDA receptors. *J Neurosci*. 2014; 34: 1748–1759. <https://doi.org/10.1523/JNEUROSCI.3013-13.2014> PMID: 24478357
14. Takahashi N, Kitamura K, Matsuo N, Mayford M, Kano M, Matsuki N, et al. Locally Synchronized Synaptic Inputs. *Science* (80-). 2012; 335: 353–356. <https://doi.org/10.1126/science.1210362> PMID: 22267814
15. Caroni P, Donato F, Muller D. Structural plasticity upon learning: regulation and functions. *Nat Rev Neurosci*. 2012; 13: 478–490. <https://doi.org/10.1038/nrn3258> PMID: 22714019
16. Trachtenberg JT, Chen BE, Knott GW, Feng G, Sanes JR, Egbert Welker, et al. Long-term in vivo imaging of experience-dependent synaptic plasticity in adult cortex. *Nature*. 2002; 420: 789–794. <https://doi.org/10.1038/nature01273> PMID: 12490942
17. Yang G, Pan F, Gan WB. Stably maintained dendritic spines are associated with lifelong memories. *Nature*. 2009; 462: 920–924. <https://doi.org/10.1038/nature08577> PMID: 19946265
18. Elston GN, Fujita I. Pyramidal cell development: Postnatal spinogenesis, dendritic growth, axon growth, and electrophysiology. *Front Neuroanat*. 2014; 8: 1–20. <https://doi.org/10.3389/fnana.2014.00001> PMID: 24523676
19. McBride TJ, Rodriguez-Contreras A, Trinh A, Bailey R, DeBello WM. Learning drives differential clustering of axodendritic contacts in the barn owl auditory system. *J Neurosci*. 2008; 28: 6960–6973. <https://doi.org/10.1523/JNEUROSCI.1352-08.2008> PMID: 18596170
20. Lu J, Zuo Y. Clustered structural and functional plasticity of dendritic spines. *Brain Res Bull*. 2017; 129: 18–22. <https://doi.org/10.1016/j.brainresbull.2016.09.008> PMID: 27637453
21. El-Boustani S, Ip JPK, Breton-Provencher V, Knott GW, Okuno H, Bito H, et al. Locally coordinated synaptic plasticity of visual cortex neurons in vivo. *Science* (80-). 2018; 360: 1349–1354. <https://doi.org/10.1126/science.aao0862> PMID: 29930137
22. Winnubst J, Cheyne JE, Niculescu D, Lohmann C. Spontaneous Activity Drives Local Synaptic Plasticity In Vivo. *Neuron*. 2015; 87: 399–411. <https://doi.org/10.1016/j.neuron.2015.06.029> PMID: 26182421
23. Kasthuri N, Jeffrey K, Berger DR, Priebe CE, Pfister H, Kasthuri N, et al. Saturated Reconstruction of a Volume of Neocortex Resource Saturated Reconstruction of a Volume of Neocortex. 2015; 648–661. <https://doi.org/10.1016/j.cell.2015.06.054> PMID: 26232230
24. Bartol TM, Bromer C, Kinney J, Chirillo MA, Bourne JN, Harris KM, et al. Nanoconnectomic upper bound on the variability of synaptic plasticity. *eLife*. 2015; 4: 1–18. <https://doi.org/10.7554/eLife.10778> PMID: 26618907
25. Motta A, Berning M, Boergens KM, Staffler B, Beining M, Loomba S, et al. Dense connectomic reconstruction in layer 4 of the somatosensory cortex. 2019;3134. <https://doi.org/10.1126/science.aay3134> PMID: 31649140
26. Fares T, Stepanyants A. Cooperative synapse formation in the neocortex. 2009;2009.
27. Jones TA, Klintsova AY, Kilman VL, Sirevaag AM, Greenough WT. Induction of multiple synapses by experience in the visual cortex of adult rats. *Neurobiol Learn Mem*. 1997; 68: 13–20. <https://doi.org/10.1006/nlme.1997.3774> PMID: 9195585
28. Mel BW. The clusteron: Toward a simple abstraction for a complex neuron. *Nips*. 1991; 35–42.
29. Kastellakis G, Cai DJ, Mednick SC, Silva AJ, Poirazi P. Synaptic clustering within dendrites: An emerging theory of memory formation. *Progress in Neurobiology*. Elsevier Ltd; 2015. pp. 19–35. <https://doi.org/10.1016/j.pneurobio.2014.12.002> PMID: 25576663
30. Druckmann S, Feng L, Lee B, Yook C, Zhao T, Magee JC, et al. Structured Synaptic Connectivity between Hippocampal Regions. *Neuron*. 2014; 81: 629–640. <https://doi.org/10.1016/j.neuron.2013.11.026> PMID: 24412418
31. DeBello WM, McBride TJ, Nichols GS, Pannone KE, Sanculi D, Totten DJ. Input clustering and the microscale structure of local circuits. *Front Neural Circuits*. 2014; 8: 112. <https://doi.org/10.3389/fncir.2014.00112> PMID: 25309336
32. Wilson DE, Whitney DE, Scholl B, Fitzpatrick D. Orientation selectivity and the functional clustering of synaptic inputs in primary visual cortex. *Nat Neurosci*. 2016; 19: 1003–1009. <https://doi.org/10.1038/nn.4323> PMID: 27294510
33. Kerlin A, Boaz M, Flickinger D, MacLennan BJ, Dean MB, Davis C, et al. Functional clustering of dendritic activity during decision-making. *eLife*. 2019;8. <https://doi.org/10.7554/eLife.46966> PMID: 31663507

34. Minsky ML, Papert S. *Perceptrons*. MIT Press. 1969. <https://doi.org/10.1364/AO.8.001341> PMID: 20072436
35. Koch C. *Biophysics of Computation: Information Processing in Single Neurons* (Computational Neuroscience Series). USA: Oxford University Press, Inc.; 2004.
36. LeCun Y, Cortes C, Burges CJ. MNIST handwritten digit database. ATT Labs [Online] Available <http://yann.lecun.com/exdb/mnist>. 2010;2. <https://doi.org/10.1186/gb-2010-11-12-r123> PMID: 21182762
37. Duan K, Keerthi SS, Chu W, Shevade SK, Poo AN. Multi-category Classification by Soft-Max Combination of Binary Classifiers. In: Windeatt T, Roli F, editors. *Multiple Classifier Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg; 2003. pp. 125–134.
38. Kingma DP, Ba JL. Adam: A method for stochastic optimization. 3rd Int Conf Learn Represent ICLR 2015—Conf Track Proc. 2015; 1–15.
39. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. *Nature*. 1986; 323: 533–536. <https://doi.org/10.1038/323533a0>
40. Poirazi P, Mel BW. Impact of active dendrites and structural plasticity on the memory capacity of neural tissue. *Neuron*. 2001; 29: 779–796. [https://doi.org/10.1016/s0896-6273\(01\)00252-5](https://doi.org/10.1016/s0896-6273(01)00252-5) PMID: 11301036
41. Poirazi P, Brannon T, Mel BW. Pyramidal Neuron as Two-Layer Neural Network. *Neuron*. 2003; 37: 989–999. [https://doi.org/10.1016/s0896-6273\(03\)00149-1](https://doi.org/10.1016/s0896-6273(03)00149-1) PMID: 12670427
42. Poirazi P, Mel BW. Towards the memory capacity of neurons with active dendrites. *Neurocomputing*. 1999; 26–27: 237–245. [https://doi.org/10.1016/S0925-2312\(99\)00078-8](https://doi.org/10.1016/S0925-2312(99)00078-8)
43. Gidon A, Zolnik TA, Fidzinski P, Bolduan F, Papoutsis A, Poirazi P, et al. Dendritic action potentials and computation in human layer 2/3 cortical neurons. *Science (80-)*. 2020; 367: 83–87. <https://doi.org/10.1126/science.aax6239> PMID: 31896716
44. Jones IS, Kording KP. Can Single Neurons Solve MNIST? The Computational Power of Biological Dendritic Trees. *arXiv*. 2020.
45. Schiess M, Urbanczik R, Senn W. Somato-dendritic Synaptic Plasticity and Error-backpropagation in Active Dendrites. *PLoS Comput Biol*. 2016; 12: 1–18. <https://doi.org/10.1371/journal.pcbi.1004638> PMID: 26841235
46. Urbanczik R, Senn W. Learning by the Dendritic Prediction of Somatic Spiking. *Neuron*. 2014; 81: 521–528. <https://doi.org/10.1016/j.neuron.2013.11.030> PMID: 24507189
47. Hawkins J, Ahmad S. Why Neurons Have Thousands of Synapses, a Theory of Sequence Memory in Neocortex. *Front Neural Circuits*. 2016; 10: 23. <https://doi.org/10.3389/fncir.2016.00023> PMID: 27065813
48. Moldwin T, Segev I. Perceptron Learning and Classification in a Modeled Cortical Pyramidal Cell. *Front Comput Neurosci*. 2020; 14: 1–13. <https://doi.org/10.3389/fncom.2020.00001> PMID: 32082133
49. Mel BW. NMDA-Based Pattern Discrimination in a Modeled Cortical Neuron. *Neural Comput*. 1992; 4: 502–517. <https://doi.org/10.1162/heco.1992.4.4.502>
50. Gilson M, Dahmen D, Insabato A, Helias M. The covariance perceptron: A new framework for classification and processing of time series in recurrent neural networks. *bioRxiv*. 2019; 1–37. <https://doi.org/10.1101/562546>
51. Chen JL, Villa KL, Cha JW, So PTCC, Kubota Y, Nedivi E. Clustered Dynamics of Inhibitory Synapses and Dendritic Spines in the Adult Neocortex. *Neuron*. 2012; 74: 361–373. <https://doi.org/10.1016/j.neuron.2012.02.030> PMID: 22542188
52. Rall W. Distinguishing theoretical synaptic potentials computed for different soma-dendritic distributions of synaptic input. *J Neurophysiol*. 1967; 30: 1138–1168. <https://doi.org/10.1152/jn.1967.30.5.1138> PMID: 6055351
53. Rall W. Branching dendritic trees and motoneuron membrane resistivity. *Exp Neurol*. 1959. [https://doi.org/10.1016/0014-4886\(59\)90046-9](https://doi.org/10.1016/0014-4886(59)90046-9) PMID: 14435979
54. Rumsey CC, Abbott LF. Synaptic Democracy in Active Dendrites. *J Neurophysiol*. 2006; 96: 2307–2318. <https://doi.org/10.1152/jn.00149.2006> PMID: 16837665
55. Magee JC, Cook EP. Somatic EPSP amplitude is independent of synapse location in hippocampal pyramidal neurons. *Nat Neurosci*. 2000; 3: 895–903. <https://doi.org/10.1038/78800> PMID: 10966620
56. Komendantov AO, Ascoli GA. Dendritic excitability and neuronal morphology as determinants of synaptic efficacy. *J Neurophysiol*. 2009; 101: 1847–1866. <https://doi.org/10.1152/jn.01235.2007> PMID: 19176614
57. Branco T, Häusser M. The single dendritic branch as a fundamental functional unit in the nervous system. *Curr Opin Neurobiol*. 2010; 20: 494–502. <https://doi.org/10.1016/j.conb.2010.07.009> PMID: 20800473

58. Losonczy A, Magee JC. Integrative Properties of Radial Oblique Dendrites in Hippocampal CA1 Pyramidal Neurons. *Neuron*. 2006; 50: 291–307. <https://doi.org/10.1016/j.neuron.2006.03.016> PMID: 16630839
59. Harvey CD, Yasuda R, Zhong H, Svoboda K. The Spread of Ras Activity. *Science* (80-). 2008; 321: 136–140. <https://doi.org/10.1126/science.1159675> PMID: 18556515
60. Harvey CD, Svoboda K. Locally dynamic synaptic learning rules in pyramidal neuron dendrites. *Nature*. 2007; 450. <https://doi.org/10.1038/nature06416> PMID: 18097401
61. Mendez P, Garcia-Segura LM, Muller D. Estradiol promotes spine growth and synapse formation without affecting pre-established networks. *Hippocampus*. 2011; 21: 1263–1267. <https://doi.org/10.1002/hipo.20875> PMID: 20928832
62. Murphy DD, Cole NB, Segal M. Brain-derived neurotrophic factor mediates estradiol-induced dendritic spine formation in hippocampal neurons. *Proc Natl Acad Sci U S A*. 1998; 95: 11412–11417. <https://doi.org/10.1073/pnas.95.19.11412> PMID: 9736750
63. Stein IS, Zito K. Dendritic Spine Elimination: Molecular Mechanisms and Implications. *Neuroscientist*. 2019; 25: 27–47. <https://doi.org/10.1177/1073858418769644> PMID: 29716431
64. Bliss TVP, Collingridge GL. A synaptic model of memory: Long-term potentiation in the hippocampus. *Nature*. Nature Publishing Group; 1993. pp. 31–39. <https://doi.org/10.1038/361031a0> PMID: 8421494
65. Patterson M, Yasuda R. Signalling pathways underlying structural plasticity of dendritic spines. *British Journal of Pharmacology*. Wiley-Blackwell; 2011. pp. 1626–1638. <https://doi.org/10.1111/j.1476-5381.2011.01328.x> PMID: 21410464
66. Nägerl UV, Eberhorn N, Cambridge SB, Bonhoeffer T. Bidirectional activity-dependent morphological plasticity in hippocampal neurons. *Neuron*. 2004; 44: 759–767. <https://doi.org/10.1016/j.neuron.2004.11.016> PMID: 15572108
67. Harvey CD, Svoboda K. Locally dynamic synaptic learning rules in pyramidal neuron dendrites. *Nature*. 2007; 450: 1195–1200. <https://doi.org/10.1038/nature06416> PMID: 18097401
68. Schultz W. Predictive Reward Signal of Dopamine Neurons. *J Neurophysiol*. 1998; 80: 1–27. <https://doi.org/10.1152/jn.1998.80.1.1> PMID: 9658025
69. Lisman J, Grace AA, Duzel E. A neoHebbian framework for episodic memory; role of dopamine-dependent late LTP. *Trends Neurosci*. 2011; 34: 536–547. <https://doi.org/10.1016/j.tins.2011.07.006> PMID: 21851992
70. Cohen JD, Braver TS, Brown JW. Computational perspectives on dopamine function in prefrontal cortex. *Current Opinion in Neurobiology*. Elsevier Ltd; 2002. pp. 223–229. [https://doi.org/10.1016/s0959-4388\(02\)00314-8](https://doi.org/10.1016/s0959-4388(02)00314-8) PMID: 12015241
71. Jadi MP, Behabadi BF, Poleg-Polsky A, Schiller J, Mel BW. An augmented two-layer model captures nonlinear analog spatial integration effects in pyramidal neuron dendrites. *Proc IEEE*. 2014; 102: 782–798. <https://doi.org/10.1109/JPROC.2014.2312671> PMID: 25554708
72. Behabadi BF, Polsky A, Jadi M, Schiller J, Mel BW. Location-dependent excitatory synaptic interactions in pyramidal neuron dendrites. *PLoS Comput Biol*. 2012. <https://doi.org/10.1371/journal.pcbi.1002599> PMID: 22829759
73. Jadi M, Polsky A, Schiller J, Mel BW. Location-dependent effects of inhibition on local spiking in pyramidal neuron dendrites. *PLoS Comput Biol*. 2012. <https://doi.org/10.1371/journal.pcbi.1002550> PMID: 22719240
74. Hiratani N, Fukai T. Redundancy in synaptic connections enables neurons to learn optimally. *Proc Natl Acad Sci U S A*. 2018; 115: E6871–E6879. <https://doi.org/10.1073/pnas.1803274115> PMID: 29967182
75. Roelants P. Logistic classification with cross-entropy. [cited 14 Mar 2021]. Available: <https://peterroelants.github.io/posts/cross-entropy-logistic/>
76. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, et al. TensorFlow: A system for large-scale machine learning. *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016*. 2016.
77. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine learning in Python. *J Mach Learn Res*. 2011.
78. Liu DC, Nocedal J. On the limited memory BFGS method for large scale optimization. *Math Program*. 1989; 45: 503–528. <https://doi.org/10.1007/BF01589116>
79. Polyak BT, Juditsky AB. Acceleration of stochastic approximation by averaging. *SIAM J Control Optim*. 1992; 30: 838–855. <https://doi.org/10.1137/0330046>

Neuronal Excitability

Asymmetric Voltage Attenuation in Dendrites Can Enable Hierarchical Heterosynaptic Plasticity

Toviah Moldwin,^{1,*} Menachem Kalmenson,^{2,*} and Idan Segev^{1,2}

<https://doi.org/10.1523/ENEURO.0014-23.2023>

¹Edmond and Lily Safra Center for Brain Sciences and ²Department of Neurobiology, The Hebrew University of Jerusalem, 91904 Jerusalem, Israel

Abstract

Long-term synaptic plasticity is mediated via cytosolic calcium concentrations ($[Ca^{2+}]$). Using a synaptic model that implements calcium-based long-term plasticity via two sources of Ca^{2+} — NMDA receptors and voltage-gated calcium channels (VGCCs) — we show in dendritic cable simulations that the interplay between these two calcium sources can result in a diverse array of heterosynaptic effects. When spatially clustered synaptic input produces a local NMDA spike, the resulting dendritic depolarization can activate VGCCs at non-activated spines, resulting in heterosynaptic plasticity. NMDA spike activation at a given dendritic location will tend to depolarize dendritic regions that are located distally to the input site more than dendritic sites that are proximal to it. This asymmetry can produce a hierarchical effect in branching dendrites, where an NMDA spike at a proximal branch can induce heterosynaptic plasticity primarily at branches that are distal to it. We also explored how simultaneously activated synaptic clusters located at different dendritic locations synergistically affect the plasticity at the active synapses, as well as the heterosynaptic plasticity of an inactive synapse “sandwiched” between them. We conclude that the inherent electrical asymmetry of dendritic trees enables sophisticated schemes for spatially targeted supervision of heterosynaptic plasticity.

Key words: calcium; dendrites; heterosynaptic plasticity; NMDA; synaptic plasticity; VGCC

Significance Statement

Our simulations suggest a novel framework for understanding synaptic plasticity. As opposed to plasticity being controlled only locally at the target synapse (as with frequency-dependent protocols) or globally via a backpropagating action potential (as with spike timing-dependent plasticity, STDP), our results indicate that plasticity can be controlled in a sophisticated hierarchical and branch-dependent manner. Our work makes experimentally verifiable predictions for experimentalists studying plasticity and also provides a basis for further theoretical research about dendritic computation and learning.

Introduction

The brain is believed to learn and store information via modifying the strengths of the synapses between neurons, a process known as long-term plasticity (Hebb, 1949; Bliss and Lomo, 1973; Bliss and Collingridge, 1993; Whitlock et al., 2006; Nabavi et al., 2014; Humeau and Choquet, 2019). Experimentally, plasticity can be induced via a variety of stimulation protocols (Bliss and Lomo, 1973; Rose and Dunwiddie, 1986; Artola et al., 1990; Bliss

and Collingridge, 1993; Bi and Poo, 1998; O'Connor et al., 2005b; Shouval et al., 2010). While some plasticity-inducing protocols such as spike timing-dependent plasticity require postsynaptic depolarization (Bi and Poo, 1998), in many cases it is possible to produce long-term potentiation (LTP) or long-term depression (LTD) via presynaptic stimulation alone (e.g., using high-frequency or low-frequency stimulation, respectively; Artola et al., 1990; O'Connor et al., 2005a). Some have argued that presynaptic inputs (without

Received December 17, 2022; accepted June 14, 2023; First published July 6, 2023.

The authors declare no competing financial interests.

Author contributions: T.M., M.K., and I.S. designed research; T.M., M.K., and I.S. performed research; T.M. and M.K. analyzed the data; T.M., M.K. and I.S. wrote the paper.

postsynaptic spiking activity) are the primary driver of plasticity in the hippocampus (White et al., 1988; Lisman and Spruston, 2005; Hardie and Spruston, 2009) as well as in some cases in the cortex (Kumar et al., 2021).

Over the past decades, since first proposed by Lisman (1989), evidence has mounted for a calcium-based theory of plasticity, known as the calcium control hypothesis (Lisman, 1989; Mulkey and Malenka, 1992; Cummings et al., 1996; Yang et al., 1999; Cho et al., 2001; Shouval et al., 2002). In this framework, synapses change their strength depending on the cytosolic calcium concentration ($[Ca^{2+}]$) at the postsynaptic dendritic spine. If the $[Ca^{2+}]$ is low, no change occurs. If the $[Ca^{2+}]$ rises above a critical threshold for depression (θ_D), LTD occurs and the synaptic strength is decreased. If the $[Ca^{2+}]$ is above the critical threshold for potentiation (θ_P), LTP occurs and the synaptic strength is increased (Fig. 1A). [It is usually assumed that $\theta_P > \theta_D$ for cortical and hippocampal neurons (Lisman, 1989; Artola et al., 1990), but the reverse may be true for cerebellar Purkinje cells (Coesmans et al., 2004; Piochon et al., 2016)]. It is believed that calcium promotes LTP via pathways involving protein kinases such as calcium/calmodulin-dependent protein kinase II (CaMKII; Lisman, 1989; Malenka et al., 1989; Malinow et al., 1989; Neveu and Zucker, 1996), while promoting LTD via phosphatases such as calcineurin (Lisman, 1989; Mulkey et al., 1993, 1994).

There are several sources of plasticity-inducing calcium at synapses. Two of the most prominent sources are the ligand-gated and voltage-gated NMDA receptor and the voltage-gated calcium channel (VGCC, also known as the voltage-dependent calcium channel, or VDCC). Experimentally-induced plasticity is disrupted or prevented when NMDA receptors or VGCCs are blocked, indicating that the calcium current through these sources is essential for long-term plasticity (Dudek and Bear, 1992; Bi and Poo, 1998; Golding et al., 2002; Fino et al., 2010; Shindou et al., 2011). We note that internal calcium stores can also contribute to long-term plasticity (Nishiyama et al., 2000; Rose and Konnerth, 2001; Royer and Paré, 2003; Jo et al., 2008; Camiré and Topolnik, 2014; Evans and Blackwell, 2015; O'Hare et al., 2022; see Discussion).

One of the original motivations for the calcium control hypothesis (Lisman, 1989, 2001) was the phenomenon of heterosynaptic plasticity: sometimes, when a target synapse is subjected to a plasticity protocol, other nonactivated synapses are affected as well (for review, see

This work received generous support from the Drahi family foundation, the ETH domain for the Blue Brain Project, the Gatsby Charitable Foundation, and from the Office of Naval Research, agreement No. 138151-5125597.

*T.M. and M.K. contributed equally to this work.

Acknowledgments: We thank András Ecker and Giuseppe Chindemi for the use of their plastic synapse model (Chindemi et al., 2022), for support and helpful discussions in dealing with technical and conceptual issues over the course of this project, and for helpful comments on the manuscript. We also thank Yuri Rodrigues and Thomas Chater for thorough comments on the manuscript.

Correspondence should be addressed to Toviah Moldwin at Toviah.moldwin@mail.huji.ac.il.

<https://doi.org/10.1523/ENEURO.0014-23.2023>

Copyright © 2023 Moldwin et al.

This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International license, which permits unrestricted use, distribution and reproduction in any medium provided that the original work is properly attributed.

Chistiakova et al., 2014; Chater and Goda, 2021). For example, when LTP is induced at a target synapse, other synapses in the neuron can be depressed (Lynch et al., 1977). A calcium-based model can explain this phenomenon if the potentiating protocol produced a large $[Ca^{2+}]$ influx (above θ_P) in the target spine, and a smaller $[Ca^{2+}]$ (above θ_D but below θ_P) in nontarget synapses. Lisman (2001) proposed that this might happen in the following manner: when a target synapse is activated, such as by stimulating its presynaptic axons, NMDA receptors in the target spine are activated by the presynaptic glutamate, producing a calcium influx sufficient to potentiate the synapse. In addition to increasing the $[Ca^{2+}]$ in the target spine locally, the excitatory current also depolarizes the dendrite. If the depolarization is sufficient to activate VGCCs in other spines, those spines will also experience an influx of calcium, but smaller than that of the target spine, where calcium can accumulate from both NMDA receptors and VGCCs. If the $[Ca^{2+}]$ produced by the VGCCs is above θ_D but below θ_P , the nontarget synapses (where NMDA receptors are not activated) will depress (Fig. 1C).

Heterosynaptic plasticity has also been shown to be spatially sensitive, with different plastic effects being observed at nontarget synapses depending on where they are located relative to the target synapse. Some studies show heterosynaptic plasticity within short distances ($\sim 10\text{ }\mu\text{m}$) from the target synapse (Royer and Paré, 2003; Chater and Goda, 2021; Tong et al., 2021), whereas other studies show heterosynaptic effects at up to $70\text{ }\mu\text{m}$ away from the activated synapses (Engert and Bonhoeffer, 1997) or even effects that spread from the basal to the apical tree in hippocampal pyramidal neurons (Lynch et al., 1977). While the short-range effects can be potentially explained by molecular diffusion (Chater and Goda, 2021), it is unclear what the underlying principles are that determine the spatial spread of heterosynaptic plasticity over long distances, or what the functional significance of such heterosynaptic changes might be. One experimental finding demonstrated that simultaneous activation of multiple nearby synapses on a dendritic branch can induce branch-level NMDA-dependent and VGCC-dependent calcium signals (Losonczy and Magee, 2006), pointing to the possibility of branch-level plastic changes.

Another issue that arises under the calcium control hypothesis pertains to how simultaneous synaptic input at different regions of the dendrite affects plasticity. It is known that NMDA synapses can interact synergistically such that when multiple nearby synapses are activated simultaneously, the observed somatic EPSP is larger than the linear sum of individual EPSPs, because of the voltage dependence of the NMDA receptor (Polsky et al., 2004). However, how simultaneous synaptic activity at different locations on the dendrite affect plastic changes at both activated and nonactivated synapses was not systematically explored.

Recently, a model synapse was developed as part of the Blue Brain Project (Chindemi et al., 2022) which incorporates NMDA receptors, VGCCs, and calcium-dependent long-term plasticity dynamics. This synapse model

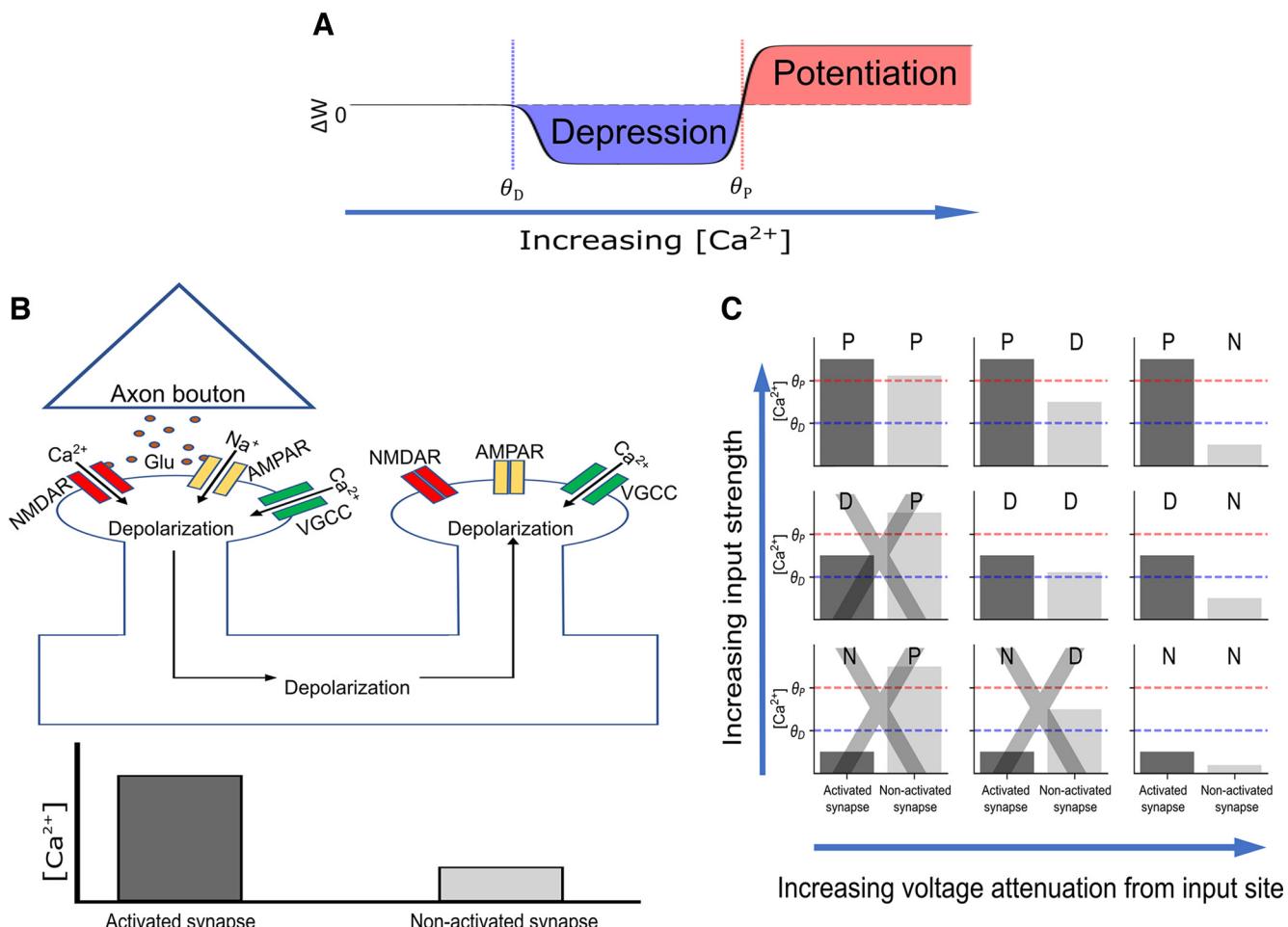


Figure 1. Induction of homosynaptic and heterosynaptic plasticity with NMDA receptors and VGCCs. **A**, Calcium control hypothesis. The synapse is weakened when the spine $[Ca^{2+}]$ crosses the depression threshold θ_D and strengthened when the $[Ca^{2+}]$ crosses the potentiation threshold θ_P . **B**, VGCC hypothesis for heterosynaptic plasticity. A presynaptic neuron spikes, releasing glutamate from its axonal bouton which binds to the AMPA and NMDA receptors of the (homosynaptic) postsynaptic spine, causing calcium influx through the NMDA channel and depolarization of the spine. The depolarization also opens the VGCCs in the activated spine, causing additional calcium influx. The depolarization spreads and depolarizes other (heterosynaptic) spines that had not been activated, opening VGCCs in these spines. However, the overall $[Ca^{2+}]$ in the nonactivated spines is smaller, as NMDA receptors were not activated. **C**, Schematic diagram of how input strength (e.g., cluster size) and spatial voltage attenuation can affect homosynaptic and heterosynaptic plasticity. N, No change; P, Potentiation; D, Depression. Panels with a gray "X" indicate scenarios that violate the assumption that activated spines have at least as much $[Ca^{2+}]$ as nonactivated spines.

(with some modifications described below in Materials and Methods) enables us to explore the [Lisman \(2001\)](#) hypothesis about the calcium basis of heterosynaptic plasticity in a dendritic cable model, which provides insight into the spatial properties of heterosynaptic plasticity.

Materials and Methods

Simulations were done using NEURON with a Python wrapper ([Hines and Carnevale, 1997](#); [Hines et al., 2009](#)). Code was written using Python 3.7.6 and NEURON version 7.7.2. Figures were made with the Matplotlib package and inkscape.

Model parameters for ball-and-stick and branched dendritic model can be found in [Table 1](#). The dendrite models were largely based on the layer 5 pyramidal cell model of

[Hay et al. \(2011\)](#) except where described otherwise in [Table 1](#). The dendritic axial resistance and diameter were chosen so as to fit with empirically described results ([Yuste and Denk, 1995](#)) and to ensure that a robust NMDA spike could be obtained with activation of ~ 20 local excitatory synapses ([Eyal et al., 2018](#)). The ball-and-stick model had a dendrite of 200 μm , composed of 50 electrical segments (cylinders).

The branching dendritic model was an order-3 branching dendritic tree (i.e., three bifurcating branching levels emanating from the zero-order main branch) coupled to a soma compartment ([Fig. 2D](#)). Each branch section was 50 μm long, resulting in a path length of 200 μm from the soma to the distal tip of each branch of the dendritic tree, consistent with the total dendritic length of the ball-and-stick model. The branched model has the same biophysical parameters as the ball-and-

Table 1: Model parameters for ball-and-stick and branched dendritic models

Model parameters	Property	Value	Reference
	R_a	Dendrite: 150 Ωcm	
	C_m	Soma: 1 $\mu\text{F}/\text{cm}^2$	Hay et al. (2011)
		Dendrite: = 1* $\mu\text{F}/\text{cm}^2$	Hay et al. (2011)
	E_{pas}	-77 mV	Hay et al. (2011)
	R_m	Soma: 30 $\text{K}\Omega\text{cm}^2$	Hay et al. (2011)
		Dendrite*: = 44* $\text{K}\Omega\text{cm}^2$	Hay et al. (2011)
	g_{max}	AMPA (INITIAL): 1.5 nS	Behabadi et al. (2012)
		AMPA (UP): 2 nS	
		AMPA (DOWN): 1 nS	
		VGCC: ~0.2 nS**	Fisher et al. (1990); Snutch et al. (2013); Tsien et al. (1988)
		NMDA: 1.31 nS	Eyal et al. (2018)
Morphologic parameters	Diameter	Dendrite: 0.75 μm	Araya et al. (2006)
	Length	Soma: 718 μm	See Materials and Methods and Extended Data Fig. 2-1
		Dendrite (ball and stick): 200 μm	
		(Branched model): 50 $\mu\text{m}/\text{branch}$	
		Soma: 23 μm	
Spine parameters	R_a	150 Ωcm	Hay et al. (2011)
	R_m	10.7 $\text{K}\Omega\text{cm}^2$	Hay et al. (2011) (no spine comp.)
	Diameter	Head: 0.4 μm	Konur et al. (2003)
		Neck: 0.07 μm	Arellano et al. (2007) (fit to ensure R_{neck} of 226.6 $\text{M}\Omega$)
	Length	Neck: 0.66 μm	Arellano et al. (2007)
	R_{neck}	Neck: 226.6 $\text{M}\Omega$	Cornejo et al. (2021)

* R_m used in the simulation was divided by 2 and C_m multiplied by 2 to compensate for surface area of unmodeled spines while maintaining the membrane time constant (Eyal et al., 2018).

** Overall conductance in the spine (gca_bar_abs_VDCC), based on the assumption of 20 pS unitary conductance per calcium channel and 20 channels per μm^2 (Sabatini and Svoboda 2000).

stick model. Each 50- μm -long branch was composed of 10 electrical segments.

To create a ball-and-stick model that replicates the electrical sink effect observed in the soma of a neuron with a full complement of extended dendrites, we expanded the diameter of the soma in the ball-and-stick model so as to have the same input resistance by applying the formula $R_N = \frac{R_m}{A}$, where R_N is the somatic input resistance of the pyramidal cell model in Ω , R_m is the membrane resistivity in Ωcm^2 , and A is the area of the compensated soma in cm^2 , resulting in a compensated soma diameter of 718 μm . As in the Hay et al. (2011) model, effective dendritic membrane resistivity dendritic R_m is divided by 2 to compensate for the surface area of (unmodeled) spines, and dendritic membrane capacitance dendritic C_m was doubled to ensure that the membrane time constant τ_m does not change.

To validate that the dendritic attenuation profile of our ball-and-stick model with its compensated soma diameter indeed replicated the attenuation profile one would observe in a real neuron, we selected one of the basal dendritic branches in the L5PC neuron and modified it to have the same length and passive electrical properties as the dendrite in the ball-and-stick model. We computed the point-to-point transfer resistance between each dendritic segment in both the ball-and-stick model and the branch from the L5PC cell model. A comparison of the resultant transfer resistance matrices demonstrated that the two models possessed the same passive electrical properties. Moreover, it was easy to see the pronounced asymmetric attenuation effect in both the L5PC model and the ball-and-stick model from the transfer resistance

matrices; the transfer resistances from all points on the dendrite exhibited a rapid drop toward the proximal direction while remaining relatively constant toward the distal direction (Extended Data Fig. 2-1).

For synapses, we modified the Blue Brain Project synapse model with NMDA receptors, VGCCs, and calcium-based long-term plasticity model (Chindemi et al., 2022); the original unmodified synapse file (GluSynapse.mod) can be found at <https://zenodo.org/record/6352774>. The calcium-based plasticity model itself is based on the work of Graupner and Brunel (2012).

We made the following several modifications to the Blue Brain synapse: (1) we initialized the synapses in a neutral state of 1.5 nS (equivalent to $\rho = 0.5$; i.e., the unstable fixed point in the Graupner–Brunel model), so synapses could be easily depressed or potentiated when the calcium accumulator crossed the plasticity thresholds; (2) we changed the maximum conductance of the NMDA receptor (gmax_NMDA) to 1.31 nS based on the study by Eyal et al. (2018); (3) we increased the unitary conductance of the VGCCs to 20 pS based on previous studies (Tsien et al., 1988; Fisher et al., 1990; Snutch et al., 2013), and assumed a channel density of 20 channels per μm^2 (Sabatini and Svoboda 2000). (While the kinetics of the VGCC model are based on the R-type VGCC, 20 pS was chosen as an approximate estimate of the typical unitary conductance over all high-voltage activated channels); and (4) we modified the equations for total spine calcium conductance and concentration to account for the fact that we were modeling the spine as a cylindrical neck with a spherical head with the parameters in Table 1. Code for the revised synapse (GluSynapse_TM_MK.mod), and all the models and

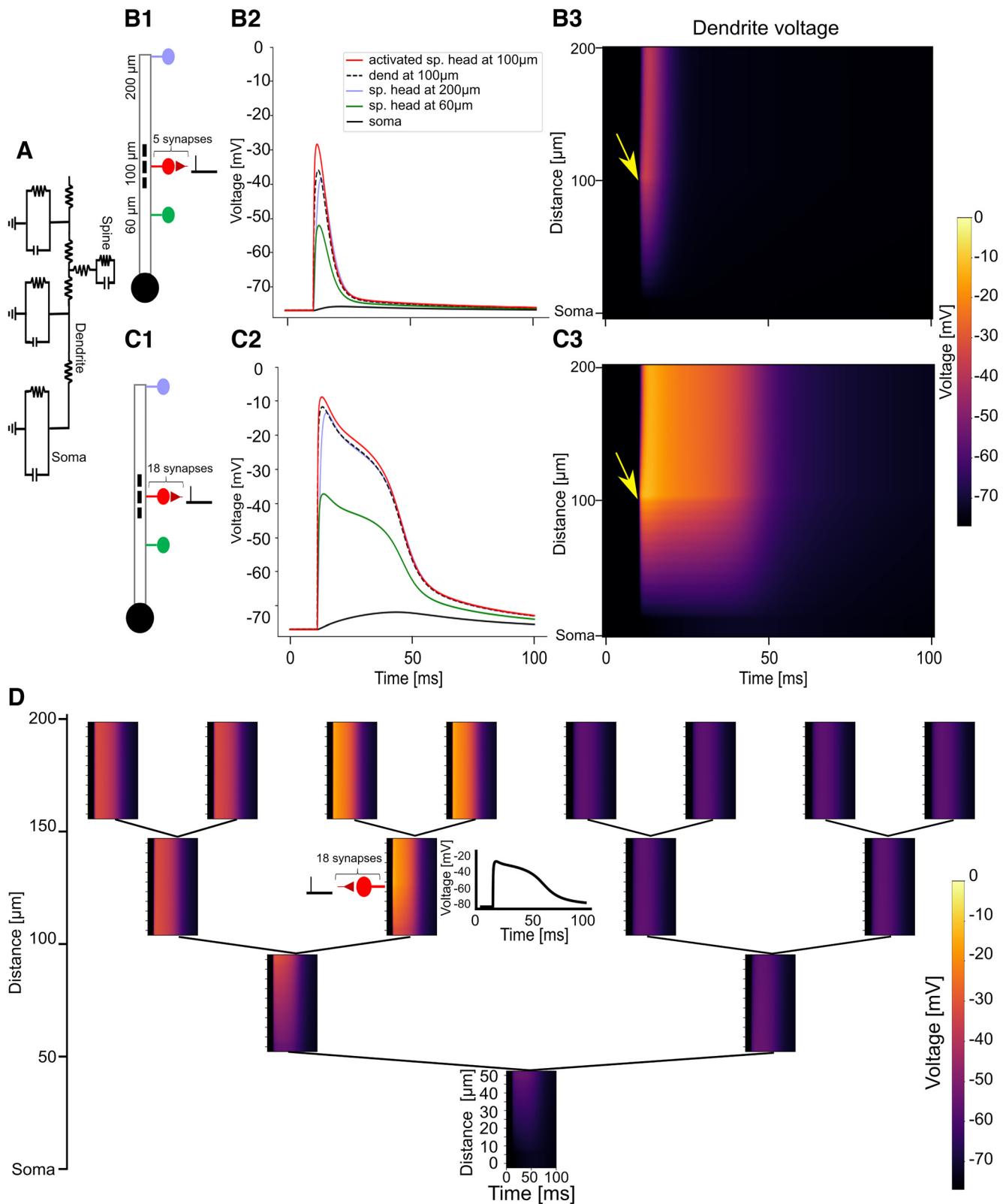


Figure 2. Asymmetric attenuation of EPSPs and NMDA spikes in dendritic cable models. **A**, Circuit diagram of a ball-and-stick dendritic model with spines. See Extended Data Figure 2-1 for transfer resistance comparison with pyramidal neuron model branch. **B1**, Experiment schematic. A cluster of 5 spine synapses located at 100 μm from the soma are simultaneously activated. Voltage is recorded from one of the activated spine heads (red spine), its base (black dashed line), a nonactivated spine at 200 μm from the soma (blue spine), and a nonactivated spine at 60 μm from the soma (green spine) and the soma (black solid circle). **B2**, Voltage

continued

traces from recording sites depicted in **B1**. Voltage is largest at the activated spine head (red solid line); it attenuates somewhat to the spine base (dashed black line). Very little attenuation occurs from the activated spine base to the distal spine head (blue), but significant attenuation is observed toward the proximal spines (green line) and the soma (black line). **B3**, Voltage recordings along the dendrite during the experiment depicted in **B1**. Color depicts dendritic voltage as a function of time (horizontal axis) and distance from soma (vertical axis). Arrow indicates time and location of the activated synaptic cluster. **C1–C3**, Same as **B1–B3** except a cluster of 18 synapses are simultaneously activated to generate an NMDA spike. See Extended Data Figure 2-2 for comparison with pyramidal neuron model branch and models with constant and scaled I_h conductance. **D**, Dendritic voltage heatmaps in each branch of an order-3 branching dendritic model in response to an NMDA spike initiated via activating a cluster of 18 synapses at the indicated dendritic location. Inset, Voltage trace at the base of the activated spine cluster. See Extended Data Figure 2-3 for voltage responses when activated synapses are less densely clustered. See Extended Data Figure 2-4 for results with asymmetric branching morphology.

simulations in this article can be found in the repository linked in the Data availability subsection.

Except where indicated, for Figures 3–6C, plasticity thresholds for the $[Ca^{2+}]$ were as follows: $\theta_D = 0.5$ and $\theta_P = 1$. For Figure 6D–F and Extended Data Figure 6-2, plasticity thresholds were $\theta_D = 0.2$ and $\theta_P = 0.4$. Plasticity thresholds can vary from cell to cell, so all plasticity results presented in all figures should be taken as qualitative illustrations of possible plastic effects rather than specific quantitative predictions.

Data availability

Code for this project can be found at <https://github.com/mkblitz/Hierarchical-hetero>.

Results

Possible heterosynaptic effects

We begin by considering the range of possible heterosynaptic effects that may occur according to the hypothesis that homosynaptic plastic effects from presynaptic plasticity induction protocols are induced by calcium influx from both NMDA receptors and VGCCs, whereas heterosynaptic effects are induced only via calcium influx through VGCCs. In this view, a spine activated with presynaptic input will almost inevitably have a higher calcium concentration than nonactivated spines, as both NMDA receptors and VGCCs can enable calcium influx in the activated spine, but only VGCCs can be opened in the nonactivated spine (Fig. 1B).

Because calcium thresholds for LTP and LTD can vary from cell to cell (Yang et al., 1999), and can also be changed via metaplastic processes (Abraham and Bear, 1996), we generically map out several possible results that can occur to an activated and nonactivated synapse given a few basic assumptions, as follows: (1) an activated synapse has a higher calcium influx than a nonactivated synapse; (2) plasticity thresholds and voltage-gated channel densities are approximately the same from spine to spine within the same neuron; and (3) that the $[Ca^{2+}]$ threshold for potentiation is higher than that of depression (i.e., $\theta_P > \theta_D$). We also disregard the magnitude of the plastic change and only consider the direction (potentiation or depression), as we assume that after inducing plasticity, synapses eventually drift toward a binary potentiated or depressed state, based on (Graupner and Brunel, 2012). (Some other assumptions also apply; for more details, see Discussion.)

Given these assumptions, the following plastic effects can result (Fig. 1C): if the activated synapse is potentiated, nonactivated synapses can also be potentiated (PP), but they can also be depressed (PD), or undergo no change (PN). If the activated synapse is depressed, nonactivated synapses can also be depressed (DD) or undergo no change (DN). If the activated synapse does not change, neither will the nonactivated synapse (NN). Given our assumptions above, the following possibilities are not possible: DP, NP, and ND (Fig. 1C). (In the event that $\theta_D > \theta_P$, as in Purkinje cells, the allowed possibilities are PP, PN, DP, DD, DN, NN, and the disallowed possibilities are PD, NP, and ND; however, the simulations used in this study assume $\theta_P > \theta_D$ to match known results from the hippocampus and cortex (Mulkey and Malenka, 1992; Cho et al., 2001). We also assume that the input to the activated synapse is not sufficiently large to put it into a ‘postpotentiative neutral zone’ where the calcium concentration is so high that potentiation mechanisms are inactivated (Tigaret et al., 2016).

Intuitively, given values for θ_D and θ_P , homosynaptic effects from presynaptic input protocols will vary based on the “strength” of synaptic input (e.g., input frequency or cluster size, as we use here). Strong inputs can potentiate the synapse, medium strength inputs can depress it, and weak inputs will induce no change. As heterosynaptic effects are mediated by dendritic depolarization, heterosynaptic effects will also depend on the input strength to the activated synapse, in addition to other factors that determine the “spillover effect” of the dendritic depolarization of the activated synapse on nonactive synapses.

One important factor that can affect this electrical spillover between synapses is the distance of the nonactivated synapse from the activated synapse. Because voltage attenuates with electrotonic distance in dendrites (Rall, 1967; Rall and Rinzel, 1973), we would naively expect that nonactivated synapses that are closer to the activated synapse will see more dendritic depolarization, and are thus likely to have a larger calcium influx through VGCCs than synapses that are further away from the activated synapse.

Asymmetric attenuation of EPSPs and NMDA dendritic spikes in dendrites

The distance-dependent attenuation description of heterosynaptic plasticity is complicated by the fact that voltage attenuation in the dendrite is highly asymmetric. For distal dendritic inputs, the proximal dendrites and soma

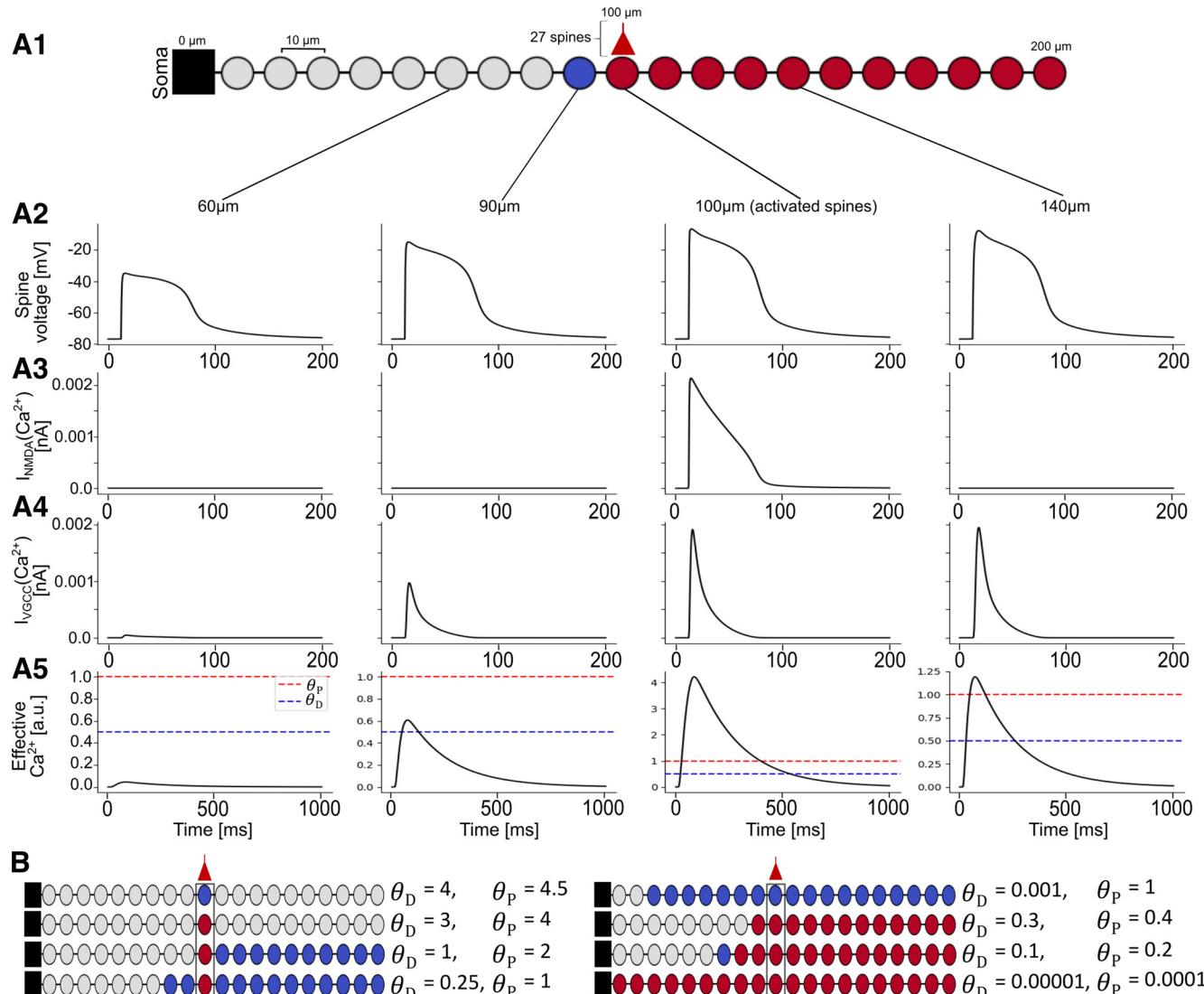


Figure 3. Asymmetric heterosynaptic plasticity induced by VGCCs. **A1**, Top, A ball-and-stick model dendrite with spines (circles) placed every 10 μm. A cluster of 27 spines (shown as a single circle with an input) is activated at the center of the dendrite (100 μm), generating an NMDA spike that results in homosynaptic and heterosynaptic plasticity. The activated spines and the spines distal to it are potentiated (red), the spine 10 μm proximal to the activated spine is depressed (blue), and the other proximal spines do not change (gray). **A2**, Spine head voltage traces shown at 60 μm, 90 μm, 100 μm (exemplar activated spine), and 140 μm. The NMDA spike is seen at all spines, but the voltage at the proximal location (60 μm) is substantially attenuated. **A3**, Ca^{2+} current through the NMDA receptor at the 4 depicted spines. Only the activated spine has NMDA current because NMDA receptors are ligand gated. **A4**, Ca^{2+} current through the VGCCs at the depicted spines; Ca^{2+} current depends on local voltage (shown in **A2**). **A5**, Effective $[\text{Ca}^{2+}]$ at the depicted spines. At 60 μm, the $[\text{Ca}^{2+}]$ is below θ_D (blue dashed line), so no change occurs; at 90 μm, the $[\text{Ca}^{2+}]$ reaches above θ_D but below θ_P (red dashed line), so depression occurs; at 100 and 140 μm, the $[\text{Ca}^{2+}]$ reaches above θ_P , so the synapses are potentiated. **B**, As in **A1** (27 synapses activated at 100 μm) but with different calcium thresholds for plasticity, resulting in different heterosynaptic effects.

act as a current “sink.” This gives rise to a strong asymmetry of voltage attenuation in dendrites (Rall and Rinzel, 1973).

To demonstrate the effects of dendritic location on voltage attenuation, we created a ball-and-stick cable model with a 200-μm-long cylindrical cable coupled to an isopotential soma (Fig. 2A). We enlarged the diameter of the soma to replicate the electrical sink effect that would occur in a layer 5 cortical pyramidal neuron with a full dendritic morphology (Hay et al., 2011; see Materials and

Methods; Extended Data Fig. 2–1). We placed a cluster of five dendritic spines with excitatory synapses at 100 μm from the soma. We also placed nonactivated spines at 60 and 200 μm from the soma. We simultaneously activated all synapses in the spine cluster located at 100 μm from the soma and recorded the local voltage at the soma, the dendrite, and at the heads of both the activated and non-activated spines. Voltage attenuated slightly from the heads of the activated spines to the spine base, but almost no attenuation was visible from the base of the

activated spines at 100 μm to the head of the distal non-active spine at 200 μm . By contrast, the voltage attenuated substantially from the base of the activated spines to the base of the proximal nonactive spine (Fig. 2B; Segev and Rall, 1988). This attenuation profile is also qualitatively maintained when the same experiment is performed in a branch from the original pyramidal neuron model or in a ball-and-stick model with uniform I_h conductance along the dendrite. If I_h conductances are scaled exponentially along the dendrite, more attenuation may be observed in the distal direction than in the passive cable case (Extended Data Fig. 2-2). Qualitatively, substantial voltage attenuation from spine to dendrite but not from dendrite to spine, is consistent with recent experimental work (Cornejo et al., 2021).

We replicated this experiment with a cluster of 18 synapses at 100 μm from the soma, which was sufficient to generate an NMDA spike in these spines (Eyal et al., 2018). [We note that it is possible to create an NMDA spike with fewer clustered synapses if the synapses are activated at a high frequency (Polsky et al., 2009; Dembrow and Spain, 2022); however, in this work, for simplicity we only vary the cluster size.] The same asymmetric effect as described above was qualitatively observed for the NMDA spike; voltage attenuation was very minor from the activation site to the distal tip, and very substantial from the activation site toward the soma (Fig. 2B,C).

We next demonstrated how the asymmetric attenuation manifests in a branching dendrite model. We created an order-3 branching dendritic tree (see Materials and Methods) and we simultaneously activated a cluster of 18 synapses in the center of a second-order branch, generating a local NMDA spike there. The NMDA spike propagated to the distal daughter branches with minimal attenuation, propagated to the sister branches of the activated branch and their daughter branches with mild attenuation, and propagated to the rest of the dendritic tree with substantial attenuation (Fig. 2D). This same attenuation profile is qualitatively observed if the 18 synapses are not clustered at the exact same location but instead are uniformly dispersed along the 50 μm branch section (Extended Data Fig. 2-3) or if the branching structure of the dendritic tree is asymmetric (Extended Data Fig. 2-4). The stark contrast in the depolarization magnitude of different regions of the dendritic tree in response to a local NMDA spike raises the possibility that asymmetric voltage attenuation may play a functional role in governing plasticity processes in different parts of the dendritic tree.

Asymmetric voltage attenuation produces asymmetric heterosynaptic plasticity

To explore how asymmetric voltage attenuation can impact heterosynaptic plasticity, we placed a spine at each segment of the ball-and-stick model shown in Figure 3A1 (one spine every 10 μm) and activated a cluster of 27 synapses at the center of the dendrite. This produced a large NMDA spike at the activated spines, depolarizing the dendrite sufficiently to open VGCCs at nonactivated spines (Fig. 3A4).

At the active site, both NMDA channels and VGCCs provided a substantial amount of calcium current into the cell, allowing $[\text{Ca}^{2+}]$ to surpass θ_P , generating homosynaptic potentiation. The depolarization spreading from the activated spines was sufficient to open VGCCs at distal spines, providing enough calcium current to induce heterosynaptic potentiation. However, at a spine located 10 μm proximal to the input site, the voltage had already attenuated sufficiently such that the $[\text{Ca}^{2+}]$ there following the opening of the local VGCCs induced depression (Fig. 3A1,A5). At 20 μm proximal to the input site, the voltage attenuated such that the $[\text{Ca}^{2+}]$ from the VGCCs was insufficient to cross θ_D , so the synapses on this spine and all other proximal spines were left unchanged (Fig. 3A).

Of course, the specific plasticity outcomes that we observed hold true only for the specific calcium thresholds for plasticity used in our simulation. We therefore performed the same experiment with a variety of different values for the plasticity thresholds. We observe that it is generally easier to induce heterosynaptic plasticity at spines that are distal to the activation site rather than proximal to it (Fig. 3B).

This directional asymmetry in heterosynaptic effects is especially pronounced when considering a branching dendrite, as asymmetric attenuation from an input site can create branch-dependent dendritic depolarization (Rall and Rinzel, 1973). To demonstrate this effect, we placed spines every 10 μm on the order-3 branched model described above. We activated clusters of 20, 30, 40, or 50 synapses at the each branching level and observed the plastic changes at all modeled spines (Fig. 4).

If the activated synaptic cluster is placed on the zero-order branch emerging from the soma, because of the relatively low input resistance there, it is difficult to induce an NMDA spike or homosynaptic plasticity (30 spines are required for depression, 50 are required for potentiation). Even 50 synapses are insufficient in this model to produce an NMDA spike at the most proximal branch, and thus heterosynaptic plasticity is not induced (Fig. 4A1). If we move the cluster up to a first-order branch, homosynaptic potentiation is induced with 20 synapses, and an NMDA spike is elicited with 30 synapses. When 40 spines are activated, spines on the dendritic tree distal to the input site are depressed, and when 50 spines are activated, these distal synapses are potentiated (Fig. 4A2). When placed at a second-order branch, 20 activated synapses are already sufficient to produce an NMDA spike and heterosynaptic depression at dendritic spines distal to the input site, and 30 spines turns the heterosynaptic depression to potentiation (Fig. 4A3). At the third-order branch, the input resistance is sufficiently large to give rise to both homosynaptic and heterosynaptic potentiation at spines distal to the input site with 20 synapses, and with 30 synapses the voltage propagates sufficiently in the proximal direction to depress the sister branches of the input site. The fact that more input was necessary to produce an NMDA spike at proximal locations than distal locations is consistent with experimental findings that distal dendritic locations tend to integrate their input more nonlinearly than proximal

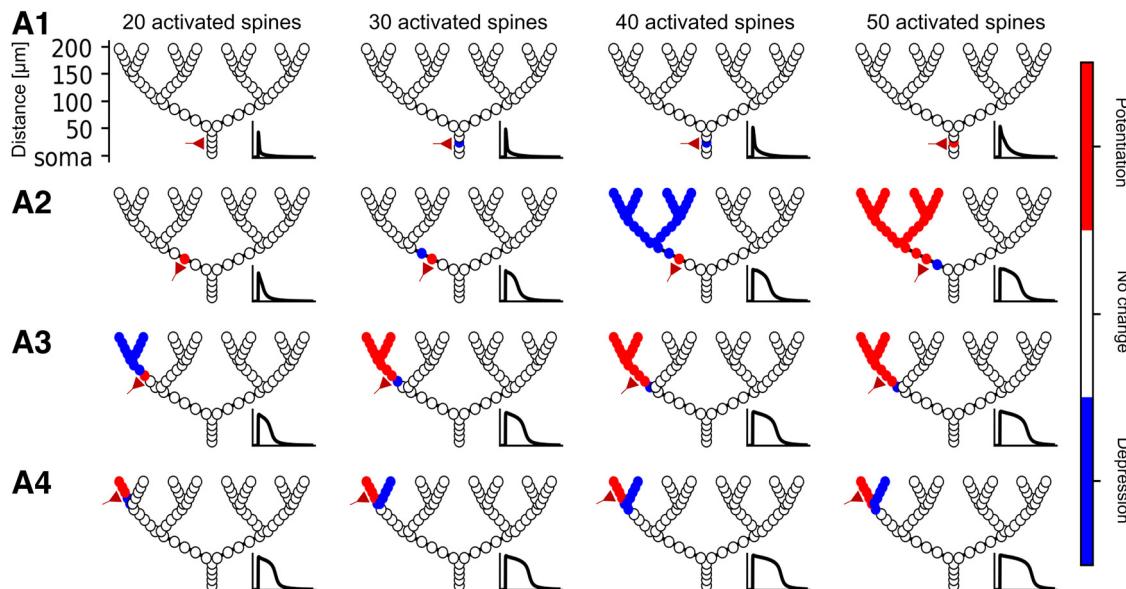


Figure 4. Hierarchical heterosynaptic plasticity in a branching dendritic model. **A1**, A synaptic cluster of 20 (first column), 30 (second column), 40 (third column), or 50 (fourth column) spines are simultaneously activated at the indicated location (schematic red synapse) on the proximal zero-order branch. Insets, Spine head voltage at the activated sites. Because of the low input resistance at the proximal branch, 20 synapses are insufficient to generate homosynaptic plasticity, while 30 and 40 synapses generate local homosynaptic depression (blue circles), and 50 synapses generate homosynaptic potentiation (red circles). **A2**, Same as **A1**, but the activated cluster is now at the first-order branch. Twenty activated synapses create homosynaptic potentiation, 30 or 40 activated synapses create a short-duration NMDA spike and heterosynaptic depression at synapses distal to the activation site, whereas 50 activated synapses create a prolonged NMDA spike and heterosynaptic potentiation at distal sites. **A3**, Activated cluster at the second-order branch; 20 activated synapses cause a short NMDA spike and distal heterosynaptic depression; ≥ 30 synapses create a prolonged NMDA spike and heterosynaptic potentiation at distal synapses. **A4**, Activated cluster at third-order branch. Twenty activated synapses are sufficient to cause distal heterosynaptic potentiation, and ≥ 30 activated synapses also produce heterosynaptic depression at the input site.

locations (Branco and Häusser, 2011). The tiered nature of heterosynaptic plasticity in dendrites, where proximal inputs can induce heterosynaptic plasticity at branches that are distal to it, suggests that dendritic branches might supervise each other in a hierarchical manner. Branches that are closer to the soma (although not so close that it is difficult to generate an NMDA spike) can “teach” the branches that are distal to it because the NMDA spike preferentially propagates backward toward distal locations, leading to heterosynaptic potentiation or depression in descendant branches. Moreover, distal branches with high input resistances may be able to supervise plasticity in their sibling branches via a competitive process where an input sufficient to depress the branch with homosynaptic input will heterosynaptically depress synapses on a sibling branch.

Synergistic synaptic “sandwiching”

Until now, we have only looked at heterosynaptic effects produced by the activation of a single cluster of co-localized spines, generating a single local NMDA spike. It is possible that multiple clusters can be activated simultaneously, generating diverse depolarization effects in the dendritic tree (Palmer et al., 2014). From a plasticity standpoint, it is important to think about how the clusters of activated synapses can affect each other (through both

VGCC-dependent and NMDA-dependent activations) as well as how they affect inactive synapses via heterosynaptic plasticity (through VGCC activation). While it is not feasible to explore the full combinatorial space of cluster activations, we consider a canonical case in our ball-and-stick model where an inactive spine, placed 130 μm from the soma, is “sandwiched” in between two spine clusters, one proximal (60 μm from the soma) and one distal (200 μm from the soma; Fig. 5A). This case is important for understanding the mechanisms governing heterosynaptic plasticity because it illustrates the trade-off between two principles. On the one hand, voltage attenuates more steeply toward the soma. On the other hand, it is easier to generate a large/prolonged NMDA spike at distal synapses, because of the higher input resistance at distal locations (Poleg-Polsky, 2015; Doron et al., 2017).

To illustrate this trade-off, suppose we have 40 active synapses to distribute between the proximal cluster and the distal cluster with the goal of maximizing the depolarization, and thus the heterosynaptic calcium influx, at the centrally located inactive synapse. If the input resistance effect dominates, it would be better to place all synapses distally. If the asymmetric voltage attenuation effect dominates, we might assign all 40 synapses to the proximal cluster. In fact, however, it seems that the answer lies in between these two extremes: placing 20 synapses each

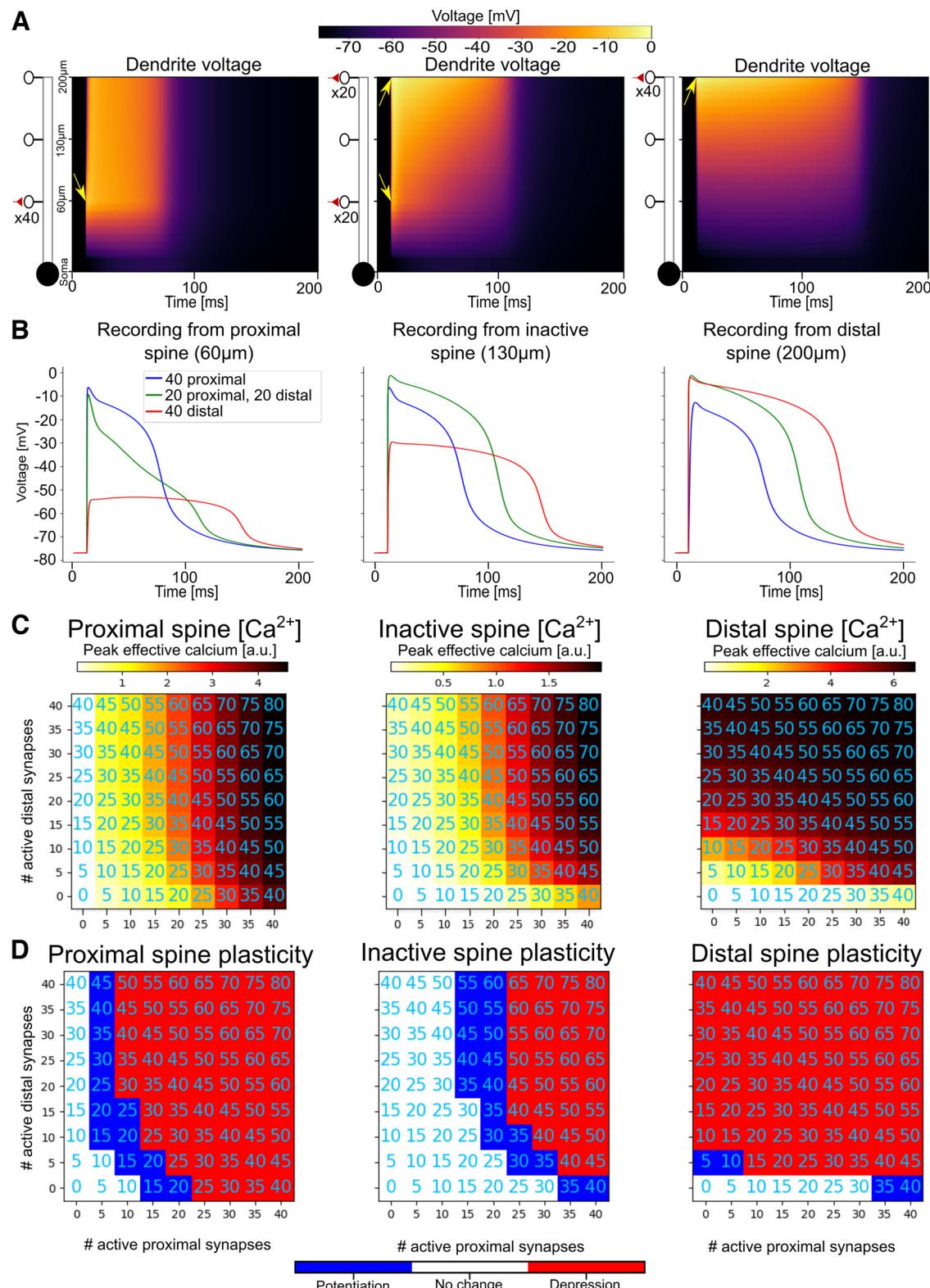


Figure 5. Proximal and distal clusters can create synergistic heterosynaptic effects for “sandwiched” synapses. **A**, Dendritic voltage over time in a ball-and-stick model (schematic at left) when 40 synapses are placed proximally (left) or distally (right), or are distributed evenly between the proximal and distal locations (center). Arrows show the location and time of the activated synapses. **B**, Spine head voltage recordings from a proximal activated spine (left), the central nonactivated spine (center), and a distal activated spine (right) for the 3 cases shown in **A**. **C**, Peak calcium influx at the proximal (left), central (middle), or distal spine (right) as a heatmap for different numbers of spines placed in the distal and proximal clusters. Annotations indicate total number of activated spines (proximal + distal). **D**, Plastic effect (red, potentiation; blue, depression; white, no change) resulting from the calcium influx shown in **C**.

at the proximal and distal locations results in a slightly larger depolarization at the heterosynaptic synapse than placing all 40 synapses together in a single cluster at either the proximal or distal location (Fig. 5A,B). The synergy between distal and proximal clusters is not restricted to the case of 40 synapses; for any given number of synapses there appears to be a “sweet spot” for distributing those synapses between the proximal and distal locations to maximize heterosynaptic effects at the central location, albeit with a tendency to assign more synapses to the proximal location (Fig. 5C,D).

The increased depolarization when the synapses are divided into separate clusters can be explained by the fact that there are diminishing returns for placing additional synapses at the same location because of the reduced driving force when the dendrite is depolarized to near its reversal potential. It is thus better to separate the synapses into separate clusters at locations that are somewhat electrically separated to avoid “wasting” synapses on a dendritic segment that is already maximally depolarized. (Additional synapses can still increase the duration of an NMDA spike when the branch is depolarized to near its electrical reversal; however, for the purposes of heterosynaptic plasticity, it is often crucial to maximize the peak voltage at the heterosynaptic synapse to ensure that the peak calcium through the VGCCs passes the plasticity thresholds). The duration above the plasticity thresholds also affects the magnitude of the plastic changes in the early phase of long-term plasticity, but in the bistable Graupner–Brunel model, magnitude information is lost after several hours in the late phase of plasticity when the synaptic weights are stabilized into a binary UP/DOWN state (Graupner and Brunel, 2012).

The benefit of dividing synapses into two groups is not observed at the proximal and distal locations themselves. While active proximal synapses do increase calcium influx at distal synapses and vice versa (because of both NMDA and VGCC voltage dependence), to maximize peak calcium influx at proximal spines it is best to put all the synapses proximally, and to maximize peak calcium influx at distal spines it is best to put all the synapses distally (Fig. 5C,D).

The synergistic heterosynaptic sandwiching effect also pertains in a branched neuron model. We placed varying numbers of activated spines on a proximal (first-order) branch and a distal (third-order) branch in our four-layer branched model to observe the heterosynaptic effects at a nonactivated spine on the central (second-order) branch. As in the ball-and-stick model, the peak calcium at the nonactive, central spine was maximized when active synapses were distributed between the proximal and distal branch (Fig. 6A–C, Extended Data Fig. 6-1).

In addition to this “vertical sandwiching” scenario, we also explore a “horizontal sandwiching” case, where an inactive spine is placed in the middle of a branch at the second branching layer, and varying numbers of active synapses are placed at its left and right daughter branches at the third branching layer. We again observe in this context that, from the perspective of the inactive

spine on the proximal parent branch, dividing the active spines between the left and right daughter branches tends to maximize the peak calcium available for producing heterosynaptic plasticity. As we would expect from the symmetry of the left and right branches relative to the parent branch, the peak heterosynaptic calcium tends to be maximal when the left and right branches have the same number of activated spines (Fig. 6D–F, Extended Data Fig. 6-2).

We have thus shown that when an inactive synapse is placed between two synaptic clusters, whether it is vertically sandwiched between a distal and proximal branch or horizontally sandwiched between two of its daughter branches, plasticity-inducing calcium influx tends to be greater than if all the active synapses were placed in a single cluster. This raises the possibility that in addition to the hierarchical supervision effect we showed above, it may be possible to engineer synapse placement in a sophisticated manner to maximize heterosynaptic plasticity induction without requiring an excessive number of active synapses at the same location.

Discussion

Our simulations have shown a wide range of consequences for synaptic plasticity arising from the hypothesis that heterosynaptic plasticity might result from dendritic depolarization-induced calcium influx through VGCCs. Simple dendritic cable models, combined with model synapses containing NMDA and VGCCs channels, were sufficient to produce spatially sensitive heterosynaptic plasticity effects using a standard calcium-based plasticity mechanism. Specifically, we have demonstrated that a strong dendritic input that generates an NMDA spike can induce heterosynaptic plasticity at dendritic sites that are distal to the input because of asymmetric voltage attenuation in dendrites. This asymmetry can create a hierarchical heterosynaptic effect in a branching dendrite structure, whereby clustered inputs to a branch closer to the soma can act as “supervisors” to synapses located on more distal branches. Moreover, when two input clusters are active, each cluster can increase the plasticity-inducing calcium influx at spines in the other cluster, as well as at nonactive spines. Additionally, calcium influx to a nonactive spine can be maximized by dividing activated spines into two clusters, rather than placing all activated spines at the same location.

The extent to which these phenomena occur in biology remains an open question, and we encourage experimentalists to use the predictions of our model to design experiments to test whether hierarchical heterosynaptic plastic effects indeed occur in the brain. If our predictions are borne out by experiments, then heterosynaptic plasticity can produce a richer repertoire of plastic effects than have been previously considered. If dendritic NMDA spikes indeed act as heterosynaptic supervisors for other (more distal) synapses, the dendritic branching structure and the location of NMDA spike induction become essential for plasticity induction. Location-sensitive NMDA spike-dependent plasticity rules are particularly critical in light of findings that backpropagating somatic action potentials

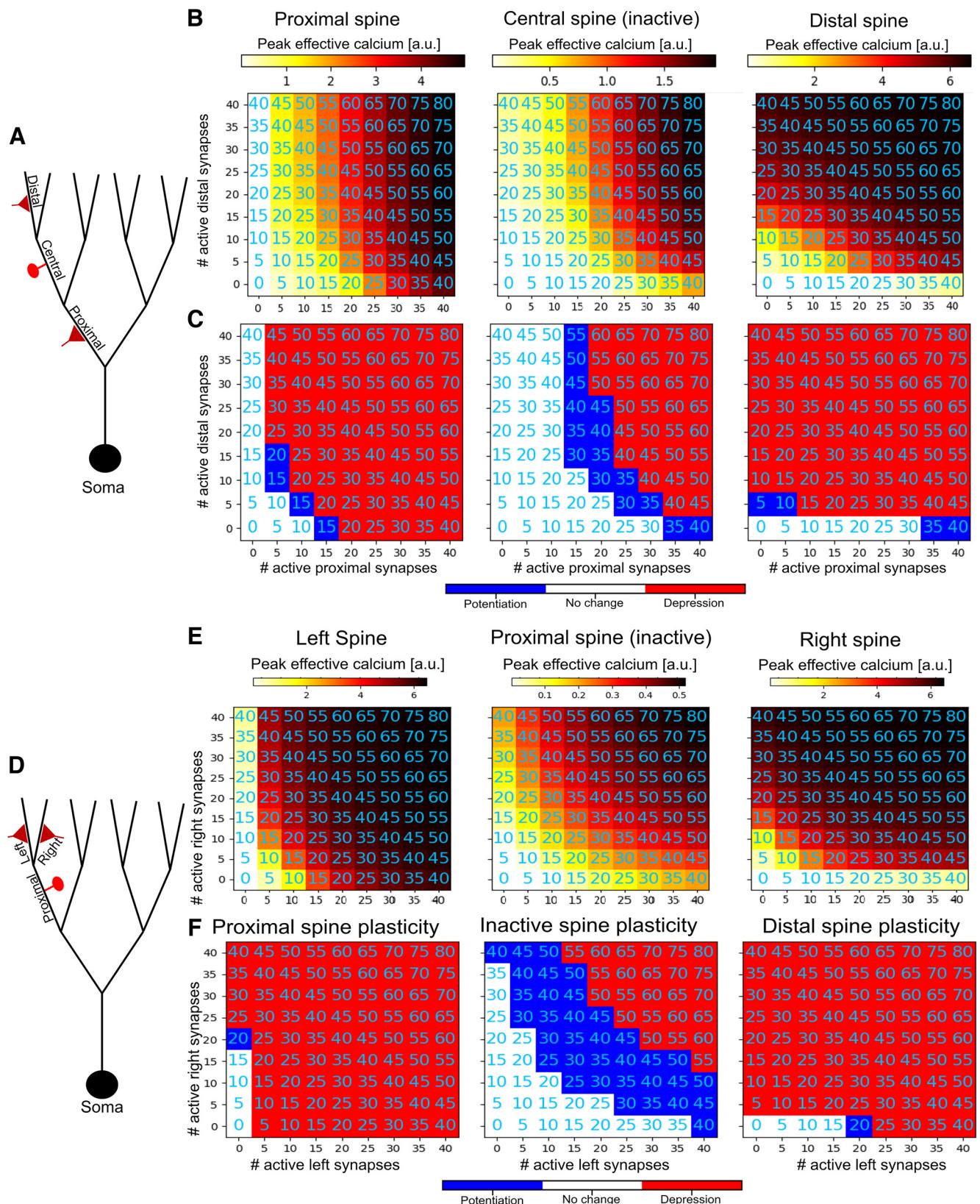


Figure 6. Vertical and horizontal heterosynaptic sandwiching in a branched dendritic model. **A**, Experiment schematic for vertical sandwiching in a branched model. Clusters of spines are activated at proximal (first-order) and distal (third-order) branches to explore their effect on each other and on a nonactivated spine located at a “central” branch (second order) in between them. **B**, Peak calcium concentration (in A.U.) at an exemplar spine on the proximal (left), central (middle), or distal (right) branches. **C**, Plastic effect

continued

on each spine as a function of cluster sizes (red, potentiation; blue, depression; white, no change). See Extended Data Figure 6-1 for voltage traces and heatmaps. **D**, Experiment schematic for horizontal sandwiching in a branched model. Clusters of spines are activated at third-order left and right branches to explore how clusters on sister branches affect each other as well as a nonactivated synapse on their parent (second-order) branch. **E, F**, Calcium and plasticity for the horizontal sandwiching experiment as in **C** and **D**. See Extended Data Figure 6-2 for voltage traces and heatmaps.

may not reach distal synapses, and therefore do not induce plasticity in these synapses, whereas NMDA spikes can induce plasticity there (Kumar et al., 2021; see also Lisman and Spruston, 2005; Hardie and Spruston, 2009).

Further work can explore diverse neuronal types with different dendritic morphologies to examine whether the branching structure of different neurons may lend themselves to different kinds of plasticity computations. For example, the elaborate fractal branching structure of Purkinje neurons may lend those neurons to be optimized for segregated hierarchical units (Piochon et al., 2007, 2010). (See (Liu et al., 2016) regarding the presence of NMDARs in Purkinje neurons and their influence on plasticity). Conversely, neurons with sufficiently long, branching dendrites (e.g., apical dendrites of L2/3 cells) may exhibit more attenuation in the proximal-to-distal direction and thus behave less hierarchically, as sufficiently long distal branches themselves can act as electrical sinks relative to the parent branch (Landau et al., 2022).

The branch-dependent variation of heterosynaptic plasticity we show in our model is in line with the theory that the dendritic branch may be a fundamental computational unit in the neuron (Koch et al., 1983; Segev and Rall, 1998; Branco and Häusser, 2010). Consistent with the idea that neurons can behave as a two-layer and even a multiple-layer neural network (Poirazi and Mel, 2001; Poirazi et al., 2003; Beniaguev et al., 2021), hierarchical plasticity can potentially serve as a biophysical basis for a multilayer learning algorithm within a single neuron, perhaps akin to the backpropagation algorithm in deep neural networks in feedforward artificial neural networks (Rumelhart et al., 1986; Jones and Kording, 2021). The details of how such an algorithm would operate remain an open avenue for investigation.

One way to make use of these hierarchical and branch-level plastic effects is to take advantage of extant structural input specificity. For example, if a multimodal integration neuron receives auditory input on one branch and visual input on a different branch, a single supervisory NMDA spike can induce the potentiation of all auditory inputs simultaneously, which could be useful in any situation where auditory inputs should be weighted preferentially (e.g., in a low-light environment). If the more distal branching levels also contain additional levels of input specificity (e.g., the visually sensitive dendritic branch might bifurcate into branches sensitive to Gabor-like filters of different frequencies or orientations), a supervisory signal sent to one of these branches might be able to similarly treat one category of inputs preferentially. It should be noted, however, that the assumption that different kinds of sensory inputs make synapses on different dendritic branches is speculative, and some evidence points to a more random, “salt-and-pepper” distribution (Jia et al., 2010).

An alternative possibility is that instead of assuming that each branch receives different categories of inputs, dendritic branches may be sensitive to input features that are correlated in the real world (e.g., the sound of crying with the image of an infant). The hypothesis that correlated inputs are mapped to nearby locations on the dendrite is known as synaptic clustering and has been explored in various theoretical and experimental studies (Mel, 1991, 1992; Kleindienst et al., 2011; Kastellakis et al., 2015; Moldwin et al., 2021). The branch-level plasticity we describe here can operate on these clustered input features to treat the set of clustered inputs at the branch level as a single unit for the purposes of plasticity.

The hierarchical plasticity phenomenon as suggested here is complicated somewhat by our sandwiching results, which demonstrate that, for a given number of activated synapses, heterosynaptic effects can be maximized by distributing them into two (or possibly more) spatially segregated clusters instead of placing them all at the same location. This points to the possibility of an even more sophisticated supervision scheme, where multiple synaptic clusters can be strategically placed at different dendritic locations to produce spatially targeted heterosynaptic plasticity. Spatiotemporally targeted inhibition may also help shape the spread of heterosynaptic plasticity. Further experimental and theoretical work could explore these possibilities in more detail. In any event, the diverse heterosynaptic effects we have shown here provide support for the claim that neurons may behave as complex nonlinear units (Koch and Segev, 2000; Poirazi et al., 2003; Beniaguev et al., 2021; Jones and Kording, 2021; Larkum, 2022) as opposed to simple perceptrons where synapses are modified independently (Moldwin and Segev, 2020). Moreover, the pronounced asymmetrical voltage attenuation in dendrites and the attendant consequences for heterosynaptic plasticity shown in our simulations indicate that computational models that make use of distance-dependent NMDA superlinearities (Mel, 1991; Moldwin et al., 2021) should take into account branching structure and synaptic location relative to the soma in addition to the relative distance of synapses from each other.

Our model, in line with the proposal of Lisman (2001), assumes that the only medium of communication between active and inactive synapses is dendritic voltage depolarization, which can activate VGCCs of other non-activated dendritic spines. We note that many other mechanisms for the induction of heterosynaptic plasticity have been suggested (for review, see Chistiakova et al., 2014; Chater and Goda, 2021). One alternative possibility is that calcium itself diffuses from one synapse to another; however, experimental evidence suggests that calcium diffusion from the spine head into the dendritic

shaft is negligible (Yuste and Denk, 1995; Sabatini et al., 2002). Other molecules have also been implicated in inducing heterosynaptic effects, such as h-Ras, Rac1, RhoA, Arc, BDNF-TrkB, CaMKII, and calcineurin (Chater and Goda, 2021; Tong et al., 2021); however, these molecules have only been shown to diffuse up to 10 μm along the dendrite, while heterosynaptic effects have been shown to occur at much larger distances between activated and nonactivated synapses (Lynch et al., 1977; Engert and Bonhoeffer, 1997). As such, the depolarization-based model remains an important candidate mechanism of heterosynaptic plasticity. It may be that there are different short-distance and long-distance heterosynaptic effects, with short-distance effects occurring via molecular mechanisms such as local CamKII and calcineurin activity, while long-distance effects may be because of the voltage mechanism we describe here.

Regarding the fidelity of the parameters in our simulation to biological reality, there are several questions that would require additional experimental evidence and more detailed models to fully confirm. Our calcium channel model assumed a single type of calcium channel, and we chose a conductance value that approximately corresponds to what we might expect as the aggregate conductance of all high-voltage activated VGCCs. A more precise model that includes all forms of VGCCs with their appropriate unitary conductances, kinetics, and densities would allow us to examine our claims with greater precision. Additionally, the kinetics of calcium accumulation and the plasticity thresholds for calcium used here could be better constrained with more experimental evidence. We also assumed that calcium channel density and plasticity thresholds were the same from spine to spine; in biology, these may differ on a spine-by-spine basis even with a single neuron. Moreover, the spatial effects we observed in our simulations assume a passive dendritic cable; active mechanisms in biological dendrites such as voltage-gated sodium, calcium, and potassium channels have been shown to differentially modulate voltage propagation in different neurons (Golding et al., 2001), so these mechanisms would consequently be expected to modify the spatial dynamics of heterosynaptic plasticity as well.

Inhibitory synapses also likely play an important role in the spatial reach of heterosynaptic plasticity. Inhibition can have different consequences for the neuronal voltage depending on the location of the inhibitory synapses (Gidon and Segev, 2012; Jadi et al., 2012, 2014; Bar-Ilan et al., 2013) as well as their timing relative to excitatory NMDA inputs (Doron et al., 2017). As such, spatiotemporally targeted inhibition can be used to modulate the heterosynaptic effects we describe here, enabling a bidirectional control system for heterosynaptic plasticity.

Several studies have shown that internal calcium stores play an important role in both homosynaptic and heterosynaptic plasticity (Nishiyama et al., 2000; Rose and Konnerth, 2001; Royer and Paré, 2003; Jo et al., 2008; Camiré and Topolnik, 2014; Evans and Blackwell, 2015; O'Hare et al., 2022). Ryanodine and IP₃ receptors produce calcium-induced calcium release (CICR), which can affect plasticity in a variety of ways. Although experimental results regarding the role of CICR are more subtle

than the model we have presented here, one way to think about CICR is as an amplifier of calcium coming from NMDA and voltage-gated calcium channels. As such, assuming that CICR increases monotonically with calcium from extracellular sources, the basic qualitative principle that activated synapses will experience more calcium release than nonactivated synapses because of NMDA receptor activation still holds, but it may shift the homosynaptic and heterosynaptic effects observed (e.g., from homosynaptic depression and no heterosynaptic effect to homosynaptic potentiation and heterosynaptic depression; Fig. 1C). However, there is evidence that the effect of internal calcium stores is highly localized into microdomains and depends on various second messengers, resulting in a more complex picture of homosynaptic and heterosynaptic effects (Evans and Blackwell, 2015).

Another biological mechanism that may affect the plastic results we predict here are small-conductance Ca²⁺-activated K⁺ channels (SK channels). SK channels can repolarize the membrane in response to calcium influx (Adelman et al., 2012; Tigaret et al., 2016; Rodrigues et al., 2021), potentially reducing both homosynaptic and heterosynaptic effects.

One additional crucial biological question is whether the early stage heterosynaptic plasticity induced by calcium influx is stabilized into late-term plasticity via protein synthesis, which has been shown to be necessary to make plastic changes last longer than an hour (Frey and Morris, 1997; Barco et al., 2008; Redondo et al., 2010). One recent study (Sun et al., 2021) showed that plasticity-induced protein synthesis may primarily occur within 3 μm of potentiated synapses, suggesting that heterosynaptic effects may not necessarily be long lived. However, it is possible that a very strong clustered stimulation, such as we described here, may induce protein synthesis at more distant locations.

References

- Abraham WC, Bear MF (1996) Metaplasticity: the plasticity of synaptic plasticity. *Trends Neurosci* 19:126–130.
- Adelman JP, Maylie J, Sah P (2012) Small-conductance Ca²⁺-activated K⁺ channels: form and function. *Annu Rev Physiol* 74:245–269.
- Araya R, Eisenthal KB, Yuste R (2006) Dendritic spines linearize the summation of excitatory potentials. *Proc Natl Acad Sci U S A* 103:18799–18804.
- Arellano JL, Benavides-Piccione R, DeFelipe J, Yuste R (2007) Ultrastructure of dendritic spines: correlation between synaptic and spine morphologies. *Front Neurosci* 1:131–143.
- Artola A, Bröcher S, Singer W (1990) Different voltage-dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex. *Nature* 347:69–72.
- Barco A, Lopez de Armentia M, Alarcon JM (2008) Synapse-specific stabilization of plasticity processes: the synaptic tagging and capture hypothesis revisited 10 years later. *Neurosci Biobehav Rev* 32:831–851.
- Bar-Ilan L, Gidon A, Segev I (2013) The role of dendritic inhibition in shaping the plasticity of excitatory synapses. *Front Neural Circuits* 6:118.
- Behabadi BF, Polsky A, Jadi M, Schiller J, Mel BW (2012) Location-dependent excitatory synaptic interactions in pyramidal neuron dendrites. *PLoS Comput Biol* 8:e1002599.

- Beniaguev D, Segev I, London M (2021) Single cortical neurons as deep artificial neural networks. *Neuron* 109:2727–2739.e3.
- Bi G, Poo M (1998) Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J Neurosci* 18:10464–10472.
- Bliss T, Collingridge G (1993) A synaptic model of memory: long-term potentiation in the hippocampus. *Nature* 361:31–39.
- Bliss T, Lomo T (1973) Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path. *J Physiol* 232:331–356.
- Branco T, Häusser M (2010) The single dendritic branch as a fundamental functional unit in the nervous system. *Curr Opin Neurobiol* 20:494–502.
- Branco T, Häusser M (2011) Synaptic integration gradients in single cortical pyramidal cell dendrites. *Neuron* 69:885–892.
- Camiré O, Topolnik L (2014) Dendritic calcium nonlinearities switch the direction of synaptic plasticity in fast-spiking interneurons. *J Neurosci* 34:3864–3877.
- Chater TE, Goda Y (2021) My Neighbour Hetero—deconstructing the mechanisms underlying heterosynaptic plasticity. *Curr Opin Neurobiol* 67:106–114.
- Chindemi G, Abdellah M, Amsalem O, Benavides-Piccione R, Delattre V, Doron M, Ecker A, Jaquier AT, King J, Kumbhar P, Monney C, Perin R, Rössert C, Tuncel AM, Van Geit W, DeFelipe J, Graupner M, Segev I, Markram H, Muller EB (2022) A calcium-based plasticity model for predicting long-term potentiation and depression in the neocortex. *Nat Commun* 13:3038.
- Chistiakova M, Bannon NM, Bazhenov M, Volgushev M (2014) Heterosynaptic plasticity. *Neuroscientist* 20:483–498.
- Cho K, Aggleton JP, Brown MW, Bashir ZI (2001) An experimental test of the role of postsynaptic calcium levels in determining synaptic strength using perirhinal cortex of rat. *J Physiol* 532:459–466.
- Coesmans M, Weber JT, De Zeeuw Cl, Hansel C (2004) Bidirectional parallel fiber plasticity in the cerebellum under climbing fiber control. *Neuron* 44:691–700.
- Cornejo VH, Ofer N, Yuste R (2021) Voltage compartmentalization in dendritic spines in vivo. *Science* 375:82–86.
- Cummings JA, Mulkey RM, Nicoll RA, Malenka RC (1996) Ca²⁺ signaling requirements for long-term depression in the hippocampus. *Neuron* 16:825–833.
- Dembrow NC, Spain WJ (2022) Input rate encoding and gain control in dendrites of neocortical pyramidal neurons. *Cell Rep* 38:110382.
- Doron M, Chindemi G, Muller E, Markram H, Segev I, Eilif Muller HM, Segev I, Muller E, Markram H, Segev I, Eilif Muller HM, Segev I (2017) Timed synaptic inhibition shapes NMDA spikes, influencing local dendritic processing and global I/O properties of cortical neurons. *Cell Rep* 21:1550–1561.
- Dudek SM, Bear MF (1992) Homosynaptic long-term depression in area CA1 of hippocampus and effects of N-methyl-D-aspartate receptor blockade. *Proc Natl Acad Sci U S A* 89:4363–4367.
- Engert F, Bonhoeffer T (1997) Synapse specificity of long-term potentiation breaks down at short distances. *Nature* 388:279–284.
- Evans R, Blackwell K (2015) Calcium: amplitude, duration, or location? *Biol Bull* 228:75–83.
- Eyal G, Verhoog MB, Testa-Silva G, Deitcher Y, Benavides-Piccione R, DeFelipe J, de Kock CPJ, Mansvelder HD, Segev I (2018) Human cortical pyramidal neurons: from spines to spikes via models. *Front Cell Neurosci* 12:181.
- Fino E, Paille V, Cui Y, Morera-Herreras T, Deniau J-M, Venance L (2010) Distinct coincidence detectors govern the corticostratial spike timing-dependent plasticity. *J Physiol* 588:3045–3062.
- Fisher RE, Gray R, Johnston D (1990) Properties and distribution of single voltage-gated calcium channels in adult hippocampal neurons. *J Neurophysiol* 64:91–104.
- Frey U, Morris RGM (1997) Synaptic tagging and long-term potentiation. *Nature* 385:533–536.
- Gidon A, Segev I (2012) Principles governing the operation of synaptic inhibition in dendrites. *Neuron* 75:330–341.
- Golding NL, Kath WL, Spruston N (2001) Dichotomy of action-potential backpropagation in CA1 pyramidal neuron dendrites. *J Neurophysiol* 86:2998–3010.
- Golding NL, Staff NP, Spruston N (2002) Dendritic spikes as a mechanism for cooperative long-term potentiation. *Nature* 418:326–331.
- Graupner M, Brunel N (2012) Calcium-based plasticity model explains sensitivity of synaptic changes to spike pattern, rate, and dendritic location. *Proc Natl Acad Sci U S A* 109:3991–3996.
- Hardie J, Spruston N (2009) Synaptic depolarization is more effective than back-propagating action potentials during induction of associative long-term potentiation in hippocampal pyramidal neurons. *J Neurosci* 29:3233–3241.
- Hay E, Hill S, Schürmann F, Markram H, Segev I (2011) Models of neocortical layer 5b pyramidal cells capturing a wide range of dendritic and perisomatic active properties. *PLoS Comput Biol* 7: e1002107.
- Hebb D (1949) *The organization of behavior: a neuropsychological theory*. New York: Wiley.
- Hines ML, Carnevale NT (1997) The NEURON simulation environment. *Neural Comput* 12:99–1209.
- Hines ML, Davison AP, Muller E (2009) NEURON and Python. *Front Neuroinform* 3:1.
- Humeau Y, Choquet D (2019) The next generation of approaches to investigate the link between synaptic plasticity and learning. *Nat Neurosci* 22:1536–1543.
- Jadi M, Polksy A, Schiller J, Mel BW (2012) Location-dependent effects of inhibition on local spiking in pyramidal neuron dendrites. *PLoS Comput Biol* 8:e1002550.
- Jadi MP, Behabadi BF, Poleg-Polsky A, Schiller J, Mel BW (2014) An augmented two-layer model captures nonlinear analog spatial integration effects in pyramidal neuron dendrites. *Proc IEEE 102:782–798.*
- Jia H, Rochefort NL, Chen X, Konnerth A (2010) Dendritic organization of sensory input to cortical neurons in vivo. *Nature* 464:1307–1312.
- Jo J, Heon S, Kim MJ, Son GH, Park Y, Henley JM, Weiss JL, Sheng M, Collingridge GL, Cho K (2008) Metabotropic glutamate receptor-mediated LTD involves two interacting Ca²⁺ sensors, NCS-1 and PICK1. *Neuron* 60:1095–1111.
- Jones IS, Kording KP (2021) Might a single neuron solve interesting machine learning problems through successive computations on its dendritic tree? *Neural Comput* 33:1554–1571.
- Kastellakis G, Cai DJ, Mednick SC, Silva AJ, Poirazi P (2015) Synaptic clustering within dendrites: an emerging theory of memory formation. *Prog Neurobiol* 126:19–35.
- Kleindienst T, Winnubst J, Roth-Alpermann C, Bonhoeffer T, Lohmann C (2011) Activity-dependent clustering of functional synaptic inputs on developing hippocampal dendrites. *Neuron* 72:1012–1024.
- Koch C, Segev I (2000) The role of single neurons in information processing. *Nat Neurosci* 3:1171–1177.
- Koch C, Poggio T, Torre V (1983) Nonlinear interactions in a dendritic tree: localization, timing, and role in information processing. *Proc Natl Acad Sci U S A* 80:2799–2802.
- Kole MHP, Hallermann S, Stuart GJ (2006) Single I_h channels in pyramidal neuron dendrites: properties, distribution, and impact on action potential output. *J Neurosci* 26:1677–1687.
- Konur S, Rabinowitz D, Fenstermaker VL, Yuste R (2003) Systematic regulation of spine sizes and densities in pyramidal neurons. *J Neurobiol* 56:95–112.
- Kumar A, Barkai E, Schiller J (2021) Plasticity of olfactory bulb inputs mediated by dendritic NMDA-spikes in rodent piriform cortex. *Elife* 10:e70383.
- Landau AT, Park P, Wong-Campos JD, Tian H, Cohen AE, Sabatini BL (2022) Dendritic branch structure compartmentalizes voltage-dependent calcium influx in cortical layer 2/3 pyramidal cells. *ELife* 11:
- Larkum ME (2022) Are dendrites conceptually useful? *Neuroscience* 489:4–14.

- Lisman J (1989) A mechanism for the Hebb and the anti-Hebb processes underlying learning and memory. *Proc Natl Acad Sci U S A* 86:9574–9578.
- Lisman JE (2001) Three Ca^{2+} levels affect plasticity differently: the LTP zone, the LTD zone and no man's land. *J Physiol* 532:285–285.
- Lisman J, Spruston N (2005) Postsynaptic depolarization requirements for LTP and LTD: a critique of spike timing-dependent plasticity. *Nat Neurosci* 8:839–841.
- Liu H, Lan Y, Bing YH, Chu CP, Qiu DL (2016) N-Methyl-D-aspartate receptors contribute to complex spike signaling in cerebellar Purkinje cells: an *in vivo* study in mice. *Front Cell Neurosci* 10:172.
- Losonczy A, Magee JC (2006) Integrative properties of radial oblique dendrites in hippocampal CA1 pyramidal neurons. *Neuron* 50:291–307.
- Lynch GS, Dunwiddie T, Gribkoff V (1977) Heterosynaptic depression: a postsynaptic correlate of long-term potentiation. *Nature* 266:737–739.
- Malenka RC, Kauer JA, Perkel DJ, Mauk MD, Kelly PT, Nicoll RA, Waxham MN (1989) An essential role for postsynaptic calmodulin and protein kinase activity in long-term potentiation. *Nature* 340:554–557.
- Malinow R, Schulman H, Tsien RW (1989) Inhibition of postsynaptic PKC or CaMKII blocks induction but not expression of LTP. *Science* 245:862–866.
- Mel BW (1991) The clusteron: toward a simple abstraction for a complex neuron. In: *Advances in neural information processing systems*, Vol 4 (Moody J, Hanson S, Lippmann R, eds), pp 35–42. San Mateo, CA: Morgan Kaufmann.
- Mel BW (1992) NMDA-based pattern discrimination in a modeled cortical neuron. *Neural Comput* 4:502–517.
- Moldwin T, Segev I (2020) Perceptron learning and classification in a modeled cortical pyramidal cell. *Front Comput Neurosci* 14:33.
- Moldwin T, Kalmenson M, Segev I (2021) The gradient clusteron: a model neuron that learns to solve classification tasks via dendritic nonlinearities, structural plasticity, and gradient descent. *PLoS Comput Biol* 17:e1009015.
- Mulkey RM, Malenka RC (1992) Mechanisms underlying induction of homosynaptic long-term depression in area CA1 of the hippocampus. *Neuron* 9:967–975.
- Mulkey RM, Herron CE, Malenka RC (1993) An essential role for protein phosphatases in hippocampal long-term depression. *Science* 261:1051–1055.
- Mulkey RM, Endo S, Shenolikar S, Malenka RC (1994) Involvement of a calcineurin/inhibitor-1 phosphatase cascade in hippocampal long-term depression. *Nature* 369:486–488.
- Nabavi S, Fox R, Proulx CD, Lin JY, Tsien RY, Malinow R (2014) Engineering a memory with LTD and LTP. *Nature* 511:348–352.
- Neveu D, Zucker RSRS (1996) Postsynaptic levels of $[\text{Ca}^{2+}]_i$ needed to trigger LTD and LTP. *Neuron* 16:619–629.
- Nishiyama M, Hong K, Mikoshiba K, Poo MM, Kato K (2000) Calcium stores regulate the polarity and input specificity of synaptic modification. *Nature* 408:584–588.
- O'Connor DH, Wittenberg GM, Wang SS-H (2005a) Graded bidirectional synaptic plasticity is composed of switch-like unitary events. *Proc Natl Acad Sci U S A* 102:9679–9684.
- O'Connor DH, Wittenberg GM, Wang SS-H (2005b) Dissection of bi-directional synaptic plasticity into saturable unidirectional processes. *J Neurophysiol* 94:1565–1573.
- O'Hare JK, Gonzalez KC, Herrlinger SA, Hirabayashi Y, Hewitt VL, Blockus H, Szoboszlay M, Rolotti SV, Geiller TC, Negrean A, Chelur V, Polleux F, Losonczy A (2022) Compartment-specific tuning of dendritic feature selectivity by intracellular Ca^{2+} release. *Science* 375:eabm1670.
- Palmer LM, Shai AS, Reeve JE, Anderson HL, Paulsen O, Larkum ME (2014) NMDA spikes enhance action potential generation during sensory input. *Nat Neurosci* 17:383–390.
- Piochon C, Irinopoulou T, Brusciano D, Bailly Y, Mariani J, Levenes C (2007) NMDA receptor contribution to the climbing fiber response in the adult mouse Purkinje cell. *J Neurosci* 27:10797–10809.
- Piochon C, Levenes C, Ohtsuki G, Hansel C (2010) Purkinje cell NMDA receptors assume a key role in synaptic gain control in the mature cerebellum. *J Neurosci* 30:15330–15335.
- Piochon C, Titley HK, Simmons DH, Grasselli G, Elgersma Y, Hansel C (2016) Calcium threshold shift enables frequency-independent control of plasticity by an instructive signal. *Proc Natl Acad Sci U S A* 113:13221–13226.
- Poirazi P, Mel BW (2001) Impact of active dendrites and structural plasticity on the memory capacity of neural tissue. *Neuron* 29:779–796.
- Poirazi P, Brannon T, Mel BW (2003) Pyramidal neuron as two-layer neural network. *Neuron* 37:989–999.
- Poleg-Polsky A (2015) Effects of neural morphology and input distribution on synaptic processing by global and focal NMDA-spikes. *PLoS One* 10:e0140254.
- Polsky A, Mel BW, Schiller J (2004) Computational subunits in thin dendrites of pyramidal cells. *Nat Neurosci* 7:621–627.
- Polsky A, Mel B, Schiller J (2009) Encoding and decoding bursts by NMDA spikes in basal dendrites of layer 5 pyramidal neurons. *J Neurosci* 29:11891–11903.
- Rall W (1967) Distinguishing theoretical synaptic potentials computed for different soma-dendritic distributions of synaptic input. *J Neurophysiol* 30:1138–1168.
- Rall W, Rinzel J (1973) Branch input resistance and steady attenuation for input to one branch of a dendritic neuron model. *Biophys J* 13:648–687.
- Redondo RL, Okuno H, Spooner PA, Frenguelli BG, Bito H, Morris RG (2010) Synaptic tagging and capture: differential role of distinct calcium/calmodulin kinases in protein synthesis-dependent long-term potentiation. *J Neurosci* 30:4981–4989.
- Rodrigues YE, Tigaret CM, Marie H, O'Donnell C, Veltz R (2021) A stochastic model of hippocampal synaptic plasticity with geometrical readout of enzyme dynamics. *BioRxiv*. Advance online publication. Retrieved July 4, 2023. doi:10.1101/2021.03.30.437703.
- Rose CR, Konnerth A (2001) Stores not just for storage: intracellular calcium release and synaptic plasticity. *Neuron* 31:519–522.
- Rose GM, Dunwiddie TV (1986) Induction of hippocampal long-term potentiation using physiologically patterned stimulation. *Neurosci Lett* 69:244–248.
- Royer S, Paré D (2003) Conservation of total synaptic weight through balanced synaptic depression and potentiation. *Nature* 422:518–522.
- Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. *Nature* 323:533–536.
- Sabatini BL, Svoboda K (2000) Analysis of calcium channels in single spines using optical fluctuation analysis. *Nature* 408:589–593.
- Sabatini BL, Oertner TG, Svoboda K (2002) The life cycle of Ca^{2+} ions in dendritic spines. *Neuron* 33:439–452.
- Segev I, Rall W (1988) Computational study of an excitable dendritic spine. *J Neurophysiol* 60:499–523.
- Segev I, Rall W (1998) Excitable dendrites and spines: earlier theoretical insights elucidate recent direct observations. *Trends Neurosci* 21:453–460.
- Shindou T, Ochi-Shindou M, Wickens JR (2011) A Ca^{2+} threshold for induction of spike-timing-dependent depression in the mouse striatum. *J Neurosci* 31:13015–13022.
- Shouval HZ, Bear MF, Cooper LN (2002) A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *Proc Natl Acad Sci U S A* 99:10831–10836.
- Shouval HZ, Wang SS-H, Wittenberg GM (2010) Spike timing dependent plasticity: a consequence of more fundamental learning rules. *Front Comput Neurosci* 4:19.
- Snutch TP, Peloquin J, Mathews E, McRory JE (2013) Molecular properties of voltage-gated calcium channels. In: *Voltage-gated calcium channels* (Zamponi GW, ed), pp 61–94. Amsterdam: Kluwer.
- Sun C, Nold A, Fusco CM, Rangaraju V, Tchumatchenko T, Heilemann M, Schuman EM (2021) The prevalence and specificity

- of local protein synthesis during neuronal synaptic plasticity. *Sci Adv* 7:eabj0790.
- Tigaret CM, Olivo V, Sadowski JHLP, Ashby MC, Mellor JR (2016) Coordinated activation of distinct Ca^{2+} sources and metabotropic glutamate receptors encodes Hebbian synaptic plasticity. *Nat Commun* 7:10289.
- Tong R, Chater TE, Emptage NJ, Goda Y (2021) Heterosynaptic cross-talk of pre- and postsynaptic strengths along segments of dendrites. *Cell Rep* 34:108693.
- Tsien RW, Lipscombe D, Madison DV, Bley KR, Fox AP (1988) Multiple types of neuronal calcium channels and their selective modulation. *Trends Neurosci* 11:431–438.
- White G, Levy WB, Steward O (1988) Evidence that associative interactions between synapses during the induction of long-term potentiation occur within local dendritic domains. *Proc Natl Acad Sci U S A* 85:2368–2372.
- Whitlock J, Heynen A, Shuler M, Bear M (2006) Learning induces long-term potentiation in the hippocampus. *Science* 313:1093–1097.
- Yang S, Tang Y, Zucker RS, Yang S-N, Tang Y-G, Zucker RS (1999). Selective induction of LTP and LTD by postsynaptic $[\text{Ca}^{2+}]_i$ elevation. *J Neurophysiol* 81:781–787.
- Yuste R, Denk W (1995) Dendritic spines as basic functional units of neuronal integration. *Nature* 375:682–684.

1 **A Generalized Framework for the Calcium Control Hypothesis Describes Weight-**
2 **Dependent Synaptic Plasticity**

5 **Toviah Moldwin¹, Li Shay Azran², Idan Segev^{1,3}**

7 ¹Edmond and Lily Safra Center for Brain Sciences ²Faculty of Science, and ³Department of
8 Neurobiology, The Hebrew University of Jerusalem, Jerusalem, Israel

9 **Correspondence:**

10 Toviah Moldwin
11 toviah.moldwin@mail.huji.ac.il

12 **Keywords:** **Synaptic weights, synaptic plasticity, calcium, calcium-based plasticity,**
13 **neural plasticity, protein synthesis, place fields, place cells, BTSP, STDP,**

1

2 Abstract

3 The brain modifies synaptic strengths to store new information via long-term potentiation
 4 (LTP) and long-term depression (LTD). Evidence has mounted that long-term plasticity is
 5 controlled via concentrations of calcium ($[Ca^{2+}]$) in postsynaptic spines. Several mathematical
 6 models describe this phenomenon, including those of Shouval, Bear, and Cooper (SBC)
 7 (Shouval et al., 2002, 2010) and Graupner and Brunel (GB)(Graupner & Brunel, 2012).
 8 Here we suggest a generalized version of the SBC and GB models, based on a fixed point –
 9 learning rate (FPLR) framework, where the synaptic $[Ca^{2+}]$ specifies a fixed point toward
 10 which the synaptic weight approaches asymptotically at a $[Ca^{2+}]$ -dependent rate. The FPLR
 11 framework offers a straightforward phenomenological interpretation of calcium-based
 12 plasticity: *the calcium concentration tells the synaptic weight where it is going and how fast it*
 13 *goes there.* The FPLR framework can flexibly incorporate various experimental findings,
 14 including the existence of multiple regions of $[Ca^{2+}]$ where no plasticity occurs, or plasticity in
 15 cerebellar Purkinje cells, where the directionality of calcium-based synaptic changes is
 16 thought to be reversed relative to cortical and hippocampal neurons. We also suggest a
 17 modeling approach that captures the dependency of late-phase plasticity stabilization on
 18 protein synthesis. We demonstrate that due to the asymptotic, saturating nature of synaptic
 19 changes in the FPLR rule, the result of frequency- and spike-timing-dependent plasticity
 20 protocols are weight-dependent. Finally, we show how the FPLR framework can explain
 21 plateau potential-induced place field formation in hippocampal CA1 neurons, also known as
 22 behavioral time scale plasticity (BTSP).

23 Introduction

24 Since the work of Donald Hebb (Hebb, 1949), it has been believed that the brain
 25 learns via modifying the strengths of synaptic connections between neurons. Decades of
 26 experimental research have shown that synaptic strengths can be increased via a process
 27 known as long-term potentiation (LTP) and decreased via another process known as long
 28 term depression (LTD). Experimentally, there are a variety of protocols that can induce either
 29 potentiation or depression, usually involving stimulating the presynaptic inputs (e.g., via
 30 electrical or optical stimulation of the axon, or glutamate uncaging at the synapse), the
 31 postsynaptic soma (e.g., by electrically inducing subthreshold depolarization or spiking), or
 32 both, for varying frequencies and durations (Shouval et al., 2010).

33 Over the past several decades, plasticity researchers have been drawn to the
 34 possibility that there may be a fundamental molecular process underlying long-term plasticity.
 35 Lisman (J. Lisman, 1989) proposed a calcium-based theory of plasticity, a theory which has
 36 subsequently been extensively validated (Cho et al., 2001; Cummings et al., 1996; J.
 37 Lisman, 1989; Shouval et al., 2002; Yang et al., 1999). In this framework, different
 38 presynaptic and postsynaptic stimulation protocols yield potentiation or depression due to the
 39 calcium concentration ($[Ca^{2+}]$) which is induced at the synapse by the stimulation. Specifically,
 40 if the $[Ca^{2+}]$ is low, no change occurs. If the $[Ca^{2+}]$ rises above a critical threshold for
 41 depression (θ_D), long-term depression (LTD) occurs and the synaptic strength is decreased. If
 42 the $[Ca^{2+}]$ is above the critical threshold for potentiation (θ_P), long-term potentiation (LTP)
 43 occurs and the synaptic strength is increased. It is believed that calcium promotes LTP via
 44 pathways involving protein kinases such as calmodulin kinase (CaMKII) (J. Lisman, 1989;
 45 Malenka et al., 1989; Neveu & Zucker, 1996; R et al., 1989), while promoting LTD via
 46 phosphatases such as calcineurin (J. Lisman, 1989; Mulkey et al., 1994; RM et al.,
 47 1993).

48 The success of the calcium theory of plasticity has attracted theoretical
 49 neuroscientists and modelers to attempt to capture the dynamics of calcium-based plasticity
 50 in equations and computer models. There are two popular models of calcium-based plasticity
 51 that are currently in use: the models of Shouval, Bear, and Cooper (SBC)(Shouval et al.,
 52 2002, 2010) and Graupner and Brunel (GB)(Graupner & Brunel, 2012). There are several
 53 differences between these models, and we will explore them at length. Fundamentally,
 54 though, both models attempt to capture 3 essential properties:

- 1 1) Synaptic weights decrease when the synaptic $[Ca^{2+}]$ is in the depressive calcium
 2 region ($\theta_D \leq [Ca^{2+}] \leq \theta_P$) and potentiate when the $[Ca^{2+}]$ is in the potentiative
 3 calcium range ($[Ca^{2+}] > \theta_P$).
- 4 2) Synaptic weights do not potentiate or decrease linearly *ad infinitum* in the presence of
 5 potentiating or depressing levels of calcium, but rather they increase or decrease
 6 asymptotically toward some maximum or minimum value. This accounts for the fact
 7 that in biology synaptic weights cannot be arbitrarily large or arbitrarily small
 8 (excitatory synapses can't have negative weights) and the fact that synaptic strengths
 9 are observed to exhibit saturating behavior when undergoing a plasticity protocol
 10 (O'Connor et al., 2005b).
- 11 3) In the pre-depressive region of $[Ca^{2+}]$, neither potentiation or depression occur, but
 12 synaptic strengths may still "drift" toward some pre-determined value(s), called stable
 13 states or fixed points, over a long time scale (i.e. hours or days). This feature is
 14 motivated by evidence that synapses may have a discrete number of stable states
 15 and that synaptic strengths have been experimentally observed to slowly drift toward
 16 stable states after inducing potentiation or depression strengths (Frey & Morris,
 17 1997; Kauderer & Kandel, 2000; Manahan-Vaughan et al., 2000; Redondo et
 18 al., 2010; Sajikumar et al., 2005). The major practical difference between the SBC
 19 model and the GB model pertains to the drift in the pre-depressive region of calcium.
 20 In the SBC model, weights drift towards a single baseline whereas in the GB model,
 21 synapses that are above a certain weight are stabilized toward an UP state while
 22 synapses that are below that weight are stabilized toward a DOWN state.

In this work, we will review and compare the SBC and GB models and make several modifications to both models to make them more analytically straightforward, thus allowing both experimentalists and theoreticians to use them with greater ease. Our modifications are based on the principle that a calcium-based plasticity model should allow for explicit and independent specification of critical parameters that capture the essential dynamics of the plasticity. Specifically, it should be possible to specify the fixed points of the asymptotic dynamics of the synaptic weights in each region of the $[Ca^{2+}]$ space (and in each region of the weight space in the GB rule), and it should also be possible to independently specify the rates (time constants) at which potentiation, depression, and drift take place without constraining the fixed points. We therefore call our model the fixed point – learning rate (FPLR) framework, as its dynamics can be fully characterized by specifying the fixed points and learning rates (time constants).

The FPLR framework is a phenomenological—as opposed to mechanistic—model of plasticity. The underlying assumption is that different levels of calcium specify different processes in the cell that have different teleological endpoints (i.e., the fixed points of the weights) and that occur at different rates. In other words, *the calcium concentration tells the synaptic weight where it is going and how fast it goes there*.

The dynamics of calcium-based plasticity can be captured by performing experiments where the synaptic $[Ca^{2+}]$ is held at a fixed level for some duration of time (on the order of minutes) until the synaptic weight reaches an asymptote, and then decreased to 0 for a longer duration of time (on the order of hours), which reveals both the fixed points and the rate at which they are reached (Fig. 1A).

The FPLR framework is more generic than the original SBC and GB rules, making it easy to specify arbitrary fixed points and learning rates based on experimental results. We show how the FPLR framework can be used to explore the dynamics of experimental results that are not captured by the original SBC and GB rules, such as incorporating two additional regions of $[Ca^{2+}]$ where neither potentiating nor depressive mechanisms are active, based on (Cho et al., 2001; J. E. Lisman, 2001; Tigaret et al., 2016), and modeling plasticity in Purkinje cells where the order of the potentiative region and depressive region are switched (i.e. $\theta_D > \theta_P$), based on (Coesmans et al., 2004; Piochon et al., 2016).

To synthesize the SBC and GB rules in the FPLR framework, we take a first step at expanding $[Ca^{2+}]$ -based plasticity models to encompass protein synthesis-dependent late-LTP (L-LTP) by proposing a framework where the $[Ca^{2+}]$ -dependent rules are conditional on the presence of synaptic proteins, and suggest that plasticity may behave like the SBC rule (drift to baseline) in the absence of protein synthesis, and more like the GB rule (stabilization) when protein synthesis has occurred.

An important aspect of the FPLR framework, as well as the original GB and SBC rules, is that because the dynamics of plastic changes are asymptotic, the magnitude (and sometimes

1 direction) of plastic change in response to a given plasticity protocol will depend on the initial
 2 synaptic weight. We illustrate this by reproducing simple frequency dependent plasticity and
 3 spike-timing-dependent-plasticity (STDP) protocols and demonstrating how different initial
 4 weights lead to different plastic consequences in the FPLR framework.

5 Finally, to demonstrate a novel application of the FPLR rules, we propose a calcium-
 6 based solution to the mechanism underlying plateau potential-induced place field formation in
 7 the hippocampus, also known as behavioral time scale plasticity (BTSP) (Bittner et al.,
 8 2015, 2017; Milstein et al., 2021). We show that the FPLR framework can explain how a
 9 plateau potential can either “overwrite” a previously created place field or create a new place
 10 field without overwriting the first place field, depending on the distance between the new and
 11 old place field.

12 Results

13 Basic SBC Rule

14 We first consider the calcium-based plasticity rules of Shouval et al. (Shouval et al., 2002),
 15 starting with the simplest formulation. In this rule, synaptic strength is modified in a
 16 straightforward manner according to the depression and potentiation thresholds: at any time
 17 step t , if the calcium concentration ($[Ca^{2+}]$) is in the pre-depressive range ($[Ca^{2+}] < \theta_D$) the
 18 synaptic weight w remains unchanged. If the calcium concentration is in the depressive range
 19 ($\theta_D \leq [Ca^{2+}] \leq \theta_P$) w is decreased, and if the $[Ca^{2+}]$ is in the potentiating range ($[Ca^{2+}] >$
 20 θ_P), w is increased. Formally, the change in the synaptic weight Δw ($\Delta w = w(t + 1) - w(t)$)
 21 at a given time as a function of the local $[Ca^{2+}]$ at that time (denoted as $Ca(t)$) is given by:
 22

$$23 \quad \Delta w = \eta \Omega(Ca(t)) \quad 1.1$$

25 Where η is the learning rate and Ω is the two-threshold calcium based plasticity function
 26 described above, which can be expressed most simply as a step function:
 27

$$28 \quad \Omega(Ca(t)) = \begin{cases} 0, & Ca(t) < \theta_D \\ k_D, & \theta_D \leq Ca(t) \leq \theta_P \\ k_P, & Ca(t) > \theta_P \end{cases} \quad 1.2$$

30 Where k_D and k_P are the signed rates of depression and potentiation, respectively ($k_D <$
 31 $0, k_P > 0$), and θ_D and θ_P represent the thresholds for depression and potentiation. If smooth
 32 transitions between regions are desired, this can also be expressed with a soft threshold
 33 using the sum of sigmoids (slightly modified from (Shouval et al., 2002)):

$$35 \quad \Omega(Ca(t)) = \frac{k_D}{1 + e^{-b_D(Ca(t) - \theta_D)}} + \frac{k_P - k_D}{1 + e^{-b_P(Ca(t) - \theta_P)}} \quad 1.3$$

37 where b_D and b_P control the sharpness of the transitions between regions in the Ω function.
 38

39 In the basic SBC rule, synaptic weights increase linearly at a rate of k_P in the
 40 potentiative region of $[Ca^{2+}]$, decrease linearly at rate of k_D in the depressive region of $[Ca^{2+}]$
 41 and remain stable in the pre-depressive region of $[Ca^{2+}]$ (Fig. 1B).

42 (We note that it may be more biologically plausible to implement the plasticity as a
 43 delayed function of the calcium signal, in which case one may substitute $Ca(t - D)$ in place of
 44 $Ca(t)$, where D indicates the duration of the temporal delay between the calcium signal and
 45 the plastic effect. For simplicity, however, we will assume that there is no such delay, i.e. $D =$
 46 0.)

47 SBC Rule with Weight Decay

48 To prevent synapses becoming arbitrarily large or small, Shouval et al.(Shouval et al.,
 49 2002) added a weight decay term to the original plasticity rule.
 50

$$51 \quad \Delta w = \eta(\Omega(Ca(t)) - \lambda w) \quad 1.4$$

52 Where λ is the rate of decay. Importantly, the change in the weight Δw now depends on both
 53

the $[Ca^{2+}]$ and the present value of the weight w . It can be instructive to visualize Δw as a function of both w and $[Ca^{2+}]$. We show (Fig. 1C) how the SBC rule with weight decay differs from the rule without it. In the pre-depressive range of $[Ca^{2+}]$, w decreases (or increases, if it is negative) asymptotically to the fixed point of 0 at a rate of $\eta\lambda$. In the depressive $[Ca^{2+}]$ range, w decreases asymptotically to the fixed point of $\frac{k_D}{\lambda}$ (or increases if the weight is below that point). In the potentiating range of $[Ca^{2+}]$, w increases asymptotically to the fixed point of $\frac{k_P}{\lambda}$ (or decreases if the weight is above the fixed point). Note that in this framework, there is no way to independently change the asymptotic behavior without changing either the rates of potentiation and depression (k_D and k_P) or the decay rate λ .

One way to visualize the dynamics of the SBC plasticity rule with and without weight decay is the simple experiment mentioned above where the synapse is exposed to a fixed amount of calcium in either the depressive or potentiating range over some time period, after which the calcium is decreased to 0 (Fig. 1A). In the original SBC rule without weight decay, the weight linearly increases (or decreases) for as long as the calcium pulse is present, then immediately stops changing (i.e., remains stable) when the calcium is turned off (Fig. 1B3). By contrast, when weight decay is used, the synapse potentiates or depresses asymptotically toward the fixed points when the calcium pulse is on, then drifts toward 0 when the calcium is turned off (Fig. 1C3).

SBC Rule with Weight Decay and Ca^{2+} -dependent learning rate

Weight decay that drifts quickly toward 0 in the absence of plasticity-inducing calcium may be an undesirable feature of a plasticity rule if we wish to model synapses that are potentiated or depressed for a long duration. Shouval et al. (Shouval et al., 2002) therefore introduced a sigmoidal $[Ca^{2+}]$ -dependent learning rate, $\Omega(Ca(t))$, to mitigate the effect of the weight decay in the absence of calcium. The SBC rule with the calcium-dependent learning rate is thus defined as:

$$\Delta w = \eta(Ca(t)) * (\Omega(Ca(t)) - \lambda w(t)) \quad 1.5$$

The basic idea is that instead of having a constant learning rate, the learning rate (including the rate of the weight decay) increases in a sigmoidal fashion with the amount of calcium (Fig. 1D). Thus, at pre-depressive levels of $[Ca^{2+}]$ the weight will decay slowly, allowing for greater stability over long time horizons. (Fig. 1E).

Fixed point – learning rate version of the SBC Rule

In the FPLR framework, we propose a modified version of the SBC rule which allows the modeler to specify the fixed points and learning rates explicitly in all three region of $[Ca^{2+}]$:

$$\Delta w = \eta(Ca(t)) * (F(Ca(t)) - w(t)) \quad 2.1$$

Here, instead of $\Omega(Ca(t))$, we use $F(Ca(t))$, a step function which describes the *fixed points* of the weights as a function of the $[Ca^{2+}]$. (We note that if λ is fixed to 1 in the original SBC rule [equation 1.5], $\Omega(Ca(t))$ also specifies the fixed points of the weights, however $F(Ca(t))$ is defined this way explicitly.) Moreover, $\eta(Ca(t))$ here is also a 3-valued step function (as opposed to a sigmoid in the original SBC rule) which determines the rate at which the weight asymptotically approaches the fixed point for each calcium level. (We require $\eta(Ca) \leq 1$ for all values of Ca to prevent oscillations). In this framework, the learning rate $\eta(Ca(t))$ defines the fraction of the distance between the current weight and the fixed point $F(Ca(t))$ which is traversed at each time step. (If $\eta(Ca(t)) = 1$) the synapse immediately jumps to the fixed point specified by $F(Ca(t))$, which can be useful for modeling discrete-state synapses.)

For example, we might have:

$$F(Ca(t)) = \begin{cases} 0.5, & Ca(t) < \theta_D \\ 0, & \theta_D \leq Ca(t) < \theta_P \\ 1, & Ca(t) \geq \theta_P \end{cases} \quad 2.2$$

And

$$\eta(Ca(t)) = \begin{cases} 0.015, & Ca(t) < \theta_D \\ 0.15, & \theta_D \leq Ca(t) < \theta_P \\ 0.25, & Ca(t) \geq \theta_P \end{cases} \quad 2.3$$

(Fig 1F) This means that synapses with pre-depressive calcium concentrations eventually drift toward a “neutral” state of 0.5 at a rate of 0.015, synapses with a depressive calcium concentration will depress towards 0 at a rate of 0.15, and synapses with a potentiative $[Ca^{2+}]$ will be potentiated towards 1 at a rate of 0.25 (Fig. 1G). (See (Enoki et al., 2009) for experimental evidence that synapses at baseline can be either potentiated or depressed). (Note that the fixed points and rates used here and in subsequent sections are specified in arbitrary units and meant to illustrate qualitative dynamics of the synapse only; see Figure 5 and **Methods** for biologically plausible parameters).

We can also turn off the drift in the pre-depressive region entirely by setting $\eta(Ca(t) < \theta_D) = 0$, thus allowing for synapses that are stable at every weight unless modified by depressive or potentiative calcium concentrations (Fig 1H,1I). This version of the rule can be useful for contexts in which the modeler is primarily interested in understanding the early phase of plasticity, or for theoretical work exploring the consequences of synaptic weights that potentiate and depress asymptotically.

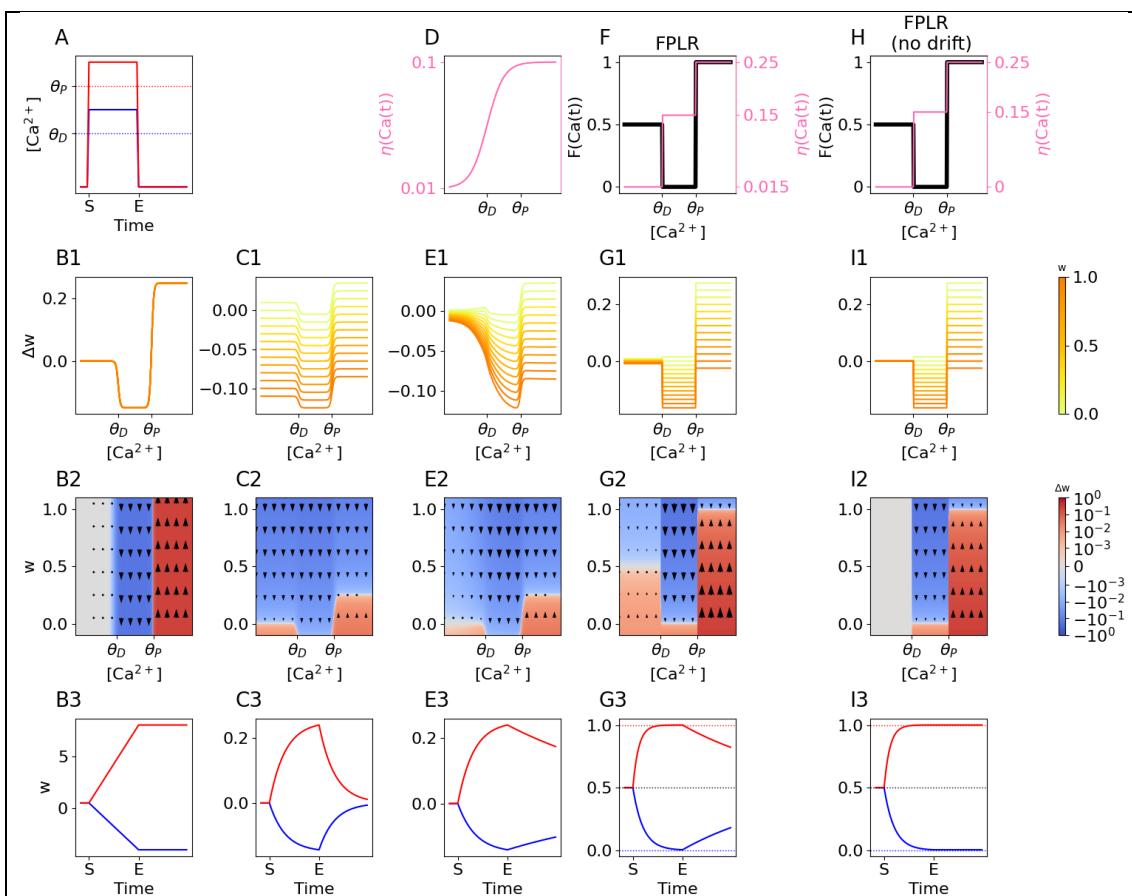


Figure 1: SBC and FPLR Rules for Calcium-based plasticity

(A) Basic stimulation to test the plastic effect of different levels of $[Ca^{2+}]$. $[Ca^{2+}]$ is either raised to a depressive level ($\theta_D \leq [Ca^{2+}] \leq \theta_P$, blue line) or to a potentiative level ($[Ca^{2+}] > \theta_P$, red line) for several minutes and is then reduced to and held at 0 for several hours to observe calcium-independent drift effects. (B1) Basic SBC two-threshold rule for $[Ca^{2+}]$ -dependent weight changes. (B2) Same as (B1) but with Δw presented as a heatmap of both the present value of w and the $[Ca^{2+}]$. In the basic SBC rule, there is no dependence on the present value of w and thus no variation along the vertical axis. Colors are log-normalized for visualization purposes. Red indicates potentiation, blue indicates depression, white indicates no change. (B3) Resultant weights over time when applying the basic SBC rule to the stimulation protocol from (A). Red and blue lines correspond to the red and blue lines (potentiative and depressive protocols, respectively) from (A). In the basic SBC rule, synapses potentiate or depress linearly in the presence of calcium and remain stable in the absence of calcium. (C1) SBC rule with weight decay. Here Δw depends on both the present value of w and the $[Ca^{2+}]$. Larger weights (darker lines) depress faster and potentiate more slowly than lower weights (lighter lines, some weight values are negative for illustrative purposes). (C2) Same

as (C1), presented as a heatmap. White line indicates fixed points. (C3) Weights over time in the SBC rule with weight decay. Solid lines and dashed lines indicate different weight initializations. Dotted horizontal lines indicate fixed points. (D) Calcium-dependent learning rate in the original SBC rule. The rate of weight change increases sigmoidally with the $[Ca^{2+}]$, allowing for a slower drift to baseline for pre-depressive levels of $[Ca^{2+}]$. (E1-E3) Dynamics of SBC rule with weight decay and $[Ca^{2+}]$ -dependent learning rate. (F) Fixed points ($F(Ca(t))$, black line, left y-axis) and learning rates ($\eta(Ca(t))$, pink line, right y-axis) for each region of $[Ca^{2+}]$ as step functions in the FPLR rule. (G1-G3) Dynamics of the FPLR rule. (H) Fixed points and learning rates for the modified SBC rule without drift. $\eta(Ca(t))$ is set to 0 in the pre-depressive region of $[Ca^{2+}]$. (I1-I3) Dynamics of the FPLR rule with no drift.

1

2 Versatility of the FPLR framework

3 Until now, all of the above rules have assumed that there are only three regions of
 4 $[Ca^{2+}]$ relevant for plasticity and that $\theta_D < \theta_P$. There are some experimental results that
 5 complicate this picture. For example, there is evidence that in Purkinje neurons, $\theta_D > \theta_P$, and
 6 that it is possible to shift the plasticity thresholds via inputs from other neurons (Coesmans
 7 et al., 2004). This can also easily be implemented in our modified plasticity rule by changing
 8 the fixed points in each $[Ca^{2+}]$ region. (Fig. 2A,B)

9 Moreover, even within hippocampal and cortical cells, there may be additional regions of
 10 $[Ca^{2+}]$ where the plasticity dynamics change. Cho et al. (Cho et al., 2001) found that
 11 within the depressive region of $[Ca^{2+}]$, the magnitude of depression exhibits a U-shaped
 12 relationship with the calcium concentration, such that a “No man’s zone” appears at the
 13 boundary between the depressive and potentiative region of $[Ca^{2+}]$, where the $[Ca^{2+}]$ is too
 14 large to induce depression but too small to induce potentiation (Cho et al., 2001; J. E.
 15 Lisman, 2001). It may make sense for a modeler to include this no-man’s zone more
 16 explicitly in the plasticity rule. There is also evidence from Tigaret et al. (Tigaret et al., 2016)
 17 that there is some maximum level of $[Ca^{2+}]$ beyond which potentiation mechanisms are
 18 inactivated. It would be worthwhile to be able to incorporate this “post-potentiative neutral
 19 zone” as well. These “neutral zones” can easily be incorporated into the modified SBC rule by
 20 adding additional thresholds into the step function from Eq. 2.1 and setting the learning rate to
 21 0 in those regions (the fixed points in the no-plasticity regions can be chosen arbitrarily, as
 22 they are irrelevant if the learning rate in those regions is 0).

23 If desired, we can also explicitly model the U-shaped dependency between $[Ca^{2+}]$ and
 24 synaptic depression. Although we have used a hard threshold step function to implement the
 25 previous examples of the modified SBC rule, in principle both $F(Ca(t))$ and $\eta(Ca(t))$ can be
 26 arbitrary functions as long as $\eta(Ca(t)) < 1$ for all values of $Ca(t)$. To implement a U-shaped
 27 region, we can extend the soft threshold step function equation from Eq. 1.3 into a general
 28 form to include arbitrary regions of $[Ca^{2+}]$:

$$30 \quad \eta(Ca(t)) = \sum_{i=1}^N \frac{\eta_i - \eta_{i-1}}{1 + e^{-b_i(Ca(t)-\theta_i)}} + \eta_0 \quad 2.4$$

31 Where $\theta_1 \dots \theta_i \dots \theta_N$ are the thresholds that differentiate between regions of $[Ca^{2+}]$ ordered
 32 such that $\theta_i < \theta_{i+1}$, $b_1 \dots b_i \dots b_N$ determine the steepness of transition between each $[Ca^{2+}]$
 33 region, $\eta_1 \dots \eta_i \dots \eta_N$ are the learning rates in each $[Ca^{2+}]$ region, and η_0 is the learning rate in
 34 the region between 0 and θ_1 . This representation approaches the equivalent step function as
 35 the values for b_i become sufficiently large. (A similar equation can be used for $F(Ca(t))$).
 36 Using Eq. 2.4, we can easily create the U-shaped relationship between $[Ca^{2+}]$ and synaptic
 37 depression (Fig. 2C,D).
 38

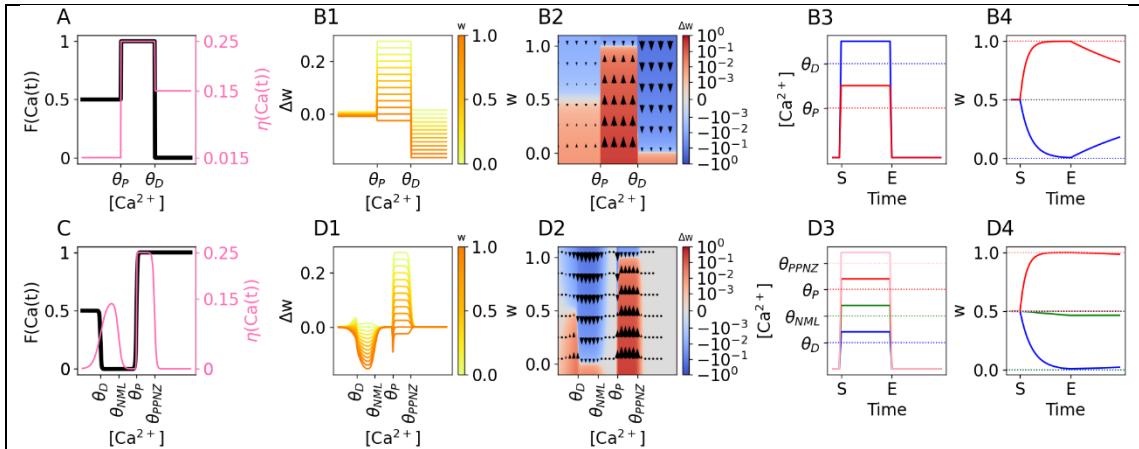


Figure 2: Incorporating Multiple Thresholds with the FPLR Rule

(A) Possible fixed points and learning rates for Purkinje cells, where the potentiation threshold is lower than the depression threshold ($\theta_P < \theta_D$). (B1) Δw in modified SBC rule for Purkinje cells using the fixed points and learning rates from (A). (B2) Same as (B1), presented as a heatmap. (B3) Stimulation protocol for Purkinje cells. Here, the $[Ca^{2+}]$ used for potentiation (red line) is less than the $[Ca^{2+}]$ used to induce depression. (B4) Weights over time in Purkinje cells for the stimulation protocols in (B3). Dotted lines represent fixed points. Solid and dashed lines represent different weight initializations. (C) Fixed points and learning rates, defined using a soft-threshold step function, with two additional regions of $[Ca^{2+}]$ where $\eta = 0$ and therefore no plasticity is induced. θ_{NML} : no man's land, θ_{PPNZ} : post-potentiative neutral zone. (D1) Δw in modified SBC rule using the fixed points and learning rates from (C). Note the U-shaped dependence of depression magnitude on the $[Ca^{2+}]$ resulting in a no mans zone between the depressive and potentiative regions of $[Ca^{2+}]$. Irregularities near the boundaries of $[Ca^{2+}]$ regions are due to the soft transitions in the fixed point and plasticity rate functions. (D2) Same as (D1), presented as a heatmap. (D3) Stimulation protocol for each region of $[Ca^{2+}]$ shown in (F). (D4) Weights over time for the stimulation protocols in (D3). Note that when the $[Ca^{2+}]$ is in the no man's zone (green line) depression occurs at a much slower rate, and in the post-potentiative neutral zone (black line) nearly no plasticity occurs.

1

Graupner and Brunel Model

One drawback of the SBC plasticity rules is that even in the final version with a calcium-dependent learning rate, synaptic weights eventually trend toward 0 in the presence of pre-depressive levels of calcium. There is some experimental evidence, however, that synapses are bistable, existing in a potentiated (UP) state with weight w_{UP} or a depressed (DOWN) state with weight w_{DOWN} , and that synaptic strengths slowly trend toward one of those two states depending on the early synaptic strength after inducing a plasticity protocol (Bagal et al., 2005; O'Connor et al., 2005a; Petersen et al., 1998). Graupner and Brunel (Graupner & Brunel, 2012) proposed a model that captured these dynamics. In their model, the synaptic efficacy, ρ (ρ is linearly mapped to w according to the equation $w = w_{DOWN} + \rho(w_{UP} - w_{DOWN})$) asymptotically decreases to the DOWN state (at $\rho = 0$) in the presence of depressive $[Ca^{2+}]$ or increases asymptotically to the UP state (near $\rho = 1$) in the presence of potentiating $[Ca^{2+}]$. When the $[Ca^{2+}]$ is pre-depressive, ρ either increases toward 1 or decreases toward 0 in a hyperbolic fashion on a very slow time scale depending only on the present value of ρ . Specifically, if ρ is larger than the value of an unstable fixed point ρ_* (set to 0.5), ρ trends toward the UP state whereas if ρ is smaller than this value, ρ trends toward the DOWN state (Fig. 3 - blue lines, Fig. 4A). Formally, we have:

19

$$\tau \frac{d\rho}{dt} = -\rho(1-\rho)(\rho_* - \rho) + \gamma_P(1-\rho)\Theta[Ca(t) - \theta_P] - \gamma_D\rho\Theta[Ca(t) - \theta_D] + Noise(t) \quad 3.1$$

20

Where τ is the overall time constant of synaptic change and γ_P and γ_D are parameters that denote the rate of potentiation and depression, respectively.

The first term in Equation 3.1 is always active and expresses the calcium-independent dynamics of the slow drift to the UP or DOWN state given the current value of ρ . The second term in this equation is only active while the $[Ca^{2+}]$ is above the potentiation threshold and describes the asymptotic potentiative dynamics, while the final term is active in both the depressive and potentiative regions of $[Ca^{2+}]$ and describes the asymptotic depressive dynamics.

21

22

23

24

25

26

27

28

29

1 Eq. 3.1 can also be expressed as a step function (ignoring the noise term):
 2

$$3 \tau \frac{d\rho}{dt} = \begin{cases} -\rho(1-\rho)(\rho_* - \rho), & Ca(t) < \theta_D \\ -\rho(1-\rho)(\rho_* - \rho) - \gamma_D \rho, & \theta_D \leq Ca(t) < \theta_P \\ -\rho(1-\rho)(\rho_* - \rho) - \gamma_D \rho + \gamma_P(1-\rho), & Ca(t) \geq \theta_P \end{cases} \quad 3.2$$

4
 5 Note that in the depressive region of $[Ca^{2+}]$ (second line), the calcium-independent term (first
 6 term) continues to affect the synaptic efficacy based on the current value of ρ . In the
 7 potentiating region of $[Ca^{2+}]$ (third line), both the calcium-independent term and the
 8 depressive term are active in addition to the potentiating term.
 9

10 Simplified Graupner and Brunel Model

11 While the GB rule can describe a variety of experimental results, its dynamics can be
 12 complicated by the fact that multiple processes are active simultaneously – that is, the slow
 13 calcium-independent drift of the first term is always active, and the depressive process is
 14 always active when the potentiation process is active. From a modeling standpoint, this
 15 aspect of the GB rule may not be desirable, as the dynamics in each region of $[Ca^{2+}]$ do not
 16 exhibit simple asymptotic behavior, the fixed point for potentiation is not trivial to specify (note
 17 that $\rho = 1$ is not a fixed point of ρ if $Ca(t) > \theta_P$, see Fig. 3C2), and specifying γ_P is
 18 insufficient to know the actual rate of potentiation because the depressive term γ_D also affects
 19 the potentiation rate.

20 From a biological standpoint, it is also questionable whether depressive and
 21 potentiating processes are active simultaneously. While some studies have shown that
 22 depressive and potentiative mechanisms are operative at the same time and compete with
 23 each other (Burrell & Li, 2008; O'Connor et al., 2005b), another study (Cho et al., 2001)
 24 argues that once the $[Ca^{2+}]$ reaches the potentiation threshold, the depressive mechanisms
 25 are turned off. The slow bistable drift mechanisms in the first term of Eq 3.2 may also not be
 26 perpetually active; the long-term stabilization mechanisms required for late LTP/LTD (L-LTP
 27 and L-LTD) have been shown to be protein synthesis dependent, and this protein synthesis
 28 may only occur after the induction of early-LTP/LTD (Barco et al., 2008; Frey & Morris,
 29 1997; Redondo et al., 2010).

30 We therefore propose a simplified version of the Graupner-Brunel rule which only has
 31 a single term active for any given concentration of calcium and synaptic efficacy value,
 32 resulting in a rule which achieves a qualitatively similar result but with more straightforward
 33 dynamics:

$$34 \frac{d\rho}{dt} = \begin{cases} -\gamma\rho, & Ca(t) < \theta_D \text{ and } \rho < \rho^* \\ \gamma(1-\rho), & Ca(t) < \theta_D \text{ and } \rho \geq \rho^* \\ -\gamma_D \rho, & \theta_D \leq Ca(t) < \theta_P \\ \gamma_P(1-\rho), & Ca(t) \geq \theta_P \end{cases} \quad 4.1$$

35
 36 In this rule, in the pre-depressive region of $[Ca^{2+}]$, ρ drifts toward 0 at a rate of γ if ρ is below
 37 or equal to ρ^* (first line, note that there is now no unstable fixed point at ρ^*), or asymptotically
 38 toward 1 at a rate of γ if ρ is above ρ^* (second line). In the depressive region of calcium (third
 39 line), ρ trends asymptotically toward the fixed point of 0 at a rate of γ_D , and in the potentiative
 40 region of calcium (fourth line), ρ trends asymptotically toward the fixed point of 1 at a rate of
 41 γ_P .

42 Unlike the original rule, the UP and DOWN states in the simplified rule occur precisely
 43 at $\rho = 1$ and $\rho = 0$ in all calcium regimes. Moreover, instead of a hyperbolic calcium-
 44 independent term, we use two linear terms to create a simple asymptotic increase or
 45 decrease toward the fixed points instead of a sigmoidal trend (Fig.3 - orange lines).
 46 Additionally, the rate of change within each region of the $[Ca^{2+}]$ is specified by its own
 47 learning rate ($\gamma, \gamma_D, \gamma_P$). (We note that if one wishes to simulate the dynamics of
 48 simultaneously active depressive and potentiative mechanisms, this is still possible within our
 49 framework by designing a soft-threshold learning rate that varies gradually with the $[Ca^{2+}]$ to
 50 mimic the observed dynamics of the weight changes, as in Figure 2C. This illustrates the
 51 difference between the phenomenological and the mechanistic perspective; the
 52 phenomenological approach is agnostic as to whether depressive and potentiative
 53 mechanisms are competing, all that matters is the observed endpoint and rate of the resultant
 54 plastic changes for a given $[Ca^{2+}]$.)

1 Astute readers may notice that the simplified GB rule is similar in structure to the
 2 modified SBC rule. In fact, the second two lines of Eq 4.1 are identical to the modified SBC
 3 rule with fixed points $\{F(\theta_D \leq Ca(t) \leq \theta_P) = 0, F(Ca(t) > \theta_P) = 1\}$ and learning rates
 4 $\{\eta(\theta_D \leq Ca(t) \leq \theta_P) = \gamma_D, \eta(Ca(t) > \theta_P) = \gamma_P\}$. The first two lines of Eq. 4.1, however, add
 5 a new feature, namely the dependence of the fixed points on the current weight ρ , in addition
 6 to the $[Ca^{2+}]$ (Fig. 4B).
 7

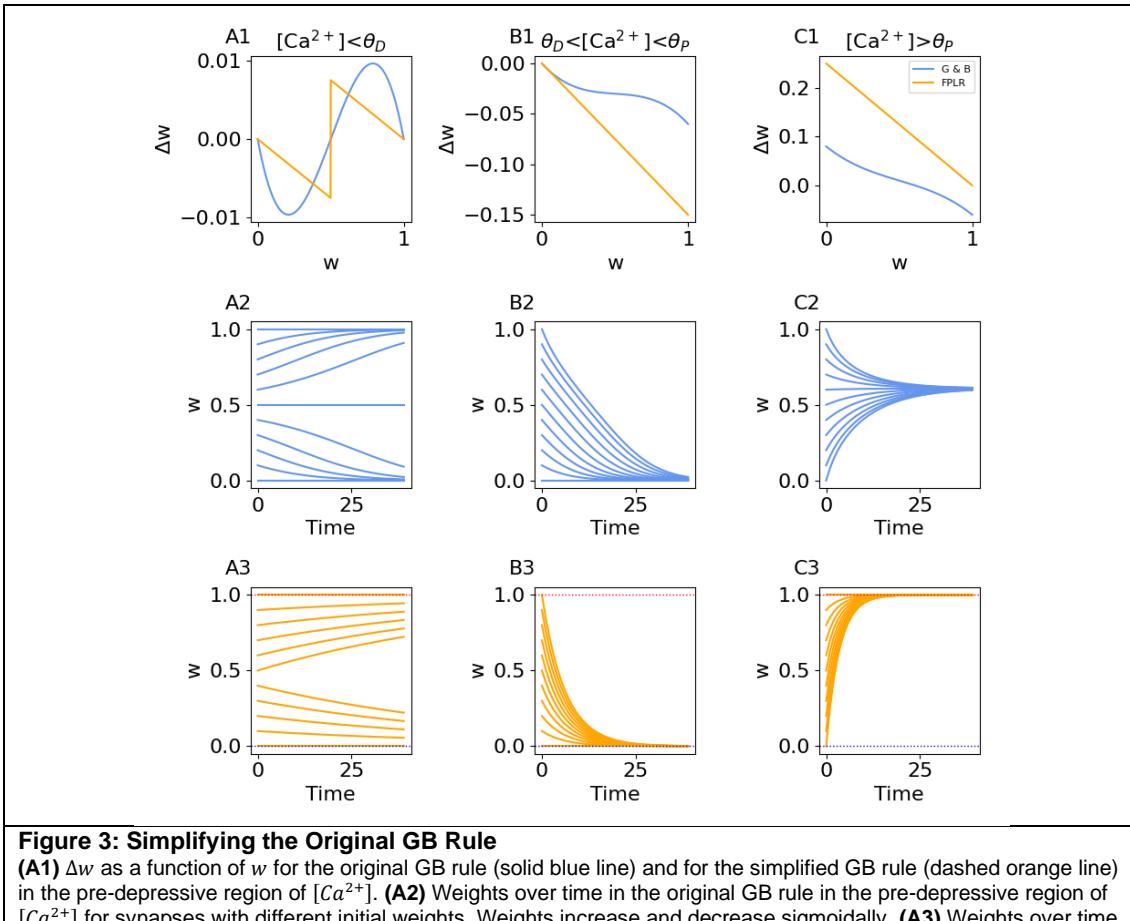


Figure 3: Simplifying the Original GB Rule

(A1) Δw as a function of w for the original GB rule (solid blue line) and for the simplified GB rule (dashed orange line) in the pre-depressive region of $[Ca^{2+}]$. (A2) Weights over time in the original GB rule in the pre-depressive region of $[Ca^{2+}]$ for synapses with different initial weights. Weights increase and decrease sigmoidally. (A3) Weights over time in the simplified GB rule, as in (A2). Weights increase and decrease asymptotically. (B1-B3) As in A for the depressive region of $[Ca^{2+}]$. (C1-C3) As in A for the potentiation region of $[Ca^{2+}]$.

FPLR version of the GB rule

It is possible to generalize the simplified GB rule into a fully generic two-dimensional FPLR plasticity rule that specifies the fixed points and learning rates as a function of both the synaptic $[Ca^{2+}]$ and the current weight w . Similar to the SBC rule, we have:

$$\Delta w = \eta(Ca(t), w(t)) * (F(Ca(t), w(t)) - w(t)) \quad 4.2$$

(We use w instead of ρ to specify the weight for consistency with the one-dimensional rule). Here, both the learning rates η and the fixed points F are two-dimensional step functions of both the $[Ca^{2+}]$ and the current weight w , as opposed to a one-dimensional step function of only the $[Ca^{2+}]$ in the SBC rule.

In addition to the bi-stable drift in the pre-depressive region of $[Ca^{2+}]$, Eq. 4.2 allows us to specify arbitrary numbers of weight-dependent fixed points in each region of $[Ca^{2+}]$. However, when specifying fixed points of the weights as a function of the present weights, care must be taken to avoid overlapping basins of attraction. For example, if, in some region of $[Ca^{2+}]$, the fixed point for a synapse with a weight of $w = 0.8$ is $w = 1$, the fixed point for weight $w = 0.9$ must also be $w = 1$, because $w = 0.8$ must pass $w = 0.9$ on its way to $w = 1$.

One way to enforce this constraint is to specify N fixed points and $N + 1$ boundaries of the basins of attraction within each region of $[Ca^{2+}]$ such that the fixed points are always

inside the closest basin boundaries on either side (Fig. 4C). For example, we can consider a rule that incorporates a tri-stable pre-depressive drift, where strong synapses drift to an UP state of $w = 0.9$, weak synapses drift to a DOWN state of $w = 0.2$, and synapses which aren't particularly strong or weak drift toward a MIDDLE state of $w = 0.5$ (instead of an unstable fixed point of $w = 0.5$). (We intentionally choose fixed points in the pre-depressive region of $[Ca^{2+}]$ that are different from those in the potentiative ($w = 1$) and depressive ($w = 0$) regions of $[Ca^{2+}]$ to reflect experimental results that synaptic weights during the early phase of LTP and LTD can overshoot/undershoot the eventual weights to which they are stabilized (Manahan-Vaughan et al., 2000; Redondo et al., 2010).) We can specify the fixed points for the pre-depressive calcium region $F(Ca(t) < \theta_D, w(t))$ by choosing the boundaries of the basins of attraction as $[0, 0.3, 0.7, \infty]$ and the fixed points as $[0.2, 0.5, 0.9]$. This translates into

$$F(Ca(t) < \theta_D, w(t)) = \begin{cases} 0.2, & w(t) \leq 0.3 \\ 0.5, & 0.3 < w(t) \leq 0.7 \\ 0.9, & w(t) > 0.7 \end{cases} \quad 2.5$$

(Fig. 4C, 4D1-D4).

We can illustrate the dynamics of the tri-stable FPLR rule by applying our canonical protocol of a step of potentiative or depressive calcium for either a short or long duration. A synapse exposed to the short-duration stimulus will briefly potentiate or depress, but insufficiently to escape the MIDDLE fixed point's basin of attraction, so it drifts back to the MIDDLE position. A synapse exposed long-duration stimulus, however, will escape the MIDDLE fixed point's basin of attraction and thus drift to either the UP or DOWN fixed point after the calcium step is turned off (Figure 4D5).

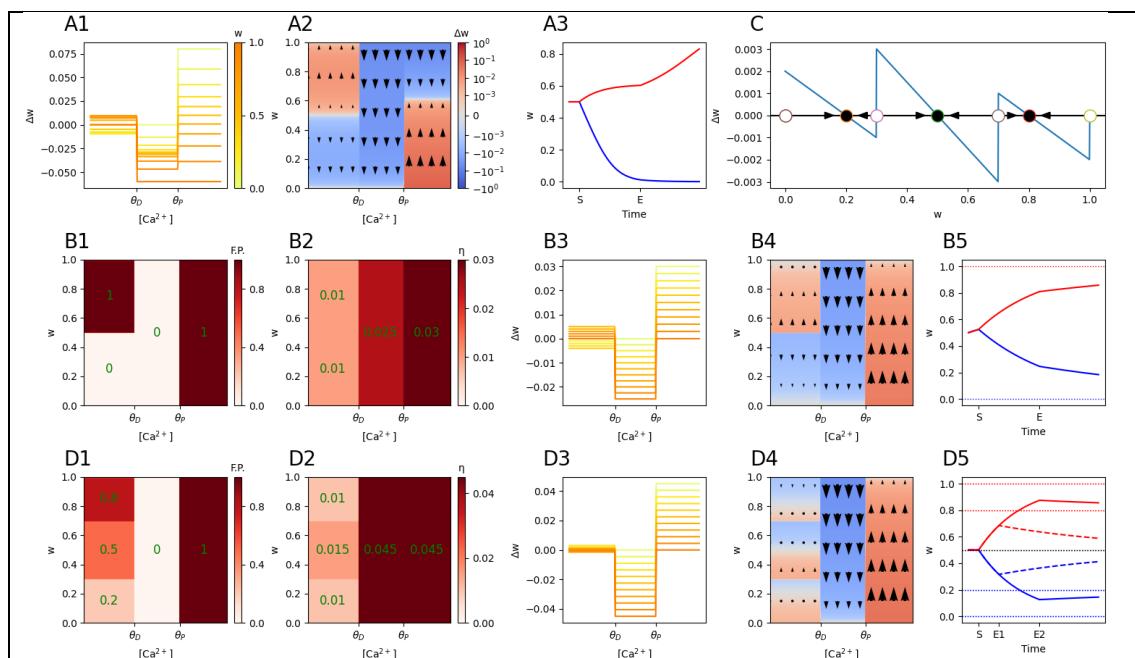


Figure 4: Comparing GB Rules

(A1) Δw in the original GB rule (darker lines indicate larger values for the current weight). Note that in the pre-depressive region of $[Ca^{2+}]$, the weight may increase or decrease (i.e. Δw can be greater than or less than 0) depending on the value of the initial weight. **(A2)** Same as (A1), presented as a heatmap. **(A3)** Weights over time in the original GB rule for the stimulation protocols from Fig. 1A. Weights drift towards either the UP or DOWN stable states depending on their value when the $[Ca^{2+}]$ is reduced to 0. **(B1)** Fixed points for the two-dimensional FPLR rule as a function of the current weight and $[Ca^{2+}]$. **(B2)** Learning rates for the two-dimensional FPLR rule **(B3-B4)** As in A1-A2 for the FPLR rule. **(B5)** Weights over time in the FPLR rule for the stimulation protocols from Fig. 1A. **(C)** Defining fixed points and basins of attraction in the generic GB rule. Within each region of $[Ca^{2+}]$, we can define an arbitrary number of stable fixed points (filled circles) as a function of the current weight w , as well as the boundaries of their respective non-overlapping basins of attraction (open circles). **(D1-D4)** As in (B) for an FPLR rule with a tristable synaptic weight in the pre-depressive drift region of $[Ca^{2+}]$, using fixed points from (C). **(D5)** A step of calcium of potentiative or depressive $[Ca^{2+}]$ is applied for either a short (2 seconds, dash-dot line, stimulation starts at S and ends at E1), or long (20 seconds, solid line, stimulation starts at S and ends at E2) duration. The long-duration stimulus escapes the middle fixed point's basin of attraction; the short duration stimulus does not. Note that

the long-duration stimulus overshoots/undershoots the late-phase fixed points.

Incorporating protein-dependent late LTP/LTD

One of the major differences between the original SBC rule and the GB rule pertains to what happens in the “late phase” of LTP/LTD, hours after an LTP/LTD stimulation protocol is completed and $[Ca^{2+}]$ levels have returned to baseline. In the SBC rule the weight eventually drifts to 0 (or some other baseline state in the modified SBC rule), while in the GB rule weights are slowly stabilized to one of two stable points, an UP state or a DOWN state, depending on the synaptic strength at the completion of the plasticity protocol.

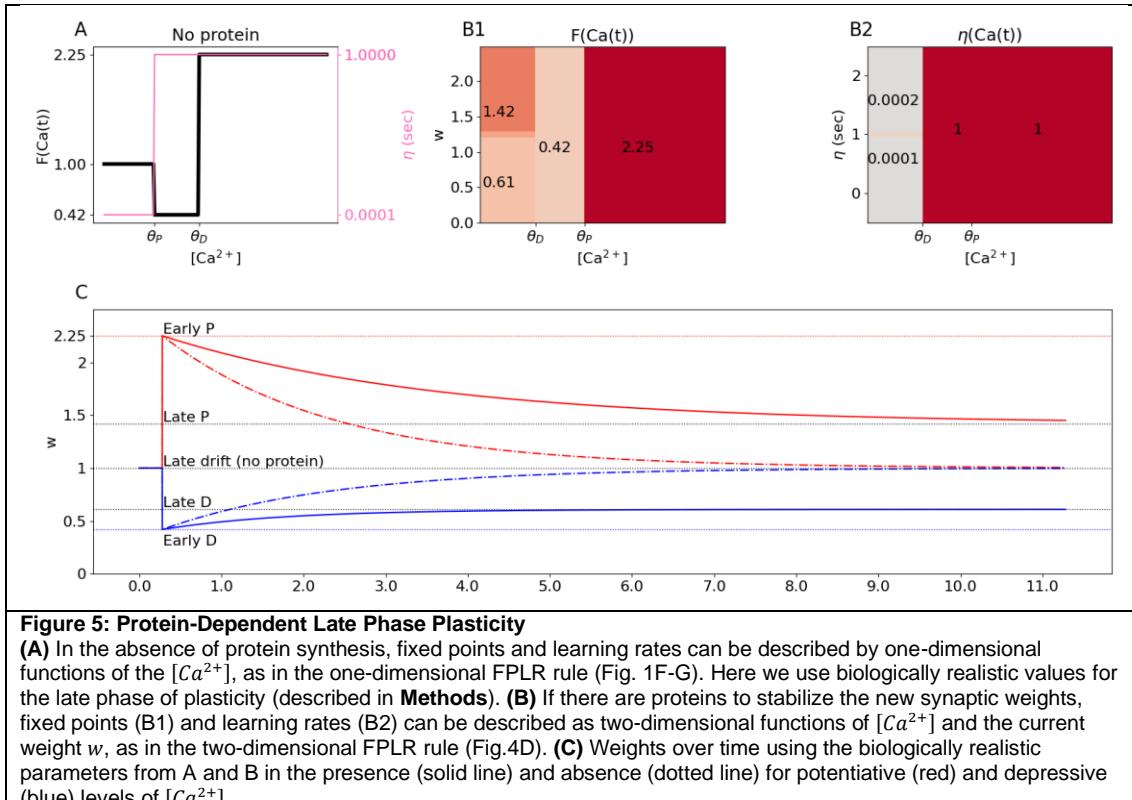
Biologically, the SBC rule is more representative of what has been observed in the absence of protein synthesis, and GB rule can be seen to correspond to the situations where proteins are synthesized to stabilize synaptic strengths for a longer period. In hippocampal cells, late-phase stabilization of potentiated and depressed synaptic states requires synthesis of proteins, without which synapses eventually drift back to their original strengths (Frey & Morris, 1997; Kauderer & Kandel, 2000; Redondo et al., 2010; Sajikumar et al., 2005). A similar phenomenon has been observed for LTD in Purkinje cells (Linden, 1996). This dependence on protein synthesis can be incorporated into the generic GB model by adding another dimension to the step functions for the weights and fixed points.

$$\Delta w = \eta(Ca(t), w(t), protein(t)) * (F(Ca(t), w(t), protein(t)) - w(t)) \quad 5.1$$

For illustration, if we make a simplifying assumption that there is a single stabilizing protein which can be either present (1) or absent (0), we might have the following rule:

$$\Delta w(Ca(t), w(t), protein(t)) = \begin{cases} \eta(Ca(t))(F(Ca(t)) - w), & protein(t) = 0 \\ \eta(Ca(t), w(t))(F(Ca(t), w(t)) - w), & protein(t) = 1 \end{cases} \quad 5.2$$

In other words, if there are no proteins to stabilize the new synaptic weights after a plasticity-inducing calcium stimulus, then synaptic weights simply drift to baseline, and the plasticity dynamics in each region of $[Ca^{2+}]$, including the pre-depressive region, can thus be described using a single weight-independent fixed point and learning rate in each region of $[Ca^{2+}]$, as in the generic SBC rule. However, if there are proteins to stabilize the new synaptic weights after a plasticity-inducing calcium stimulus, then fixed points and learning rates in the pre-depressive region of $[Ca^{2+}]$ are weight-dependent and the plasticity dynamics must be described with two-dimensional step functions, as in the generic GB rule. We illustrate these protein-dependent late phase dynamics using biologically realistic fixed points and time constants in Figure 5 (see **Methods**). Because the late phase dynamics operate on the order of hours and the early phase dynamics operate on the order of milliseconds, we can assume for simplicity that the early phase stimulation causes an instantaneous jump to the depressive or potentiative fixed point, i.e. we can set $\eta = 1 \text{ second}$ for the potentiative and depressive regions, while the late phase drift or stabilization occurs 4 orders of magnitude more slowly.



Modeling frequency and spike timing dependent plasticity

Although the FPLR framework makes the dynamics of calcium-based plasticity more straightforward, it is still quite similar to the original SBC and GB rules, and one would not expect to observe substantial discrepancies in the implementation of plasticity inducing protocols. Nevertheless, for the sake of completeness, we reproduce two classic protocols: frequency-dependent plasticity and spike timing dependent plasticity (STDP). Most of the literature about these protocols pertains to the early phase of plasticity, and we therefore use the version of the one-dimensional FPLR rule where the pre-depressive drift is turned off by setting $\eta(Ca < \theta_D) = 0$, as in Fig. 1I.

To model the calcium itself, we use the simplified formalism (Graupner & Brunel, 2012) where calcium concentration at spine k is modeled as a value which jumps by C_{pre} whenever there is a presynaptic spike at spine k , jumps by C_{post} whenever there is a postsynaptic spike, and decays toward 0 at the rate of τ_{Ca} (Graupner & Brunel, 2012). Formally, this can be expressed by the equation:

$$\Delta Ca^k(t) = Ca(t+1) - Ca(t) = -\frac{1}{\tau_{Ca}} Ca(t) + C_{pre} \sum_{\{t_i^k\}} \delta(t - t_i^k) + C_{post} \sum_{\{t_j\}} \delta(t - t_j) \quad 5.1$$

Where $\{t_i^k\}$ are the times of the presynaptic input spikes at spine k and $\{t_j\}$ are the times of the post synaptic spikes.

We first demonstrate how calcium-based plasticity results in synaptic changes that depend on the frequency of the presynaptic input. Experimentally, low-frequency stimulation (LFS) tends to produce depression, while high-frequency stimulation (HFS) tends to produce potentiation (O'Connor et al., 2005b). Intuitively, if C_{pre} is below θ_D , a single presynaptic input spike will not produce depression, but if several input spikes occur such that each spike rides on the tail of the previous spike, the $[Ca^{2+}]$ can build up such that it rises above θ_D , causing depression, and if the spikes occur with sufficiently high frequency, the $[Ca^{2+}]$ can rise above θ_P , resulting in potentiation (Figure 6A).

In order to determine the final synaptic weight after a plasticity protocol, it is not sufficient to simply compare the duration above θ_D and θ_P due to the difference in rates of the

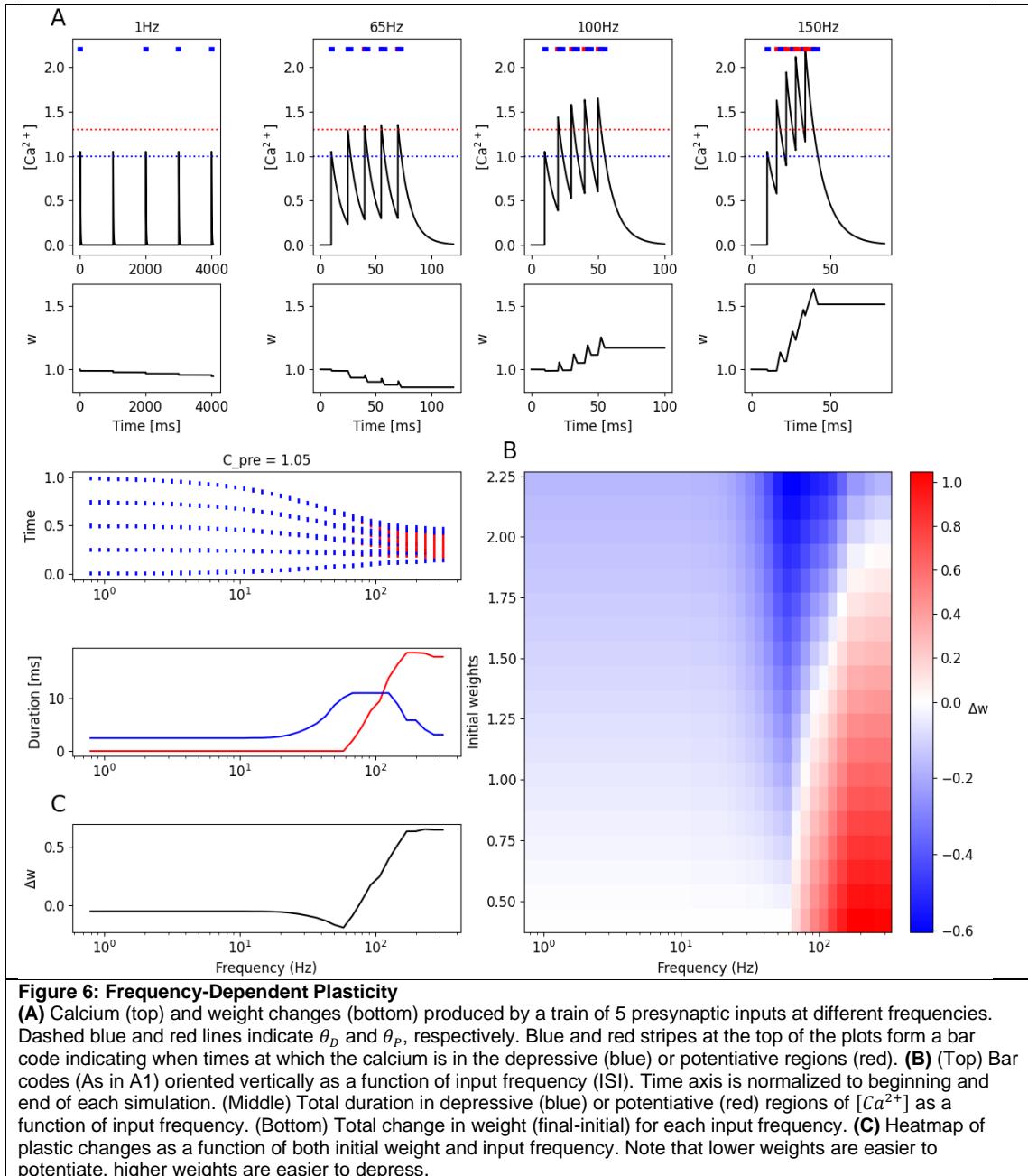
1 depressive and potentiative processes as well as the asymptotic dynamics of the plasticity,
 2 which make it easier to depress large weights and potentiate small weights. It can therefore
 3 be helpful to create a “bar code” for different stimulation protocols by indicating the times at
 4 which the $[Ca^{2+}]$ was above the θ_D and θ_P , which can help give a more intuitive feel for which
 5 protocols will result in synaptic potentiation or depression (Figure 6A-B).

6 Because the synaptic changes in the FPLR framework occur in an asymptotic
 7 manner, it is easier to potentiate weak synapses and easier to depress strong synapses. This
 8 means that the result of a frequency-dependent plasticity protocol will depend on the initial
 9 synaptic weight. If a synapse starts out closer to the depressive fixed point, it is easier to
 10 potentiate, while if it starts out closer to the potentiative fixed point, it is easier to depress. In
 11 fact, synapses which start out with large weights may even be depressed by a high-frequency
 12 protocol (Figure 6B).

13 The reason for high-frequency protocols depressing strong synapses in this model is
 14 subtle: whenever the $[Ca^{2+}]$ rises past θ_P , it will inevitably spend some time in the depressive
 15 region when the stimulation ends and the calcium decays to the baseline. Thus, for synaptic
 16 potentiation to be maintained, the magnitude of potentiation must be sufficiently large to not
 17 be completely erased by the subsequent depression. However, if the synapse starts out with
 18 a weight near the potentiation fixed point, virtually no potentiation can occur, so the
 19 depression during the decay phase of the calcium will be the only effect observed.

20 This “what goes up must come down” effect is an inevitable quirk of calcium threshold
 21 models which incorporate decaying calcium signals, and while this quirk can sometimes be
 22 helpful in explaining some experimental results, it creates complications for reproducing other
 23 experimental results, and it is also a bit counterintuitive. There is some experimental evidence
 24 that potentiation protocols will “lock in” potentiation to prevent subsequent depression
 25 (O’Connor et al., 2005b), which may help to alleviate this “what goes up must come down”
 26 problem. Our simulations here, however, do not include a lock-in feature, so potentiation
 27 protocols will always include a period of depression once the stimulation concludes and the
 28 $[Ca^{2+}]$ decays back to baseline. It is also possible to avoid this issue if the rate of depression
 29 is substantially slower than the rate of potentiation and the decay of the calcium is sufficiently
 30 fast such that the amount of depression that occurs while the calcium is decaying back to
 31 baseline after a potentiation protocol is negligible.

32



Spike Timing-Dependent Plasticity (STDP)

We can use our intuition from the above section about frequency-dependent plasticity to build an FPLR-based simulation of spike-timing dependent plasticity (STDP). We wish to replicate the “classic” STDP curve, where presynaptic input before postsynaptic stimulation causes potentiation at the activated presynaptic synapse, whereas postsynaptic spiking before presynaptic input causes depression, and both effects decrease with increased time intervals (Bi & Poo, 1998).

To reproduce this result, we simulated the calcium generated by single presynaptic spike (which creates a $[Ca^{2+}]$ pulse of height C_{pre}) at 100 ms of a 200 ms simulation. We then generate a postsynaptic spike (which creates a $[Ca^{2+}]$ pulse of height C_{post}) either before or after the presynaptic spike at varying timing intervals. By appropriately setting the values for C_{pre} and C_{post} , it is possible to replicate the classic STDP curve (different parameter values can result in different STDP curves, see (Graupner & Brunel, 2012)). As with the frequency-dependent protocol above, the magnitude of synaptic weight change in the STDP protocol will depend on the initial synaptic weight; large weights are faster to depress and

1 small weights are faster to potentiate.
 2

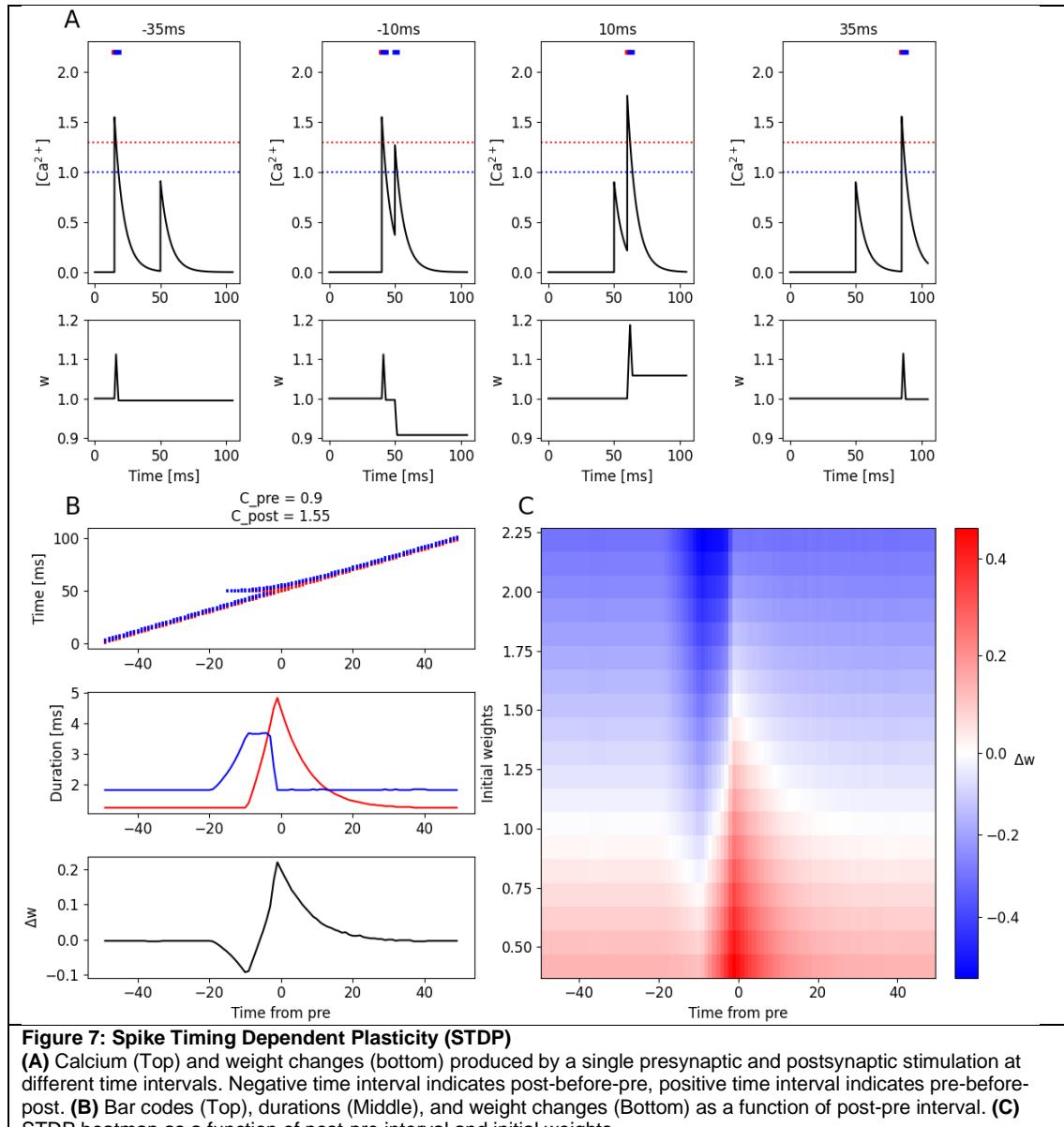


Figure 7: Spike Timing Dependent Plasticity (STDP)
(A) Calcium (Top) and weight changes (bottom) produced by a single presynaptic and postsynaptic stimulation at different time intervals. Negative time interval indicates post-before-pre, positive time interval indicates pre-before-post. **(B)** Bar codes (Top), durations (Middle), and weight changes (Bottom) as a function of post-pre interval. **(C)** STDP heatmap as a function of post-pre interval and initial weights.

Modeling Behavioral Time Scale Plasticity

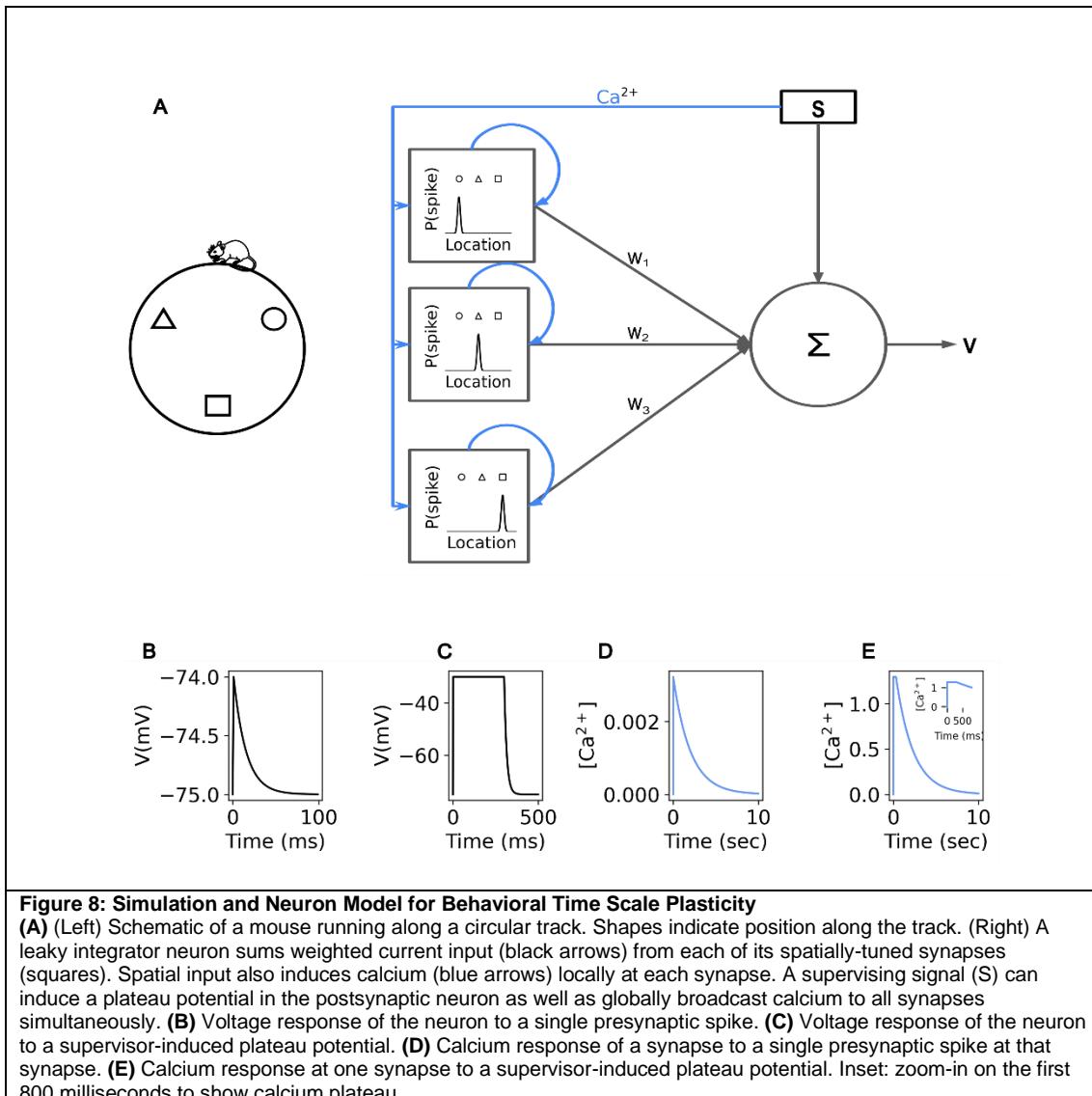
Recent experimental findings in the hippocampus have revealed a novel form of plasticity, known as behavioral time scale plasticity (BTSP) (Bittner et al., 2015, 2017; Milstein et al., 2021). A mouse running on a treadmill can spontaneously form hippocampal place fields when the soma is injected with a strong current, inducing a plateau potential. After a single induction, this plateau potential results in the neuron exhibiting a place field selective to the mouse's location few seconds before or after the time of the plateau potential. Moreover, this place field can be modified; if a second plateau is induced while the mouse is at a different location near the first place field, the place field will shift to the new location, thus "overwriting" the first place field. However, if the second location is sufficiently far away from the neuron's first place field, the neuron forms an additional place field at the second location without "overwriting" the first place field (Milstein et al., 2021).

Although BTSP has been previously modeled using an "eligibility trace" approach (Cone & Shouval, 2021; Gerstner et al., 2018; Milstein et al., 2021), we argue that the FPLR framework for calcium-based plasticity is sufficient to account for many of the results from the BTSP experiments. (See **Discussion** for further comments on the biological plausibility of each modeling approach.) The basic intuition for how calcium control can result

1 in BTSP is similar to Hebbian plasticity. Neither the presynaptic input nor the plateau potential
 2 themselves bring enough calcium into a spine for a long enough time to induce substantial
 3 changes to the synaptic weight. When the $[Ca^{2+}]$ traces from both the presynaptic sensory
 4 input and plateau potential coincide (or the $[Ca^{2+}]$ released from internal stores in the
 5 endoplasmic reticulum induced by these events, see **Discussion**), the combined $[Ca^{2+}]$ from
 6 both sources rises above the potentiation threshold for around a second, inducing potentiation
 7 at the synapses that were active at the time of plateau induction. Because the potentiated
 8 sensory inputs all correspond to the mouse's location at the time of plateau induction, these
 9 potentiated synapses comprise a place field at that location.

10 The reason why a second induction near to the location of the first induction will
 11 overwrite the first place field is because the partially decayed calcium trace from the plateau
 12 combines with the calcium from the presynaptic input corresponding to the first place field to
 13 surpass the depression threshold but not the potentiation threshold, depressing the synapses
 14 from the first place field. (This is if the second place field precedes the first place field on the
 15 track. If the first place field precedes the second place field on the track, the depression
 16 results from the decaying local calcium signal from the presynaptic input from the first place
 17 field combining with the $[Ca^{2+}]$ from the plateau potential.) If the two locations are far enough
 18 apart, however, the calcium from the plateau potential will have decayed sufficiently such that
 19 it will not surpass the depression threshold for a substantial amount of time when added to
 20 the local calcium from the presynaptic input (or vice versa). Finally, the reason why only
 21 synapses that were previously potentiated are depressed when "overwritten" in the BTSP
 22 protocol is because of the weight dependence of the FPLR rule. At baseline, all synapses that
 23 do not take part in a place field have weights near the depressive fixed point, so they cannot
 24 be further depressed.

25 To illustrate this, we simulated a mouse running at constant velocity on a circular
 26 track by sequentially presenting track locations as inputs to a leaky integrator model neuron
 27 with spatially-tuned presynaptic inputs. Calcium could enter a synapse as a consequence of
 28 its local input or due to the supervising signal, which induces a plateau potential in the neuron
 29 and globally broadcasts calcium to all synapses. The synaptic calcium decayed with a time
 30 constant of ~2 seconds (Fig. 8, see **Discussion** regarding the question of biologically
 31 plausible calcium decay time constants and potential implementation involving internal
 32 calcium stores). The synaptic weights of the model neuron implemented the FPLR calcium-
 33 based plasticity rule described in Equation 2.1 with the drift rate set to 0 as in the previous
 34 simulations.



1 The track was presented seven times to the neuron, simulating seven laps, with each
 2 lap taking ten seconds to run. At every location on the track, presynaptic inputs whose
 3 receptive fields overlapped with that location would contribute current to the neuron as well as
 4 induce a calcium signal with height C_{pre} at the associated “postsynaptic spine”. The
 5 presynaptic calcium signal decayed exponentially at a rate of τ_{pre} . On the first lap, no
 6 induction was performed, establishing a baseline of activity in the absence of a place field.
 7 During the second lap, a plateau potential was induced at 3.5 seconds into the lap
 8 depolarizing the neuron’s voltage and inducing a step of calcium for 300 milliseconds with
 9 height $C_{plateau}$ at all synapses. After the plateau induction, the calcium from the plateau
 10 decayed exponentially at a rate of τ_{post} . During the third lap, a voltage ramp was observed
 11 from ~ 1.5-5 seconds into the lap indicating that the plateau induction from the previous lap
 12 had produced a place field. In the fourth lap, a plateau potential was induced at 2 seconds
 13 into the lap. During the fifth lap, a voltage ramp was observed from ~ 0-3.5 seconds into the
 14 lap indicating that the previously induced place field had been “overwritten” by the plateau
 15 induction in the fourth lap. In the sixth lap, a plateau potential was induced at 7.5 seconds into
 16 the lap. During the seventh lap, two voltage ramps were observed, one ramp from ~ 0-3.5
 17 seconds and the other from ~ 5.5-9 seconds into the lap indicating that the plateau from the
 18 sixth lap induced a new place field but the place field observed at the fifth lap was not erased
 19 (Fig. 9).

20 To show how the FPLR calcium rule produced these results we visualized the $[Ca^{2+}]$,
 21 barcodes, and weights of each of the synapses over the entire course of the experiment (Fig.
 22 9C) focusing on the three synapses whose place fields were centered around the track
 23

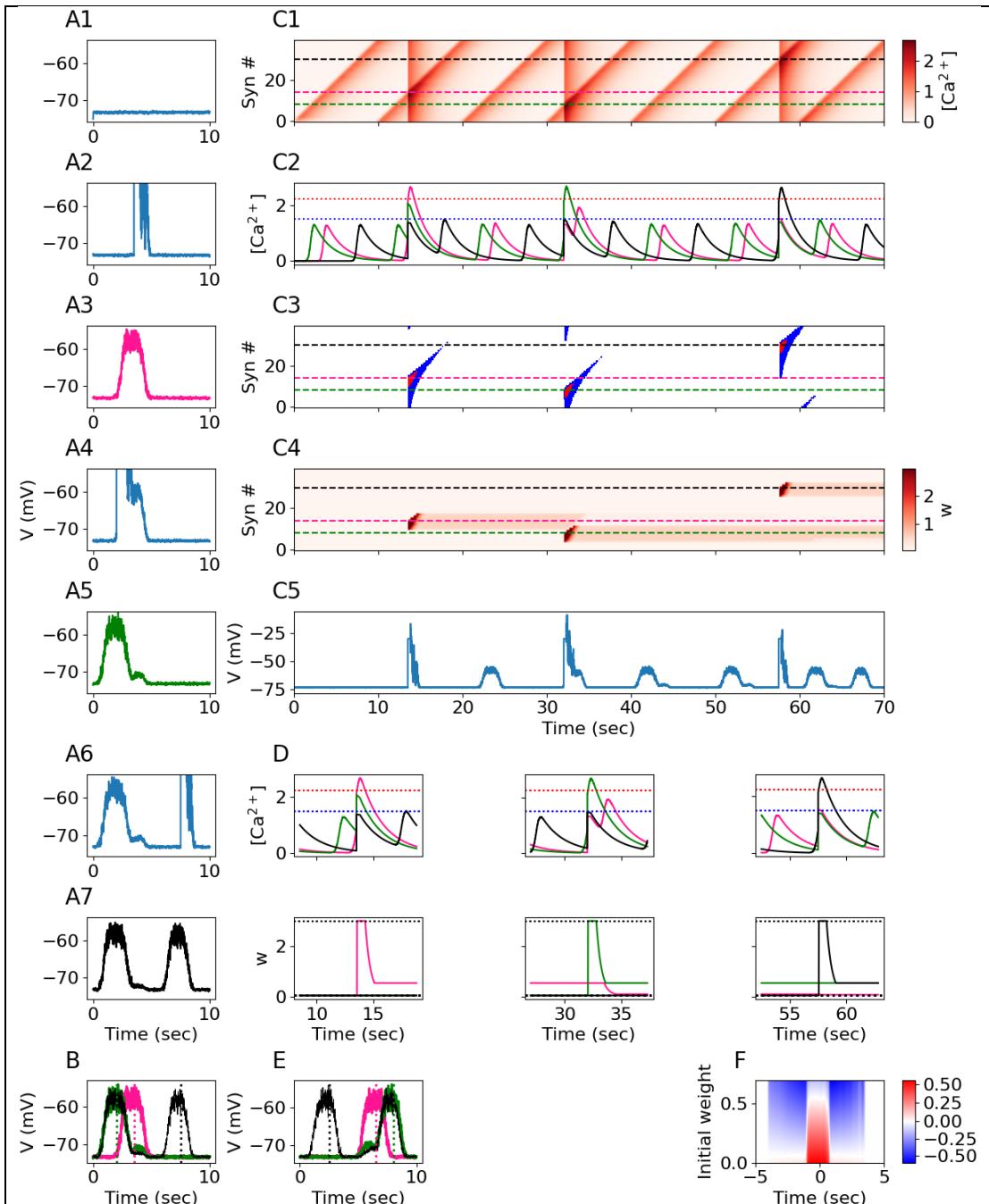
1 location of the plateau inductions of laps 2, 4 and 6 (Fig. 9B2, 8C). We call these synapses
 2 S3.5 (pink trace), S2.0 (green trace) and S7.5 (black trace), respectively, according to the
 3 track location (in units of seconds from beginning of track) at the center of their receptive
 4 fields. At the time of the first plateau induction, the $[Ca^{2+}]$ for S3.5 surpasses the potentiation
 5 threshold for ~ 1 second, potentiating it (and synapses with nearby receptive fields). Because
 6 of the “what goes up must come down” effect, S3.5 depresses substantially after it is
 7 potentiated due to the $[Ca^{2+}]$ spending some time in the depressive region while it decays.
 8 However, S3.5 still remains somewhat potentiated. The $[Ca^{2+}]$ for S2.0 also enters the
 9 depressive region, however because all the synapses were initialized at the depressive fixed
 10 point, S2.0 cannot be further depressed. This illustrates how the weight dependence of the
 11 FPLR rule has important functional consequences for place field formation. Without the
 12 weight dependence, there would always be a depressed region around the potentiated
 13 location, resulting in a Mexican hat-shaped place fields. The weight dependence ensures that
 14 only previously-potentiated synapses are ever depressed.

15 When the second plateau potential is induced in the fourth lap, the $[Ca^{2+}]$ for S2.0
 16 surpasses the potentiation threshold for ~ 1 second, potentiating it (with some subsequent
 17 depression) as before. The $[Ca^{2+}]$ of S3.5 enters the depressive region for ~ 1 second due to
 18 the combination of the local presynaptic $[Ca^{2+}]$ and the partial $[Ca^{2+}]$ from the decayed
 19 plateau potential. This results in the depression of S3.5 back to the depressive fixed point was
 20 previously. The weight S7.5 does not change, as before.

21 Finally, at the time of the third plateau induction, the $[Ca^{2+}]$ at S7.5 surpasses the
 22 potentiation threshold for ~ 1 second, potentiating it. Because the plateau induction at 7.5
 23 seconds into the track was sufficiently temporally distanced from the activation of the
 24 receptive fields of S2.0 and S3.5, the decayed local and supervisory calcium signals did not
 25 overlap to surpass the depression threshold for a significant amount of time, resulting in only
 26 a negligible magnitude of depression for S2.0 (Fig. 9C-D).

27 To ensure that all the above results hold when the locations of the inductions are
 28 reversed, we performed a “mirror image” experiment, where plateau inductions at laps 2, 4.
 29 and 6 occurred at 6.5 seconds, 8 seconds, and 2.5 seconds from the beginning of the track,
 30 respectively. The mirror image experiment yielded similar results (Fig. 9E, Fig S1).

31 To fully characterize the expected weight change induced by a plateau potential for
 32 synapses with different initial weights and receptive fields, we simulated a mouse running a
 33 single lap with a plateau potential induced in the middle of the lap (at 5 seconds). Using the
 34 $[Ca^{2+}]$ at each synapse we calculated the magnitude and direction of plasticity at each
 35 synapse for a range of initial weights. Synapses with receptive fields selective to locations
 36 near the center of the track potentiate if they started out with low weights but depress if
 37 initialized with large weights (due to the fact that synapses with large weights can't potentiate
 38 much, but do depress when the calcium signal decays because of the “what goes up must
 39 come down” effect). Synapses with receptive fields far from the center of the track will
 40 depress slightly if they are initialized with large weights but will not appreciably change if
 41 initialized with low weights (Fig. 9E). This profile is qualitatively consistent with the
 42 experimental results from (Milstein et al., 2021).
 43

**Figure 9: Behavioral Time Scale Plasticity**

(A1-A7) Voltage traces for 7 laps, each lap lasts for 10 seconds. A plateau potential is induced in the 2nd, 4th, and 6th laps. Green, pink, and black traces indicate place fields observed on the 3rd, 5th, and 7th laps, respectively. **(B)** Overlay of voltage traces for place fields from the 3rd (pink), 5th (green), and 7th (black) laps. Vertical dotted lines indicate plateau induction location during the preceding lap. **(C1)** Total $[Ca^{2+}]$ per synapse (rows, 40 synapses) over the course of all 7 laps. Dashed lines indicate three synapses whose receptive fields are centered at the location of the first (pink, S3.5, see text) second (green, S2.0) or third (black, S7.5) plateau induction. **(C2)** $[Ca^{2+}]$ traces for the three synapses indicated by the horizontal lines in (C1). Blue and red dotted lines indicate θ_D and θ_P . **(C3)** Plasticity bar codes for each synapse (rows) over all laps. **(C4)** Weights over time for each synapse over all laps. **(C5)** Voltage over time, as in A, for all 7 laps. **(D)** Zoom-in on the calcium traces (top) and weights (bottom) from the three synapses shown in B2 at the time of each plateau induction. In the second lap (left), the pink synapse's $[Ca^{2+}]$ rises beyond θ_P , inducing potentiation toward the maximum strength (dotted horizontal line at top), and the green synapse's $[Ca^{2+}]$ rises above θ_D , although the green synapse is already at the minimum strength (dotted horizontal line on bottom) so it can't depress any further. In the fourth lap (middle), the green synapse's $[Ca^{2+}]$ rises beyond θ_P , inducing potentiation, and the pink and black synapses' $[Ca^{2+}]$ rises above θ_D , but only the pink synapse can be depressed because it was previously potentiated. In the sixth lap (right), the black synapse's $[Ca^{2+}]$ rises beyond θ_P , inducing potentiation. **(E)** Result of a "mirror image" experiment, where the locations of plateau inductions are reversed (see supplementary figure S1). Green, pink, and black traces show laps 3, 5, and 7; dotted vertical lines indicate induction locations from laps, 2, 4, and 6, as in (B). **(F)** Change in weights as a function of initial weight and receptive field distance from plateau onset.

Discussion

In this work, we have developed a straightforward mathematical framework, the FPLR rule, to describe calcium-dependent long-term plasticity dynamics. The FPLR framework, based on the rules of (Graupner & Brunel, 2012; Shouval et al., 2002), enable modelers to describe plasticity dynamics in each region of $[Ca^{2+}]$ by specifying fixed points and learning rates as one-dimensional step functions of the $[Ca^{2+}]$ or as two-dimensional functions of both the $[Ca^{2+}]$ and the current value of the weight. This makes it simple to model novel experimental results such as the examples of Purkinje neurons and additional no-plasticity zones that we showed above. Additionally, we have shown the one-dimensional and two dimensional can be integrated into a single framework wherein SBC-like decay dynamics occur in the late phase of plasticity in the absence of protein synthesis, whereas GB-like stabilization dynamics occur in the late phase when proteins are available to stabilize weight changes made in the early phase of plasticity. The FPLR framework, which allows for an arbitrary number of fixed points, also fits well with the theoretical and experimental literature suggesting that synaptic weights have on the order of 10 discrete states comprised of nanoclusters of AMPA receptors (Bartol et al., 2015; Liu et al., 2017).

To demonstrate the ability of the FPLR framework to reproduce classic plasticity protocols, we used the FPLR approach to implement frequency-dependent and spike timing dependent plasticity. We introduced the technique of plasticity “bar codes” as a simple way of keeping track of the times when the $[Ca^{2+}]$ was in the potentiative or depressive region. We showed that due to the saturating nature of plasticity, the outcome of frequency-dependent plasticity and STDP (indeed, any plasticity protocol) depends on the initial synaptic weights—it is easier to depress strong weights and potentiate weak weights. Finally, we showed that this weight-dependence of the FPLR framework enables us to explain a novel experimental result – behavioral time scale plasticity (BTSP).

Alternative Calcium-Based Plasticity Models

We note that both the SBC and GB rules as well as the FPLR framework assume that synaptic weight changes depend on the magnitude of $[Ca^{2+}]$. However, there are other theories as to how plasticity might depend on $[Ca^{2+}]$. For example, changes in synaptic weights may depend both on the duration and magnitude of the calcium signal in a way that is not captured by the SBC and GB rules. Alternatively, the source of calcium – i.e. NMDA receptors, voltage-gated calcium channels or internal calcium stores – may affect long term plasticity due to different second messengers being localized to specific “nanodomains” near the different calcium sources (For a review of alternative models, see (Evans & Blackwell, 2015)).

More detailed models have also been developed that explicitly incorporate calcineurin and CamKII activity (Li et al., 2023; Rodrigues et al., 2021) as well as other presynaptic and postsynaptic mechanisms (Ebner et al., 2019). It is also possible to explicitly model protein dynamics for plasticity stabilization, for example via synaptic tag-and-capture model (Clopath et al., 2008). While more detailed plasticity rules may enable more specific experimental predictions, over 100 molecules have been implicated in long-term plasticity (Sanes & Lichtman, 1999), making it effectively impossible to have truly comprehensive mechanistic model. Our phenomenological framework can thus be useful for creating models that have relatively few parameters while still capturing essential aspects of calcium-based plasticity. Our fixed point–learning rate approach can also be extended to incorporate other molecular mechanisms, as we suggested with protein synthesis-dependent late-phase plasticity, although some molecular processes may not be well-described in this framework. Additional model complexity may also be needed to model plasticity in physiological concentrations of extracellular calcium (Inglebert et al., 2020; Inglebert & Debanne, 2021).

Modeling Calcium

Throughout this work, we have used simple exponentially decaying calcium stimuli to demonstrate the dynamics of our plasticity rules. When modeling neurons in a detailed fashion, it will usually be necessary to explicitly model the calcium influx from various sources, such as voltage-gated calcium channels and NMDA receptors, in order to explore the plastic consequences of presynaptic and postsynaptic activity. While detailed modeling of calcium dynamics is beyond the scope of this paper, work in this direction can be found in the original

1 SBC and GB papers (Graupner & Brunel, 2012; Shouval et al., 2002) as well as
 2 elsewhere in the modeling literature (Chindemi et al., 2020).

4 **Modeling Behavioral Time Scale Plasticity**

5 As we noted in **Results**, previous attempts to model BTSP suggested an “eligibility
 6 trace” approach, where separate presynaptic and postsynaptic signals must interact with each
 7 other to produce plasticity (Cone & Shouval, 2021; Gerstner et al., 2018; Milstein et al.,
 8 2021). However, given the extensive evidence for the calcium basis of plasticity in other
 9 contexts, and the fact that NMDA and VGCC channel blockers disrupt BTSP, (Bittner et al.,
 10 2017) we argue that it is worthwhile to give serious consideration to the possibility that
 11 calcium is the mechanism underlying BTSP.

12 One reason why eligibility traces were used in previous work (Cone & Shouval,
 13 2021) is that classical calcium-based protocols like STDP occur at timescales that are faster
 14 than would be necessary to model BTSP. Indeed, in our own replications of STDP and
 15 frequency-dependent plasticity, we used time constants for the decay of calcium on the order
 16 of 10 milliseconds, while modeling BTSP required calcium decay time constants on the order
 17 of seconds. Calcium imaging data often shows calcium decay constants on the order of
 18 hundreds of milliseconds and that decay constants can vary widely due to various active and
 19 passive mechanisms (Majewska et al., 2000). However, time constants extracted from
 20 calcium imaging may not necessarily be reliable for modeling calcium-based plasticity, as the
 21 calcium indicator itself may interfere with the dynamics of the calcium. Indeed, more recent
 22 work has shown that calcium indicators with lower binding affinity show smaller decay
 23 constants, on the order of tens of milliseconds (Miyazaki & Ross, 2022).

24 We suggest that various molecular mechanisms inside the cell, such as the
 25 manipulation of calcium pumps, or intracellular calcium induced calcium release (CICR) from
 26 internal calcium stores (Rose & Konnerth, 2001), can titrate the speed at which calcium
 27 decays, enabling calcium-based plasticity to operate on a spectrum of timescales. Recent
 28 experimental and modeling studies have also shown the importance of intracellular calcium
 29 release for BTSP (Caya-Bissonnette et al., 2023; O’Hare et al., 2022). Although the free
 30 calcium observed via calcium imaging decays at fast time scales, presynaptic or postsynaptic
 31 events may trigger influx of calcium into intracellular calcium stores in the endoplasmic
 32 reticulum, which is later released. As such, the calcium traces in our model can represent a
 33 simplified description of the net $[Ca^{2+}]$ contained both in the cytosol and the intracellular
 34 calcium stores, the calcium from the latter being released to the cytosol when presynaptic
 35 stimulation or a supervising signal occurs. Such a process could be more precisely described
 36 by explicitly modeling CICR dynamics (Caya-Bissonnette et al., 2023). The dynamics of
 37 second messengers, such as CaMKII and phosphatases, may also underlie the longer
 38 timescales of BTSP (Jain et al., 2023; Li et al., 2023).

39 **Future Directions**

40 The FPLR framework suggests a standardized experimental paradigm to characterize
 41 plasticity dynamics. Namely, $[Ca^{2+}]$ should be fixed at the potentiative or depressive value for
 42 a sufficient duration such that the synaptic weight no longer changes (or by performing an
 43 analogous LTP or LTD protocol such that weight saturation is observed). Then the synaptic
 44 strength should be observed over a period of hours or longer to observe the eventual late-
 45 phase drift fixed points. From these observations it is straightforward to characterize the
 46 FPLR fixed points (i.e. the weight saturation and drift values) as well as the learning rates by
 47 fitting a simple exponential to the experimentally observed dynamics.

48 Without performing these experiments, it is difficult to disambiguate the fixed points
 49 and the plasticity rates. A synapse which is potentiated via an HFS protocol can be
 50 potentiated to an even higher strength by performing another HFS protocol, thus merely
 51 looking at the peak EPSP value after a standard LTP/LTD protocol is insufficient to determine
 52 the potentiative or depressive fixed point (Enoki et al., 2009). Performing the necessary
 53 experiments to find fixed points and learning rates among different cell types, species, brain
 54 regions and even at different dendritic locations within the same neuron can lead to a deeper
 55 understanding of long-term plasticity.

56 In addition to enabling experimentalists to model their results, the FPLR framework is
 57 sufficiently simple and flexible that theoreticians can incorporate them into neuron models of

1 varying levels of complexity, ranging from simple point neuron models to detailed biophysical
 2 models. This can open avenues toward understanding how calcium-based plasticity can lead
 3 to learning at the single neuron or even network-level resolution. The flexibility of our
 4 framework to specify arbitrary fixed points and plasticity rates can also enable the exploration
 5 of activity-dependent changes in plasticity rules, known as metaplasticity (Abraham, 2008).

6 Methods

7 All figures were created in python using the Numpy (Harris et al., 2020) and Matplotlib
 8 (Hunter, 2007) packages.
 9

10 Plasticity parameter estimates

11 Everywhere in the paper except for Figure 5, learning rates and fixed points are
 12 meant to convey qualitative understanding of the dynamics of plasticity, not biologically
 13 realistic parameters.

14 For Figure 5, we used the following experimental results to approximate biologically
 15 realistic time constants: in rat hippocampus, for LTD (*in vivo*), in the early phase of
 16 depression, synapses depress to 42% of their initial strength after initializing a low-frequency
 17 stimulation (LFS) (900 pulses at 1 Hz) and either drift back to ~92% of their baseline in the
 18 presence of anisomycin after 3-4.5 hours (which blocks protein synthesis) or stabilize at
 19 ~61% of their baseline in the absence of anisomycin after 3-4.5 hours (Manahan-Vaughan
 20 et al., 2000). For LTP (*in vitro*), in the early phase of potentiation, synapses potentiate to
 21 ~225% of their initial strength minutes after initializing a high-frequency stimulation protocol
 22 (100 pulses at 100 Hz, performed 3 times) high-frequency stimulation (HFS) and either drift
 23 back to baseline in the presence of anisomycin after 10 hours or stabilize at ~142% of their
 24 baseline in the absence of anisomycin after 10 hours (Redondo et al., 2010).

25 To calculate the time constants, we assumed that the calcium signal from each pulse
 26 lasted ~10 ms. As such, the calcium signal lasted ~3 seconds and the LTD calcium signal
 27 lasted ~9 seconds.

28 Because different time constants were found for the drift back to baseline for LTP and
 29 LTD in the presence of anisomycin, we used the mean of the learning rates that were
 30 calculated (0.0046) from both experiments, although in principle it is possible to use a 2-
 31 dimensional rule to differentiate the drifts down from the potentiated state or up from the
 32 depressed state even in the absence of protein synthesis.

33 The fixed points for the weights in Figure 5 were not precisely calibrated to
 34 experimental data due to substantially different baselines in peak EPSP measured in various
 35 plasticity experiments (Enoki et al., 2009; Manahan-Vaughan et al., 2000; Redondo et
 36 al., 2010). The values used for fixed points in Figure 5 can be thought of as “change relative
 37 to baseline” where value of 1 indicates no change. When modeling experimental data,
 38 depending on the context, “synaptic weights” may refer to AMPA conductance, number of
 39 AMPA receptors/nanoclusters, integral or peak EPSP or EPSC measured at the dendrite or
 40 the soma, spine head volume or area, number of docked vesicles, release probability, or
 41 other parameters; fixed point values can be expressed directly in units of the relevant
 42 parameters.

43 Behavioral Time Scale Plasticity Simulation

44 To simulate behavioral time-scale plasticity (BTSP), we used a leaky integrator
 45 neuron where the voltage V is determined by the differential equation:

$$46 C_m \frac{dV}{dt} = I_{leak}(t) + I_{syn}(t) + I_{plateau}(t) \quad 6.1$$

47 Where $C_m = 1 \text{ nF}$, I_{leak} is the leak current, I_{syn} is the total contribution to the postsynaptic
 48 neuron from all presynaptic inputs, and $I_{plateau}$ is the current contribution from a plateau
 49 potential induction. (As we are mainly interested in the subthreshold ramp activity rather than
 50 postsynaptic spikes, this model neglects postsynaptic spiking activity). The leak current is
 51 defined as:

$$52 I_{leak}(t) = -g_L(V(t) - V_{rest}) \quad 6.2$$

53 Where g_L is the leak conductance. V_{rest} is the resting potential of the membrane. It is helpful
 54 to define the voltage leak time constant $\tau_V = \frac{C_m}{g_L}$. In the absence of input, the voltage thus

1 decays according to the equation $\frac{dV}{dt} = -\frac{1}{\tau_V}(V(t) - V_{rest})$

2

3 The presynaptic current is defined as:

4

$$I_{syn}(t) = \sum_{i=1}^N w_i * s_i(t) \quad 6.3$$

5 where w_i is the weight of synapse i and $s_i(t)$ is a binary variable indicating whether
6 synapse i produced a spike at time t . The probability of a presynaptic spike at synapse i was
7 defined as:

8 $p(s_i = 1) \sim Bernoulli(f_i(t)) \quad 6.4$

9 Where $f_i(t)$ is a bell-shaped receptive field centered around the neuron's preferred track
10 location l_i . (We can define locations in terms of milliseconds of time from the start of the lap
11 because we assume the mouse runs at constant velocity.) If the mouse's location is given as
12 $l(t) = t \bmod T$, where T is the track length in units of milliseconds of running time, we have:

13 $f_i(t) = P_{max} * e^{-\left(\frac{(l_i - l)^2}{r^2}\right)} \quad 6.5$

14 where P_{max} is the probability of the neuron firing a spike when the mouse is at the center of
15 the receptive field of that neuron, and r determines receptive field width. The presynaptic
16 receptive fields thus tile the track length with one receptive field center every $\frac{T}{N}$ milliseconds of
17 running distance.

18 To ensure that the plateau potential took the form of a rectangular voltage clamp
19 step, we defined the plateau current as:

20 $I_{plateau}(t) = (-I_{leak}(t) - I_{syn}(t) - V(t) + V_{plateau}) * \sum_{t_i \in \{t_i\}} \delta(t - t_i) \quad 6.6$

21 Where $V_{plateau}$ is the target steady state voltage during the plateau induction and $\{t_i\}$ is the
22 set of all times at which the plateau induction is active.

23 The $[Ca^{2+}]$ at each synapse i , Ca^i , is the sum of the local, presynaptically-induced
24 calcium, Ca_{pre}^i , and the global, plateau-induced calcium, $Ca_{plateau}$.

25 $Ca^i = Ca_{pre}^i(t) + Ca_{plateau}(t) \quad 6.7$

26 The presynaptic calcium is modeled as a pulse of calcium with an initial concentration of
27 pre_height , which exponentially decays with a time constant of τ_{ca} :

28 $\frac{dCa_{pre}}{dt} = -\frac{Ca(t)}{\tau_{ca}} + pre_height * s_i(t) \quad 6.8$

29 The postsynaptic calcium is modeled as a rectangular step of calcium of height
30 $plateau_height$ which also decays to baseline at a rate of τ_{ca} once the induction ends:

31 $\frac{dCa_{plateau}}{dt} = -\frac{Ca_{plateau}(t)}{\tau_{ca}} + \left(plateau_height + \frac{Ca_{plateau}(t)}{\tau_{ca}}\right) * \sum_i \delta(t - t_i) \quad 6.9$

32

33 The parameters for BTSP were fitted using the differential evolution algorithm from SciPy
34 (Virtanen et al., 2020) with a cost function designed to qualitatively reproduce the basic
35 experimental results from (Milstein et al., 2021), (i.e. to qualitatively obtain the results
36 shown in figure 9E-D).

37 Simulation Parameters

40 Frequency-dependent plasticity:

Parameter	Value
Ca_{pre}	1.05
w_0	1
τ_{ca}^*	10 ms*
θ_D	1
θ_P	1.3
η_D	0.04
η_P	0.055
F_D	0.42 (Manahan-Vaughan et al., 2000)
F_P	2.25 (Redondo et al., 2010)

dt (step size for numerical integration)	0.01 ms
1	

1 **STDP**

Parameter	Value
Ca_{pre}	0.9
Ca_{post}	1.55
w_0	1
τ_{ca}	7 ms*
θ_D	1
θ_P	1.3
η_D	0.1
η_P	0.075
F_D	0.42
F_P	2.25
dt	0.01 ms

2
3**BTSP:**

Parameters	
N	40
r	329.23
V_{rest}	-75 mv
$V_{plateau}$	-30 mv
pre_height	0.003
$plateau_height$	1.295
$plateau\ duration$	300 ms
τ_{ca}	2023.327 ms
τ_V	15 ms
P_{max}	0.9
T	10,000 ms
θ_D	1.5
θ_P	2.247
η_D	0.723
η_P	0.002182
F_D	0.06
F_P	2.98

4

5

Code Availability

6

7

Code for the simulations in this paper can be found at <https://github.com/tmoldwin/FPLR>.

8

9

Acknowledgments

10

The authors would like to thank Aaron Milstein, Christine Grienberger, and Léa Caya-Bissonnette for their helpful input with respect to the BTSP portion of this paper. This work was supported by a grant from the Office of Naval Research, No. 138151-5125597, the Drahi Family Foundation, and the ETH domain for the Blue Brain Project.

11

12

13

1

2 **References**

- 3 Abraham, W. C. (2008). Metaplasticity: Tuning synapses and networks for plasticity.
 4 In *Nature Reviews Neuroscience* (Vol. 9, Issue 5, pp. 387–399).
 5 <https://doi.org/10.1038/nrn2356>
- 6 Bagal, A. A., Kao, J. P. Y., Tang, C. M., & Thompson, S. M. (2005). Long-term
 7 potentiation of exogenous glutamate responses at single dendritic spines.
 8 *Proceedings of the National Academy of Sciences of the United States of
 9 America*, 102(40), 14434–14439. <https://doi.org/10.1073/PNAS.0501956102>
- 10 Barco, A., Lopez de Armentia, M., & Alarcon, J. M. (2008). Synapse-specific
 11 stabilization of plasticity processes: The synaptic tagging and capture
 12 hypothesis revisited 10 years later. *Neuroscience & Biobehavioral Reviews*,
 13 32(4), 831–851. <https://doi.org/10.1016/J.NEUBIOREV.2008.01.002>
- 14 Bartol, T. M., Bromer, C., Kinney, J., Chirillo, M. A., Bourne, J. N., Harris, K. M., &
 15 Sejnowski, T. J. (2015). Nanoconnectomic upper bound on the variability of
 16 synaptic plasticity. *eLife*, 4(November), 1–18.
 17 <https://doi.org/10.7554/eLife.10778>
- 18 Bi, G., & Poo, M. (1998). Synaptic Modifications in Cultured Hippocampal Neurons:
 19 Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type.
 20 *Journal of Neuroscience*, 18(24), 10464–10472.
 21 <https://doi.org/10.1523/JNEUROSCI.18-24-10464.1998>
- 22 Bittner, K. C., Grienberger, C., Vaidya, S. P., Milstein, A. D., Macklin, J. J., Suh, J.,
 23 Tonegawa, S., & Magee, J. C. (2015). Conjunctive input processing drives
 24 feature selectivity in hippocampal CA1 neurons. *Nature Neuroscience*, 18(8),
 25 1133–1142. <https://doi.org/10.1038/nn.4062>
- 26 Bittner, K. C., Milstein, A. D., Grienberger, C., Romani, S., & Magee, J. C. (2017).
 27 Behavioral time scale synaptic plasticity underlies CA1 place fields. *Science*,
 28 357(6355), 1033–1036. <https://doi.org/10.1126/science.aan3846>
- 29 Burrell, B. D., & Li, Q. (2008). Co-induction of long-term potentiation and long-term
 30 depression at a central synapse in the leech. *Neurobiology of Learning and
 31 Memory*, 90(1), 275–279. <https://doi.org/10.1016/j.nlm.2007.11.004>
- 32 Caya-Bissonnette, L., Naud, R., & Béïque, J.-C. (2023). Cellular Substrate of
 33 Eligibility Traces. *BioRxiv*, 2023.06.29.547097.
 34 <https://doi.org/10.1101/2023.06.29.547097>
- 35 Chindemi, G., Abdellah, M., Amsalem, O., Benavides-Piccione, R., Delattre, V.,
 36 Doron, M., Ecker, A., King, J., Kumbhar, P., Monney, C., Perin, R., Rössert, C.,
 37 Geit, W. Van, DeFelipe, J., Graupner, M., Segev, I., Markram, H., & Muller, E.
 38 (2020). A calcium-based plasticity model predicts long-term potentiation and
 39 depression in the neocortex. *BioRxiv*, 2020.04.19.043117.
 40 <https://doi.org/10.1101/2020.04.19.043117>
- 41 Cho, K., Aggleton, J. P., Brown, M. W., & Bashir, Z. I. (2001). An experimental test of
 42 the role of postsynaptic calcium levels in determining synaptic strength using
 43 perirhinal cortex of rat. *The Journal of Physiology*, 532(Pt 2), 459.
 44 <https://doi.org/10.1111/J.1469-7793.2001.0459F.X>
- 45 Clopath, C., Ziegler, L., Vasilaki, E., Büsing, L., & Gerstner, W. (2008). Tag-Trigger-
 46 Consolidation: A Model of Early and Late Long-Term-Potentiation and
 47 Depression. *PLOS Computational Biology*, 4(12), e1000248.
 48 <https://doi.org/10.1371/JOURNAL.PCBI.1000248>
- 49 Coesmans, M., Weber, J. T., De Zeeuw, C. I., & Hansel, C. (2004). Bidirectional
 50 Parallel Fiber Plasticity in the Cerebellum under Climbing Fiber Control. *Neuron*,
 51 44(4), 691–700. <https://doi.org/10.1016/j.neuron.2004.10.031>

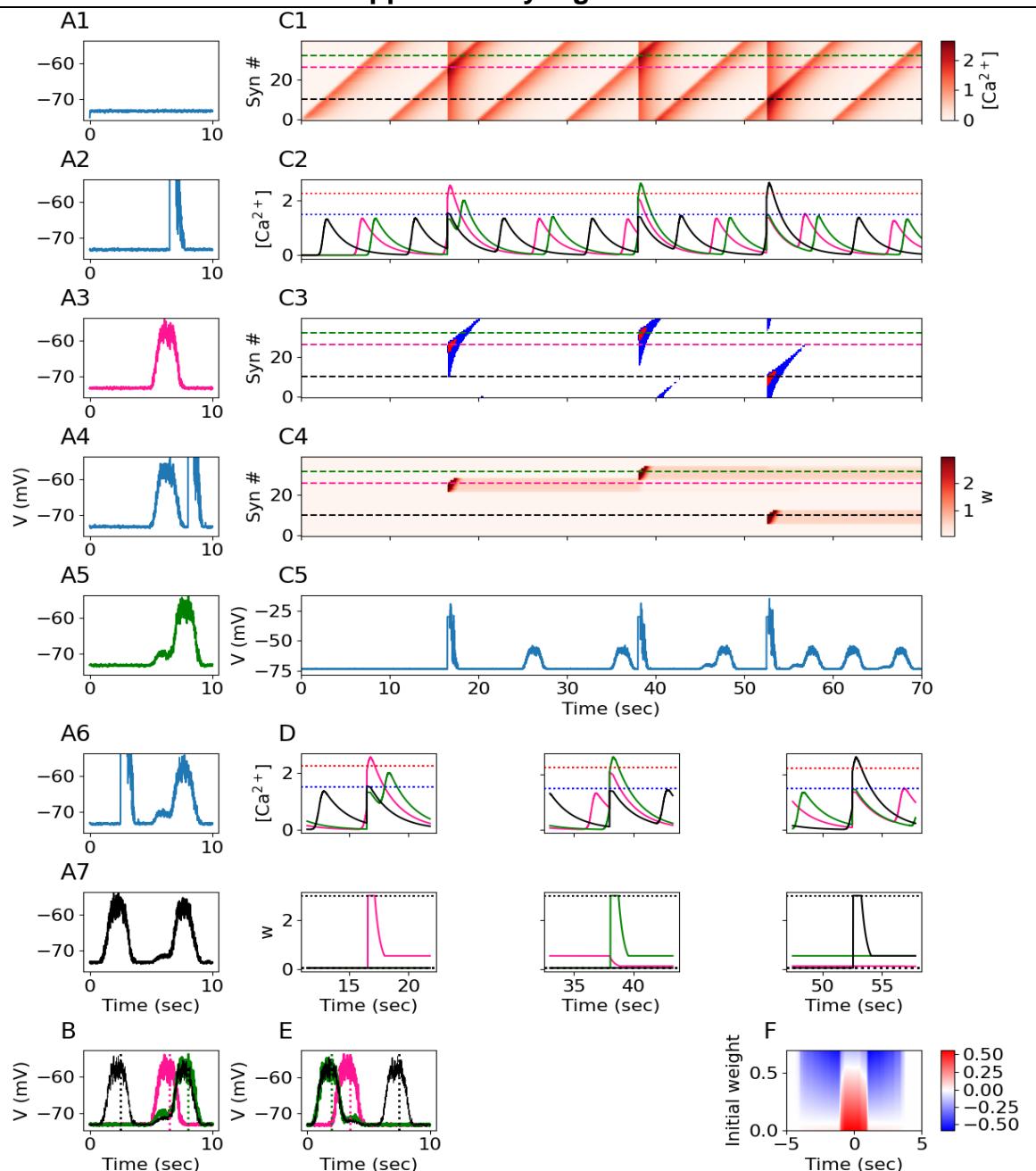
- 1 Cone, I., & Shouval, H. Z. (2021). Behavioral Time Scale Plasticity of Place Fields:
 2 Mathematical Analysis. *Frontiers in Computational Neuroscience*, 15, 19.
 3 [https://doi.org/10.3389/FNCOM.2021.640235/BIBTEX](https://doi.org/10.3389/FNCOM.2021.640235)
- 4 Cummings, J. A., Mulkey, R. M., Nicoll, R. A., & Malenka, R. C. (1996). Ca₂₊
 5 signaling requirements for long-term depression in the hippocampus. *Neuron*,
 6 16(4), 825–833. [https://doi.org/10.1016/S0896-6273\(00\)80102-6](https://doi.org/10.1016/S0896-6273(00)80102-6)
- 7 Ebner, C., Clopath, C., Jedlicka, P., & Cuntz, H. (2019). Unifying Long-Term
 8 Plasticity Rules for Excitatory Synapses by Modeling Dendrites of Cortical
 9 Pyramidal Neurons. *Cell Reports*, 29(13), 4295-4307.e6.
 10 <https://doi.org/10.1016/J.CELREP.2019.11.068>
- 11 Enoki, R., Hu, Y., Hamilton, D., & Fine, A. (2009). Expression of Long-Term Plasticity
 12 at Individual Synapses in Hippocampus Is Graded, Bidirectional, and Mainly
 13 Presynaptic: Optical Quantal Analysis. *Neuron*, 62(2), 242–253.
 14 <https://doi.org/10.1016/j.neuron.2009.02.026>
- 15 Evans, R., & Blackwell, K. (2015). Calcium: Amplitude, Duration, or Location? *The
 16 Biological Bulletin*, 228(1), 75. <https://doi.org/10.1086/BBLV228N1P75>
- 17 Frey, U., & Morris, R. G. M. (1997). Synaptic tagging and long-term potentiation.
 18 *Nature*, 385(6616), 533–536. <https://doi.org/10.1038/385533A0>
- 19 Gerstner, W., Lehmann, M., Liakoni, V., Corneil, D., & Brea, J. (2018). Eligibility
 20 Traces and Plasticity on Behavioral Time Scales: Experimental Support of
 21 NeoHebbian Three-Factor Learning Rules. *Frontiers in Neural Circuits*, 12, 53.
 22 [https://doi.org/10.3389/FNCIR.2018.00053/BIBTEX](https://doi.org/10.3389/FNCIR.2018.00053)
- 23 Graupner, M., & Brunel, N. (2012). Calcium-based plasticity model explains
 24 sensitivity of synaptic changes to spike pattern, rate, and dendritic location.
 25 *Proceedings of the National Academy of Sciences of the United States of
 26 America*, 109(10), 3991–3996. <https://doi.org/10.1073/pnas.1109359109>
- 27 Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P.,
 28 Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus,
 29 M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe,
 30 M., Peterson, P., ... Oliphant, T. E. (2020). Array programming with NumPy.
 31 *Nature*, 585(7825), 357–362. <https://doi.org/10.1038/S41586-020-2649-2>
- 32 Hebb, D. (1949). *The organization of behavior. A neuropsychological theory*.
- 33 Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science
 34 and Engineering*, 9(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
- 35 Inglebert, Y., Aljadeff, J., Brunel, N., & Debanne, D. (2020). Synaptic plasticity rules
 36 with physiological calcium levels. *Proceedings of the National Academy of
 37 Sciences of the United States of America*, 117(52), 33639–33648.
 38 [https://doi.org/10.1073/PNAS.2013663117/SUPPL_FILE/PNAS.2013663117.SA
 PP.PDF](https://doi.org/10.1073/PNAS.2013663117/SUPPL_FILE/PNAS.2013663117.SA

 39 PP.PDF)
- 40 Inglebert, Y., & Debanne, D. (2021). Calcium and Spike Timing-Dependent Plasticity.
 41 *Frontiers in Cellular Neuroscience*, 15.
 42 <https://doi.org/10.3389/FNCEL.2021.727336>
- 43 Jain, A., Nakahata, Y., Watabe, T., Rusina, P., South, K. E., Adachi, K., Yan, L.,
 44 Simorowski, N., Furukawa, H., & Yasuda, R. (2023). Dendritic, delayed, and
 45 stochastic CaMKII activation underlies behavioral time scale plasticity in CA1
 46 synapses. *BioRxiv*, 2023.08.01.549180.
 47 <https://doi.org/10.1101/2023.08.01.549180>
- 48 Kauderer, B. S., & Kandel, E. R. (2000). Capture of a protein synthesis-dependent
 49 component of long-term depression. *Proceedings of the National Academy of
 50 Sciences of the United States of America*, 97(24), 13342.
 51 <https://doi.org/10.1073/PNAS.97.24.13342>
- 52 Li, G., McLaughlin, D. W., & Peskin, C. S. (2023). A Biochemical Description of
 53 Postsynaptic Plasticity – with Timescales Ranging from Milliseconds to
 54 Seconds. *BioRxiv*, 2023.07.09.548255.
 55 <https://doi.org/10.1101/2023.07.09.548255>

- 1 Linden, D. J. (1996). A Protein Synthesis–Dependent Late Phase of Cerebellar Long-
 2 Term Depression. *Neuron*, 17(3), 483–490. [https://doi.org/10.1016/S0896-6273\(00\)80180-4](https://doi.org/10.1016/S0896-6273(00)80180-4)
- 3 Lisman, J. (1989). A mechanism for the Hebb and the anti-Hebb processes
 4 underlying learning and memory. *Proceedings of the National Academy of Sciences*, 86(23), 9574–9578. <https://doi.org/10.1073/pnas.86.23.9574>
- 5 Lisman, J. E. (2001). Three Ca²⁺ levels affect plasticity differently: the LTP zone, the
 6 LTD zone and no man's land. *The Journal of Physiology*, 532(2), 285–285.
 7 <https://doi.org/10.1111/J.1469-7793.2001.0285F.X>
- 8 Liu, K. K. L., Hagan, M. F., & Lisman, J. E. (2017). Gradation (approx. 10 size states)
 9 of synaptic strength by quantal addition of structural modules. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1715).
 10 <https://doi.org/10.1098/RSTB.2016.0328>
- 11 Majewska, A., Brown, E., Ross, J., & Yuste, R. (2000). Mechanisms of Calcium
 12 Decay Kinetics in Hippocampal Spines: Role of Spine Calcium Pumps and
 13 Calcium Diffusion through the Spine Neck in Biochemical Compartmentalization.
 14 *The Journal of Neuroscience*, 20(5), 1722.
 15 <https://doi.org/10.1523/JNEUROSCI.20-05-01722.2000>
- 16 Malenka, R. C., Kauer, J. A., Perkel, D. J., Mauk, M. D., Kelly, P. T., Nicoll, R. A., &
 17 Waxham, M. N. (1989). An essential role for postsynaptic calmodulin and
 18 protein kinase activity in long-term potentiation. 340(6234), 554–557.
- 19 Manahan-Vaughan, D., Kulla, A., & Frey, J. U. (2000). Requirement of translation but
 20 not transcription for the maintenance of long-term depression in the CA1 region
 21 of freely moving rats. *The Journal of Neuroscience : The Official Journal of the Society for Neuroscience*, 20(22), 8572–8576.
 22 <https://doi.org/10.1523/JNEUROSCI.20-22-08572.2000>
- 23 Milstein, A. D., Li, Y., Bittner, K. C., Grienberger, C., Soltesz, I., Magee, J. C., &
 24 Romani, S. (2021). Bidirectional synaptic plasticity rapidly modifies hippocampal representations. *ELife*, 10. <https://doi.org/10.7554/ELIFE.73046>
- 25 Miyazaki, K., & Ross, W. N. (2022). Fast Synaptically Activated Calcium and Sodium Kinetics in Hippocampal Pyramidal Neuron Dendritic Spines. *Eneuro*, 9(6), ENEURO.0396-22.2022. <https://doi.org/10.1523/ENEURO.0396-22.2022>
- 26 Mulkey, R. M., Endo, S., Shenolikar, S., & Malenka, R. C. (1994). Involvement of a calcineurin/ inhibitor-1 phosphatase cascade in hippocampal long-term depression. *Nature* 1994 369:6480, 369(6480), 486–488.
 27 <https://doi.org/10.1038/369486a0>
- 28 Neveu, D., & Zucker, R. S. RS. (1996). Postsynaptic Levels of [Ca²⁺]i Needed to Trigger LTD and LTP. *Neuron*, 16(3), 619–629. [https://doi.org/10.1016/S0896-6273\(00\)80081-1](https://doi.org/10.1016/S0896-6273(00)80081-1)
- 29 O'Connor, D. H., Wittenberg, G. M., & Wang, S. S. H. (2005a). Graded bidirectional synaptic plasticity is composed of switch-like unitary events.
- 30 O'Connor, D. H., Wittenberg, G. M., & Wang, S. S.-H. (2005b). Dissection of Bidirectional Synaptic Plasticity Into Saturable Unidirectional Processes. *Journal of Neurophysiology*, 94(2), 1565–1573. <https://doi.org/10.1152/jn.00047.2005>
- 31 O'Hare, J. K., Gonzalez, K. C., Herrlinger, S. A., Hirabayashi, Y., Hewitt, V. L., Blockus, H., Szoboszlay, M., Rolotti, S. v., Geiller, T. C., Negrean, A., Chelur, V., Polleux, F., & Losonczy, A. (2022). Compartment-specific tuning of dendritic feature selectivity by intracellular Ca²⁺ release. *Science*, 375(6586).
 32 <https://doi.org/10.1126/science.abm1670>
- 33 Petersen, C. C. H., Malenka, R. C., Nicoll, R. A., & Hopfield, J. J. (1998). All-or-none potentiation at CA3-CA1 synapses. *Proceedings of the National Academy of Sciences*, 95(8), 4732–4737. <https://doi.org/10.1073/PNAS.95.8.4732>
- 34 Piochon, C., Titley, H. K., Simmons, D. H., Grasselli, G., Elgersma, Y., & Hansel, C. (2016). Calcium threshold shift enables frequency-independent control of plasticity by an instructive signal. 46, 13221–13226.
- 35

- 1 R, M., H, S., & RW, T. (1989). Inhibition of postsynaptic PKC or CaMKII blocks
 2 induction but not expression of LTP. *Science (New York, N.Y.)*, 245(4920), 862–
 3 866. <https://doi.org/10.1126/SCIENCE.2549638>
- 4 Redondo, R. L., Okuno, H., Spooner, P. A., Frenguelli, B. G., Bito, H., & Morris, R. G.
 5 M. (2010). Synaptic Tagging and Capture: Differential Role of Distinct
 6 Calcium/Calmodulin Kinases in Protein Synthesis-Dependent Long-Term
 7 Potentiation. *Journal of Neuroscience*, 30(14), 4981–4989.
 8 <https://doi.org/10.1523/JNEUROSCI.3140-09.2010>
- 9 RM, M., CE, H., & RC, M. (1993). An essential role for protein phosphatases in
 10 hippocampal long-term depression. *Science (New York, N.Y.)*, 261(5124),
 11 1051–1055. <https://doi.org/10.1126/SCIENCE.8394601>
- 12 Rodrigues, Y. E., Tigaret, C. M., Marie, H., O'Donnell, C., & Veltz, R. (2021). A
 13 stochastic model of hippocampal synaptic plasticity with geometrical readout of
 14 enzyme dynamics. *BioRxiv*, 2021.03.30.437703.
 15 <https://doi.org/10.1101/2021.03.30.437703>
- 16 Rose, C. R., & Konnerth, A. (2001). Stores Not Just for Storage: Intracellular Calcium
 17 Release and Synaptic Plasticity. *Neuron*, 31(4), 519–522.
 18 [https://doi.org/10.1016/S0896-6273\(01\)00402-0](https://doi.org/10.1016/S0896-6273(01)00402-0)
- 19 Sajikumar, S., Navakkode, S., Sacktor, T. C., & Frey, J. U. (2005). Synaptic Tagging
 20 and Cross-Tagging: The Role of Protein Kinase M ζ in Maintaining Long-Term
 21 Potentiation But Not Long-Term Depression. *Journal of Neuroscience*, 25(24),
 22 5750–5756. <https://doi.org/10.1523/JNEUROSCI.1104-05.2005>
- 23 Sanes, J. R., & Lichtman, J. W. (1999). Can molecules explain long-term
 24 potentiation? *Nature Neuroscience* 1999 2:7, 2(7), 597–604.
 25 <https://doi.org/10.1038/10154>
- 26 Shouval, H. Z., Bear, M. F., & Cooper, L. N. (2002). A unified model of NMDA
 27 receptor-dependent bidirectional synaptic plasticity. *Proceedings of the National
 28 Academy of Sciences*, 99(16), 10831–10836.
 29 <https://doi.org/10.1073/pnas.152343099>
- 30 Shouval, H. Z., Wang, S. S. H., & Wittenberg, G. M. (2010). Spike timing dependent
 31 plasticity: A consequence of more fundamental learning rules. *Frontiers in
 32 Computational Neuroscience*, 4(1), 19.
 33 <https://doi.org/10.3389/fncom.2010.00019>
- 34 Tigaret, C. M., Olivo, V., Sadowski, J. H. L. P., Ashby, M. C., & Mellor, J. R. (2016).
 35 Coordinated activation of distinct Ca $^{2+}$ sources and metabotropic glutamate
 36 receptors encodes Hebbian synaptic plasticity. *Nature Communications*, 7.
 37 <https://doi.org/10.1038/ncomms10289>
- 38 Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau,
 39 D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J.,
 40 Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E.,
 41 Kern, R., Larson, E., ... Vázquez-Baeza, Y. (2020). SciPy 1.0: fundamental
 42 algorithms for scientific computing in Python. *Nature Methods*, 17(3), 261–272.
 43 <https://doi.org/10.1038/S41592-019-0686-2>
- 44 Yang, S., Tang, Y., Zucker, R. S., Yang, S.-N., Tang, Y.-G., & Zucker Selective, R. S.
 45 (1999). Selective Induction of LTP and LTD by Postsynaptic [Ca 2] i Elevation.
 46

1

Supplementary Figure S1**S1: Behavioral Time Scale Plasticity ‘Mirror Image’ Experiment**

(A1-A7) Voltage traces for 7 laps, each lap lasts for 10 seconds. A plateau potential is induced in the 2nd, 4th, and 6th laps. Green, pink, and black traces indicate place fields observed on the 3rd, 5th, and 7th laps, respectively. **(B)** Overlay of voltage traces for place fields from the 3rd (pink), 5th (green), and 7th (black) laps. Vertical dashed lines indicate plateau induction location during the preceding lap. **(C1)** Total $[Ca^{2+}]$ per synapse (rows, 40 synapses) over the course of all 7 laps. **(C2)** $[Ca^{2+}]$ over time for three synapses whose receptive fields are centered at the location of the first (pink - 15) second (green) or third (black) plateau induction. Boxes show calcium traces around the time of plateau induction (see (D)). **(C3)** Plasticity bar codes for each synapse (rows) over all laps. **(C4)** Weights over time for each synapse over all laps. **(C5)** Voltage over time, as in A, for all 7 laps. **(D)** Zoom-in on the calcium traces (Top) and weights (Bottom) from the three synapses shown in B2 at the time of each plateau induction. In the second lap (left), the pink synapse’s $[Ca^{2+}]$ rises beyond θ_P , inducing potentiation toward the maximum strength (dotted horizontal line at top), and the green synapse’s $[Ca^{2+}]$ rises above θ_D , although the green synapse is already at the minimum strength (dotted horizontal line on bottom) so it can’t depress any further. In the fourth lap (middle), the green synapse’s $[Ca^{2+}]$ rises beyond θ_P , inducing potentiation, and the pink and black synapses’ $[Ca^{2+}]$ rises above θ_D , but only the pink synapse can be depressed, because it was previously potentiated. In the sixth lap (right), the black synapse’s $[Ca^{2+}]$ rises beyond θ_P , inducing potentiation (the green synapse is above θ_D for a brief duration, but insufficiently long to substantially depress). **(E)** Result of a ‘mirror image’ experiment, where the locations of plateau inductions are reversed (see Figure 8). Green, pink, and black traces indicate locations of inductions at the 3rd, 5th, and 7th laps, as in B. **(F)** Change in weights as a function of initial weight and receptive field distance from plateau onset.

2

The Calcitron: A Simple Neuron Model That Implements Many Learning Rules via the Calcium Control Hypothesis

Toviah Moldwin¹, Li Shay Azran^{1,2}, Idan Segev^{1,3}

¹Edmond and Lily Safra Center for Brain Sciences, The Hebrew University of Jerusalem

²Department of Brain Sciences, Weizmann Institute of Science, Rehovot, Israel.

³Department of Neurobiology, The Hebrew University of Jerusalem

Correspondence:

Toviah Moldwin

Toviah.moldwin@mail.huji.ac.il

Keywords: **Synaptic weights, synaptic plasticity, calcium-based plasticity, neural plasticity, perceptron, machine learning, neural computation, Hebbian, homeostatic**

Abstract

Theoretical neuroscientists and machine learning researchers have proposed a variety of learning rules for linear neuron models to enable artificial neural networks to accomplish supervised and unsupervised learning tasks. It has not been clear, however, how these theoretically-derived rules relate to biological mechanisms of plasticity that exist in the brain, or how the brain might mechanistically implement different learning rules in different contexts and brain regions. Here, we show that the calcium control hypothesis, which relates plastic synaptic changes in the brain to calcium concentration $[Ca^{2+}]$ in dendritic spines, can reproduce a wide variety of learning rules, including some novel rules. We propose a simple, perceptron-like neuron model that has four sources of $[Ca^{2+}]$: local (following the activation of an excitatory synapse and confined to that synapse), heterosynaptic (due to activity of adjacent synapses), postsynaptic spike-dependent, and supervisor-dependent. By specifying the plasticity thresholds and amount of calcium derived from each source, it is possible to implement Hebbian and anti-Hebbian rules, one-shot learning, perceptron learning, as well as a variety of novel learning rules.

Introduction

Artificial neural networks (ANNs) have demonstrated a remarkable ability to solve both supervised and unsupervised learning tasks (Lecun et al., 2015). Because ANNs are inspired by a simple model of biological neurons and their synapses (McCulloch & Pitts, 1943), theoretical neuroscientists have used ANNs to explore questions about the brain. (Amit, 1992; Kriegeskorte, 2015; Memmesheimer et al., 2014; D. Rumelhart et al., 1986; Saxe et al., 2019). ANNs learn to solve a wide variety of problems by making use of “learning rules” whereby connection strengths between network nodes are modified (Hebb, 1949; Kohonen, 1982; Oja, 1982; Rosenblatt, 1958; D. E. Rumelhart et al., 1986). These learning rules are analogous to, and often inspired by, rules governing synaptic plasticity in biological neurons. However, it is often not apparent how biological plasticity mechanisms can result in the types of learning rules used in ANNs.

One of the dominant theories for how long-term synaptic plasticity operates in the brain is the calcium control hypothesis (Graupner & Brunel, 2012; J. Lisman, 1989; J. E. Lisman, 2001; J. Lisman & Goldring, 1988; Shouval et al., 2002, 2010). The calcium control hypothesis states that the magnitude and direction of change of synaptic strength is mediated by the intracellular calcium concentration ($[Ca^{2+}]$) at the synapse. In the classic version of the calcium control hypothesis, at low levels of $[Ca^{2+}]$, the synapse is unaffected, at medium levels of $[Ca^{2+}]$, the synapse is depressed, and at high levels of $[Ca^{2+}]$, the synapse is potentiated. There is also some evidence that at very high levels of $[Ca^{2+}]$, all plastic mechanisms are turned off (Tigaret et al., 2016). This description of the relationship between calcium concentration and plasticity fits with experimental evidence from hippocampus and cortex (Artola et al., 1990; J. Lisman, 1989). In cerebellar Purkinje cells however, the calcium thresholds for potentiation and depression seem to be reversed; medium concentrations of $[Ca^{2+}]$ cause potentiation and high concentrations lead to depression (Coesmans et al., 2004; Piochon et al., 2016). While many downstream molecular mechanisms (most notably CaMKII and calcineurin) are involved in mediating plasticity (Citri & Malenka, 2008; Sanes & Lichtman, 1999), the calcium control hypothesis remains one of the most parsimonious and effective theories for how synaptic plasticity works in the brain. It remains to be explained, however, how calcium control of plasticity can result in the incredible learning abilities of the brain, especially if we work with the assumption that the brain operates in a manner similar to artificial neural networks.

In this work, we aim to bridge the gap between learning rules in artificial neurons and networks and the biological mechanisms of synaptic plasticity. We propose a simple, perceptron-like threshold-linear neuron model, the *calciton*, that has four potential sources of local (synapse-specific) and global (common to all synapses) calcium. By adjusting the amount of calcium obtained from each calcium source and the calcium thresholds for plasticity, we show that it is possible to implement a wide variety of learning rules, such as Hebbian and anti-Hebbian learning, frequency-dependent plasticity, homeostatic plasticity, the perceptron learning rule, one-shot “writable” neural memory, and more. We thereby demonstrate that calcium control of synaptic plasticity can be a highly versatile mechanism which enables neurons to implement many different “programs” for modifying their synapses and storing information.

Results

The Calciton Model

The calciton is a simple neuron model, akin to a McCulloch and Pitts (M&P) neuron, or perceptron, which applies a transfer function to the weighted sum of its inputs. Formally we have:

$$\hat{y} = g\left(\sum_{i=1}^N w_i x_i + b\right) \quad (1)$$

where \hat{y} is the output of the neuron, g is the transfer function, w_i is the weight of synapse i , x_i is the input to synapse i , N is the total number of synapses and b is a bias term. Depending on the particular use case, the transfer function g can be a simple threshold nonlinearity (e.g. a sign function), a sigmoid, a linear function, or any of the standard transfer functions used for artificial neural networks, so the set of possible outputs \hat{y} depend on the choice of transfer function. The weights w_i and inputs x_i of the calciton are generally restricted to be non-negative to maintain fidelity to the experimental literature of calcium control hypothesis which mostly focuses on excitatory synapses (Graupner & Brunel, 2012; J. Lisman, 1989; J. E. Lisman, 2001; J. Lisman & Goldring, 1988; Shouval et al., 2002, 2010). In some contexts, the inputs x_i will be restricted to be binary (0 or 1); in other contexts, the inputs can be any positive real value (i.e., a rate model). The synaptic weights are also bounded between a minimum and maximum strength (w_{min} and w_{max} , respectively) due to the nature of the calcium-based plasticity rule; this will be discussed further below. Because the weights and inputs are restricted in our model to be excitatory, the bias term b will generally be negative or 0, thus representing the aggregate inhibitory input to the neuron.

The calciton has four sources of calcium. The first source of calcium is the local calcium due to the presynaptic input at each synapse, C_{local} . The local calcium at synapse i is defined as:

$$C_{local}^i = \alpha x_i \quad (2)$$

where α is a non-negative coefficient that determines the marginal increase in spine calcium for a unitary increase in input magnitude. Biologically, C_{local} can be thought of as the calcium that enters a dendritic spine receiving excitatory input through its NMDA receptors during synaptic stimulation. We note that according to the above formulation, the local calcium does *not* depend on the synaptic weight, w_i , only on the synaptic input, x_i . This decision is motivated by both biological and computational considerations. Biologically, synaptic strength is a consequence of AMPA receptor conductance (Citri & Malenka, 2008) while plasticity-inducing calcium primarily via NMDA receptors. Thus, while changing the synaptic weight (i.e., the number of AMPA receptors) will influence the somatic depolarization observed for a given presynaptic input, it will not necessarily change the calcium influx via the NMDA receptors. (It is true that the NMDA receptor's conductance is also voltage-dependent (Jahr et al., 1990; Jahr & Stevens, 1990), so the increase of AMPA conductance of the synapse, as well as the weighted input from other synapses, can indirectly affect the calcium influx by depolarizing the neuron. We model the aggregate calcium influx due to input-dependent depolarization as heterosynaptic calcium, see next paragraph and **Discussion**). The weight-independence of the local calcium influx has the computational advantage of avoiding feedback loops – if calcium influx was weight-dependent, potentiating or depressing a synapse would change the synapse's sensitivity to plasticity protocols.

The second source of intracellular calcium in the Calciton is calcium that globally enters dendritic spines due to the aggregate activity of all nearby synaptic inputs, resulting in the activation of voltage-gated calcium channels (VGCCs) in regions of the dendrite that are sufficiently depolarized (J. E. Lisman, 2001; Moldwin, Kalmenson, et al., 2023) (and by increasing the conductance of NMDA at active synapses via NMDA's voltage dependence, see **Discussion**). We call this calcium source heterosynaptic calcium, or C_{het} . This calcium is responsible for heterosynaptic plasticity, i.e. plasticity that can be induced at non-activated synapses by presynaptic stimulation at other nearby synapses. While heterosynaptic plasticity

is spatially sensitive, for simplicity we assume that synaptic activity is distributed uniformly on the neuron and we can thus approximate this calcium, C_{het} , by looking at the aggregate activity from all synaptic inputs. We thus have

$$C_{het} = \beta * (\sum_{i=1}^N w_i x_i) \quad (3)$$

where β is a coefficient that determines how much calcium enters each spine due to the overall depolarization of the dendritic membrane and x_i and w_i are as in Eq. (1).

The third source of calcium, C_{BAP} , comes from the backpropagating action potential, or BAP. When a neuron fires an action potential, the axonal/somatic spike backpropagates to the dendrites and depolarizes the dendritic membrane (Stuart & Sakmann, 1994), which can globally activate voltage-gated calcium channels at all spines. We model this as:

$$C_{BAP} = \gamma \hat{y} \quad (4)$$

Where γ is a coefficient that determines the amount of calcium that enters the postsynaptic spines due to each spike. (Here we ignore timing effects of the postsynaptic spike relative to the timing of the input; for the purposes of the calciton we assume that synaptic input and the spike it generates happen within a single time step. This assumption prevents the calciton from implementing spike-timing dependent plasticity (STDP (Bi & Poo, 1998)); a more detailed version of the calciton with temporal dynamics would be necessary to capture STDP.)

The fourth source of calcium, C_{sprv} , comes from an external supervisor, denoted as Z . Z may be binary or positive real-valued. In hippocampal pyramidal neurons, a likely candidate for this supervisory signal is strong input to the apical tuft, which can induce bursts of spikes at the soma, potentially leading to global calcium influx at VGCCs at basal dendrites (Bittner et al., 2015, 2017a; Grienberger & Magee, 2022a; Milstein et al., 2021). A similar calcium-based supervisory scheme exists in cerebellar Purkinje neurons, where strong input to the Purkinje neurons from climbing fibers can induce long-term depression of synapses between presynaptic parallel fibers and the postsynaptic Purkinje cell (Konnerth et al., 1992). (In our model, Z does not contribute directly to the neuron's output \hat{y} , because although the supervisory signal often does depolarize the neuron, this depolarization is treated as incidental to the plasticity induction, rather than as part of the neuron's input-output function. In other words, we assume that downstream neurons only care about "output spikes" rather than "plasticity plateaus\bursts"). The calcium influx due to this supervisory signal is defined as:

$$C_{sprv} = \delta Z \quad (5)$$

Where δ is the coefficient determining the amount of calcium that comes from the supervising signal. The total calcium per dendritic spine, C_{total}^i , is the sum of these four calcium sources:

$$C_{total}^i = C_{local}^i + C_{het} + C_{BAP} + C_{sprv} = \alpha x_i + \beta * (\sum_{i=1}^N w_i x_i) + \gamma \hat{y} + \delta Z \quad (6)$$

An equivalent way to write this equation is in terms of local and global calcium sources:

$$C_{total}^i = C_{local}^i + C_{global} \quad (7)$$

where:

$$C_{global} = C_{het} + C_{BAP} + C_{sprv} \quad (8)$$

This formulation emphasizes that global calcium signals from the total feed-forward depolarization, backpropagating action potential, and supervisor are broadcast equally to all synapses, thus the local calcium is needed to break the symmetry between different synapses at each time step.

Calcium-Based Plasticity for the Calciton

At each time step t of the calciton's operation, a vector of inputs $x = [x_1, x_2 \dots x_N]$ is presented to the calciton, and the calciton produces an output \hat{y} (Eq. (1)). The calciton calculates the calcium concentration per spine, C_{total}^i , from the inputs (x), output (\hat{y}), and the supervising signal (Z) at that time step (Eq. (6)). The calcium concentration at each dendritic spine (C_{total}^i) is used to determine the magnitude and direction of plastic change (if any) for that synapse's weight (w_i) at the next time step.

To implement calcium-based plastic changes to each synapse, we consider two versions of a calcium-based plasticity rule: a linear version and an asymptotic version. The latter rule is also referred to as the fixed point – learning rate (FPLR) rule. Although we will use the FPLR rule for most of the simulations in this study, we will first introduce the linear rule as it is simpler and provides a point of entry to the FPLR rule.

The linear version of calcium-based plasticity changes a synaptic weight by a fixed amount at each time step depending on the amount of calcium present at the synapse at that time step. Formally, following the notation of (Shouval et al., 2002) this may be described as

$$\Delta w_i(t) = w_i(t+1) - w_i(t) = \Omega(C_{total}^i(t)) \quad (9)$$

where $\Omega(C)$ is any function from positive-valued calcium concentrations to positive or negative changes in synaptic weight. As a simple representation of the classic calcium control paradigm observed in hippocampal and cortical cells, we can choose $\Omega(C)$ to be a step function with two thresholds: the depression threshold, θ_D , and the potentiation threshold, θ_P , where $\theta_D < \theta_P$. Our plasticity function $\Omega(C)$ returns 0 when the calcium is below θ_D , returns -0.01 when the calcium is between θ_D and θ_P (depressing the synapse by 0.01 units), and returns 0.1 when the calcium is above the θ_P (potentiating the synapse by 0.1 units). Formally this can be written as:

$$\Omega(C) = \begin{cases} 0, & Ca(t) < \theta_D \\ -0.01, & \theta_D \leq Ca(t) \leq \theta_P \\ 0.1, & Ca(t) > \theta_P \end{cases} \quad (10)$$

In the above formulation of the plasticity rule, synaptic weights can become arbitrarily large or small provided a sufficient number of plasticity events, including changing sign from positive (excitatory) to negative (inhibitory). This is not particularly biologically realistic, as synapses do not change sign from excitatory to inhibitory, and synaptic strengths are not observed to become arbitrarily large (there are also biophysical limitations on the maximum possible depolarization that can be achieved from a single synaptic input). We thus employ a different rule, inspired by the rules of Shouval and Graupner (Graupner & Brunel, 2012), wherein synaptic weights are modified asymptotically toward a minimum or maximum value (w_{min} and w_{max}), depending on the calcium concentration. This is known as the fixed point – learning rate (FPLR) rule, as the rule is specified by defining the fixed points of the weights (i.e. the asymptotes, w_{min} and w_{max} – also called F_D and F_P , respectively, as the depressive and potentiative fixed points) and learning rates (η_D and η_P for depression and potentiation, respectively) as a function of the calcium concentration (Moldwin, Azran, et al., 2023). In the absence of a substantial $[Ca^{2+}]$, i.e. ($Ca(t) < \theta_D$), the weight can also drift back to some baseline w_{drift} at a slower rate of η_{drift} . The FPLR rule for the calciton is formulated as:

$$\Delta w_i(t) = \eta(C_{total}^i(t)) * (F(C_{total}^i(t)) - w_i(t)) \quad (11)$$

where $\eta(C)$ defines the learning rate as a function of calcium and $F(C)$ defines the fixed points as a function of calcium. The learning rates $\eta(C)$ define the fraction of the difference between the present weight and the target fixed point that is traversed at each time step, resulting in asymptotic plasticity dynamics for a given level of calcium at a particular synapse. A standard two-threshold calcium control function would have the following structure:

$$F(C) = \begin{cases} w_{drift}, & Ca(t) < \theta_D \\ w_{min}, & \theta_D \leq Ca(t) \leq \theta_P \\ w_{max}, & Ca(t) > \theta_P \end{cases} \quad (12)$$

And

$$\eta(C) = \begin{cases} \eta_{drift}, & Ca(t) < \theta_D \\ \eta_D, & \theta_D \leq Ca(t) \leq \theta_P \\ \eta_P, & Ca(t) > \theta_P \end{cases} \quad (13)$$

For example, we might have:

$$F(C) = \begin{cases} 0.5, & Ca(t) < \theta_D \\ 0.25, & \theta_D \leq Ca(t) \leq \theta_P \\ 1, & Ca(t) > \theta_P \end{cases}$$

And

$$\eta(C) = \begin{cases} 0.04, & Ca(t) < \theta_D \\ 0.15, & \theta_D \leq Ca(t) \leq \theta_P \\ 0.25, & Ca(t) > \theta_P \end{cases}$$

This means that synaptic weights with pre-depressive calcium concentrations (C_{total}^i) eventually drift toward a “neutral” state of 0.5 at a rate of 0.04 synapses with a depressive calcium concentration will depress towards 0.25 at a rate of 0.15, and synapses with a potentiative C_{total}^i will be potentiated towards 1 at a rate of 0.25. In practice, for the purpose of this work, we will neglect the pre-depressive drift, i.e. the tendency of synapses to slowly drift back to baseline in the presence of very low levels of calcium ($C < \theta_D$). In the FPLR framework, this is accomplished by setting $\eta(C < \theta_D) = 0$.

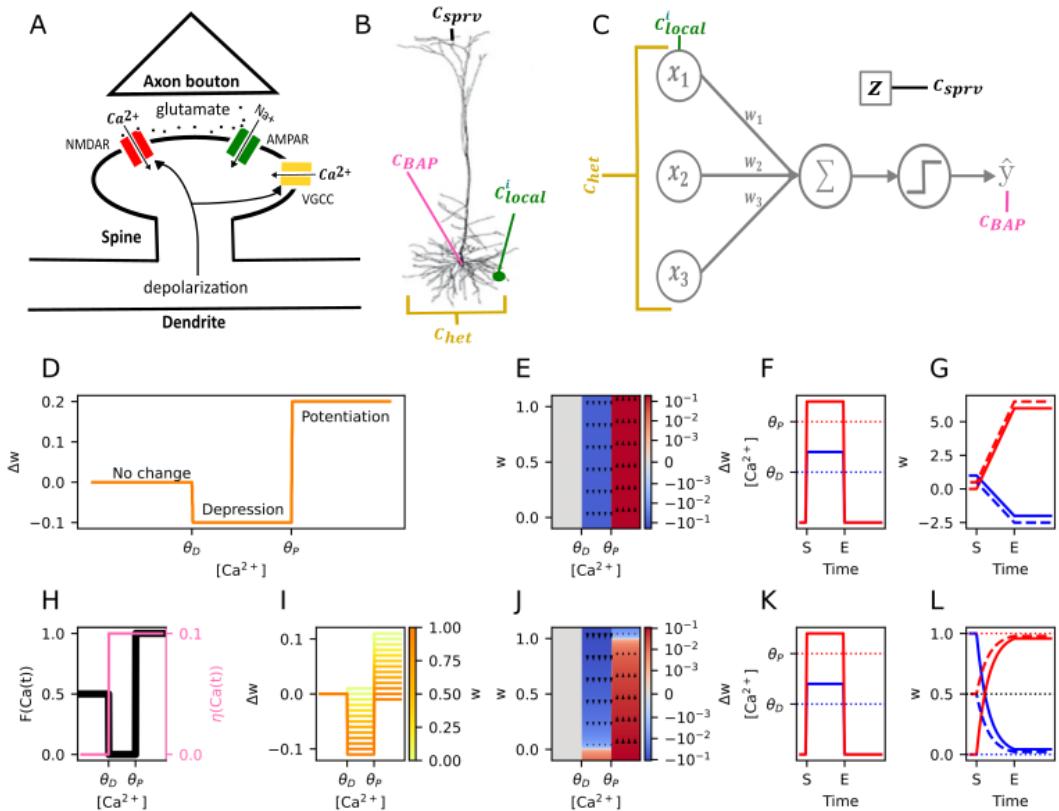


Figure 1: The Calcitron model and calcium-based plasticity rules. (A) Sources of Ca^{2+} at the synapse. Local glutamate release from an activated presynaptic axon binds to an NMDA receptor in the postsynaptic dendritic spine, enabling local Ca^{2+} influx. Depolarization of the neuron opens voltage-gated calcium channels (VGCCs), enabling calcium influx from global signals. (Glutamate also binds to AMPA receptors, enabling Na^{+} influx, and depolarization also affects NMDAR conductance.) (B) Possible sources of Ca^{2+} influx in a neuron. Ca^{2+} can enter due to presynaptic input (c_{local}^i), heterosynaptically-induced depolarization of VGCCs (c_{het}), the backpropagating action potential (C_{BAP}) or a supervisory signal, such as a somatic burst induced by input to the apical tuft (c_{sprv}). (C) The four Ca^{2+} sources in a point neuron model. Each Ca^{2+} source is associated with a respective coefficient ($\alpha, \beta, \gamma, \delta$) determining how much Ca^{2+} comes from each source. (D) The calcium control hypothesis. $[\text{Ca}^{2+}]$ between θ_D and θ_P induces depression, $[\text{Ca}^{2+}]$ above θ_P induces potentiation. (E) Weight change as a function of calcium in the linear version of Ca^{2+} -based plasticity, as in (D), shown as phase plane. Magnitude of weight change is independent of current weight. Blue indicates depression, red indicates potentiation, white indicates no change. (F) Step stimulus to show the plastic effect of different levels of $[\text{Ca}^{2+}]$. $[\text{Ca}^{2+}]$ is either raised to a depressive level ($\theta_D \leq [\text{Ca}^{2+}] \leq \theta_P$, blue line) or to a potentiative level ($[\text{Ca}^{2+}] > \theta_P$, red line) for several timesteps, then reduced to 0. S and E refer to start and end of stimulus. (G) Dynamics of the linear rule in response to the canonical stimulus from (F). Synaptic weights increase or decrease linearly in response to the potentiative or depressive levels of calcium (red and blue traces, respectively), then remain stable after calcium is turned off. (H) Fixed points (black) and learning rates (pink) in the asymptotic fixed point – learning rate (FPLR) version of the calcium control hypothesis. (I) Weight change as a function of $[\text{Ca}^{2+}]$ for different values of the current synaptic weight. Darker colors indicate higher weights. (J) Phase plane of weight changes for the FPLR rule. (K) Stimulus to demonstrate FPLR rule, identical to F. (L) Dynamics of the FPLR rule. Synaptic weights potentiate or depress asymptotically toward the potentiative or depressive fixed point.

Calcitron leaning in a binary model: Hebbian, Anti-Hebbian, and other Pre-Post Learning Rules

As simple illustration of the ability of the Calcitron to implement learning rules, we consider Hebbian learning. One simple formulation of the Hebb rule is “neurons that fire together, wire together”. In other words, presynaptic inputs that were active at the same time as a postsynaptic spike are potentiated. In the Calcitron, if we assume binary inputs and outputs (i.e. x_i and \hat{y} are either 0 or 1), this can be accomplished by setting α and γ in Eq. (6) to both be below θ_D , while enforcing that $\alpha + \gamma$ is above θ_P . This ensures that an active synapse not accompanied by a spike does not change (because $\alpha * 1 + \gamma * 0 < \theta_D$), nor does an inactive synapse that is accompanied by a spike (because $\alpha * 0 + \gamma * 1 < \theta_D$), but an active synapse accompanied by a spike will potentiate (because $\alpha * 1 + \gamma * 1 > \theta_P$) (Fig. 2A1).

Another version of the Hebbian learning rule can be stated as “fire together, wire together; out of sync, lose your link”. In other words, synapses that were active at the same time as a postsynaptic spike are potentiated, as before, but now we penalize (via depression) synapses that were active at a time when there was no postsynaptic spike, as well as synapses that were inactive at a time when a postsynaptic spike did occur. In the Calcitron, this is accomplished by setting α and γ to individually be in the depressive region (between θ_D and θ_P), which penalizes ‘out of sync’ synapses, while still maintaining $\alpha + \gamma$ is above θ_P to enforce “fire together wire together” behavior (Fig. 2A2).

It is also possible to obtain an anti-Hebbian “fire together, lose your link” plasticity rule by setting α and γ to individually be below θ_D , while enforcing that $\theta_D < \alpha + \gamma < \theta_P$. However, it is *not* possible, using the standard plasticity thresholds, to get an anti-Hebbian rule that rewards out-of-sync synapses, i.e. “fire together, lose your link, out of sync wire together”, as a synapse that fires synchronously with the postsynaptic spike will always have a higher C_{total}^i than a synapse that fires asynchronously, and thus cannot potentiate. In other words, if $\theta_D < \alpha + \gamma < \theta_P$, necessarily it holds that $\alpha < \theta_P$ and $\gamma < \theta_P$ (Fig 2A3).

However, if the plasticity thresholds are reversed, i.e. $\theta_P < \theta_D$, as in Purkinje neurons, we can get “fire together, lose your link; out of sync, wire together” plasticity. We set α and γ to both be in the potentiation region (between θ_P and θ_D), while enforcing that $\alpha + \gamma$ is above θ_D (Fig. 2A4).

As a simple demonstration of these Hebbian and anti-Hebbian rules, we created a random sequence of binary input patterns (Fig. 2B1-2B4) consisting of 10 synaptic input lines (i.e., N=10) and presented this sequence to four different Calcitrons, each implementing one of four rules described above by applying different calcium source coefficients and plasticity thresholds. We then compare how the different Calcitrons yield different weight dynamics (Fig. 2C-2F).

In the “fire together, wire together” Calcitron (2A1-2F1), synaptic weights can only potentiate or stay stable, they can never depress. Every so often, just by chance, one of the random input patterns will be sufficiently large to generate a spike (2C1, red-headed stems). The calcium from this spike combined with the calcium at active synapses yields potentiation at the active synapses (2D1-2E1). This creates a positive feedback loop: when a synapse is potentiated, that makes the neuron more likely to elicit a spike whenever that synapse participates in a pattern (because they synapse provides a greater contribution to the overall input to the neuron). Eventually, this results in all synaptic weights becoming strongly potentiated (2F1) and the Calcitron spikes in response to almost all input patterns.

In the “fire together, wire together; out of sync, lose your link” Calcitron (2A2-2F2), the potentiation at active synapses whenever there is a spike is counterbalanced by depression at inactive synapses when a spike occurs, as well as at active synapses when no spike occurs. As such, the synaptic weights do not become overly strong and the neuron doesn’t spike as aggressively.

In the “fire together, lose your link” Calcitron (2A3-2F3), synaptic weights can only depress or stay stable, they can never potentiate. As such, initially, whenever the random input elicits a spike, the synapses that were active at that time step depress. However, here we have a negative feedback loop: once a synapse depresses, it contributes less to the neuron’s overall voltage, which makes the neuron less likely to spike, so eventually the synaptic weights stop depressing.

Finally, in the “fire together, lose your link; out of sync, wire together” Calcitron (2A4-2F4), we again have a balance between depression and potentiation for synchronized and unsynchronized activity, so we again observe more moderate changes in the synaptic weights over longer time horizons.

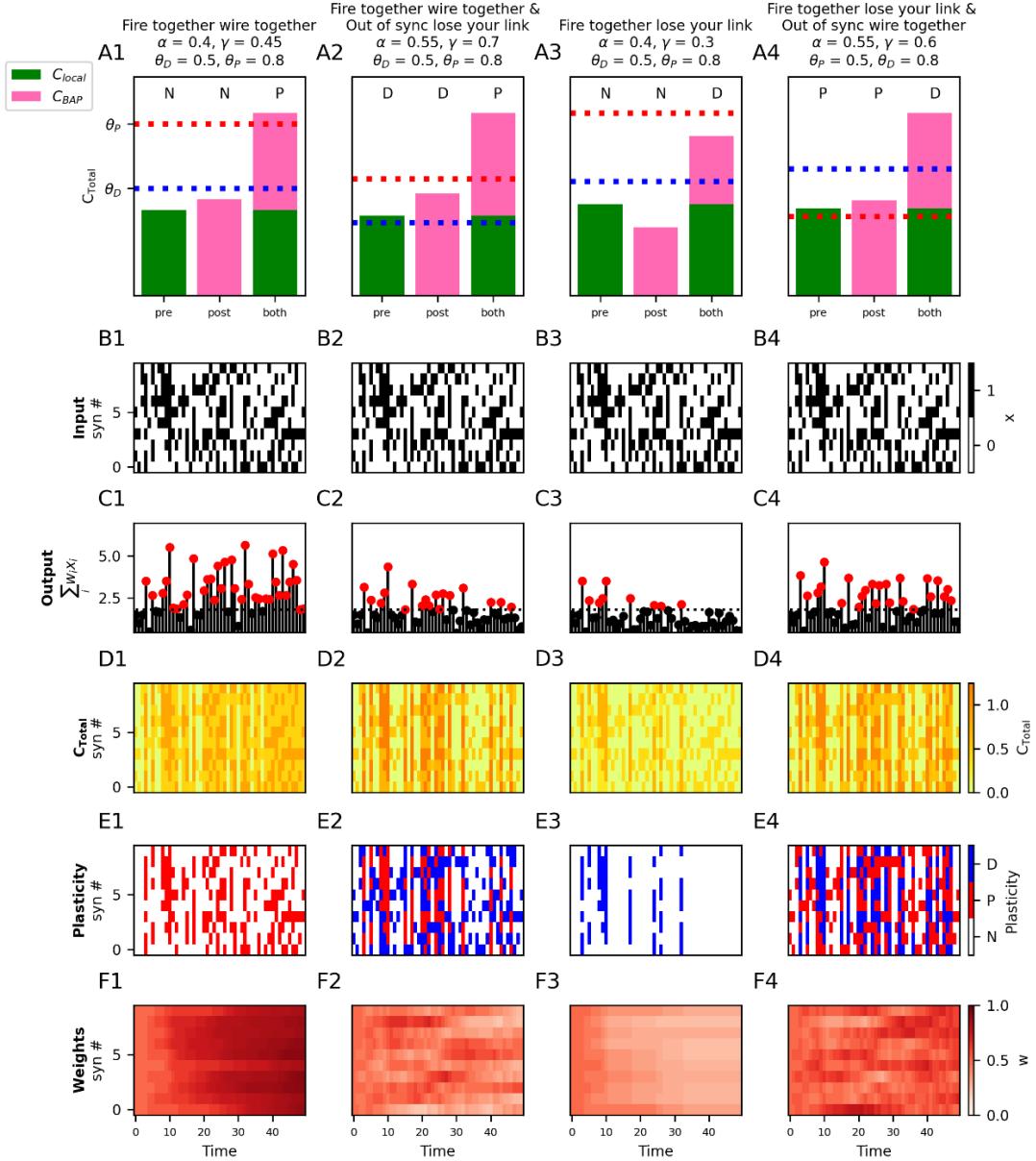


Figure 2: Four kinds of Hebbian and anti-Hebbian learning using Ca^{2+} . (A1-A4) Different versions of Hebbian and anti-Hebbian learning rules are implemented by setting the respective coefficients (α and γ in Eq. 6) for the local (C_{local}^i) and backpropagating spike-dependent (C_{BAP}) [Ca^{2+}]. For each rule, we show the direction of plastic change (indicated by the letters above the bars: “N”: no change, “D”: depression, “P”: potentiation) for three different conditions: a synapse with active presynaptic input in the absence of a postsynaptic spike (“pre”), a synapse without local input in the presence of a postsynaptic spike (“post”) and at a synapse with active presynaptic input and a postsynaptic spike (“both”). The total [Ca^{2+}] (C_{total}^i) for each condition is the sum of the local input-dependent [Ca^{2+}] (C_{local}^i , green) and the spike-dependent [Ca^{2+}] (C_{BAP} , pink). (When there is neither local input or a postsynaptic spike, the expected [Ca^{2+}] is 0) (B1 – B4) For each learning rule from (A1-A4), 10 random binary inputs (black: active, white: inactive) are presented to each synapse at each time step. (Inputs are identical for all learning rules). (C1–C4) Sum of weighted inputs at each time step for each learning rule shown in A1-A4 respectively. Dotted horizontal line indicates the spike threshold ($-b$ from equation 6). Outputs that are above the threshold (produce a postsynaptic spike) are indicated by a red circle. (D1 – D4) [Ca^{2+}] per synapse for each time step for the 4 learning rules shown in A1-A4 respectively. (E1 – E4) Bar codes indicating occurrence of potentiation (“P”, red), depression (“D”) or no change (“N”, white) shown in A1-A4 respectively. (F1–F4) Synaptic weights over the course of the simulation for A1-A4 cases respectively.

Importantly, it is possible to implement many more pre-post rules with calcium than just the standard Hebbian and anti-Hebbian rules. For example, we could have a “fire together wire together” rule that penalizes synapses that are active when there is no postsynaptic spike, but doesn’t penalize inactive synapses at the time when there is a postsynaptic spike (Figure 3, third row first column).

From a combinatoric standpoint, one can imagine 27 possible pre-post learning rules, because there are three possible synaptic scenarios (pre only, post only, pre and post) and there are three possible outcomes for each case (potentiate, depress, nothing). (If we allow for synapses to undergo plasticity when there is neither presynaptic nor postsynaptic activity, there are 81 possibilities, but we assume that synapses are stable in the absence of any activity.) However, the two-threshold calcium control hypothesis prohibits some of these scenarios, because the $[Ca^{2+}]$ from the “pre and post” scenario must be the sum of the $[Ca^{2+}]$ from the “pre only” and “post only” scenarios. This constraint leaves us with 14 potential pre-post rules that can be implemented in the calcitonin (Fig. 3).

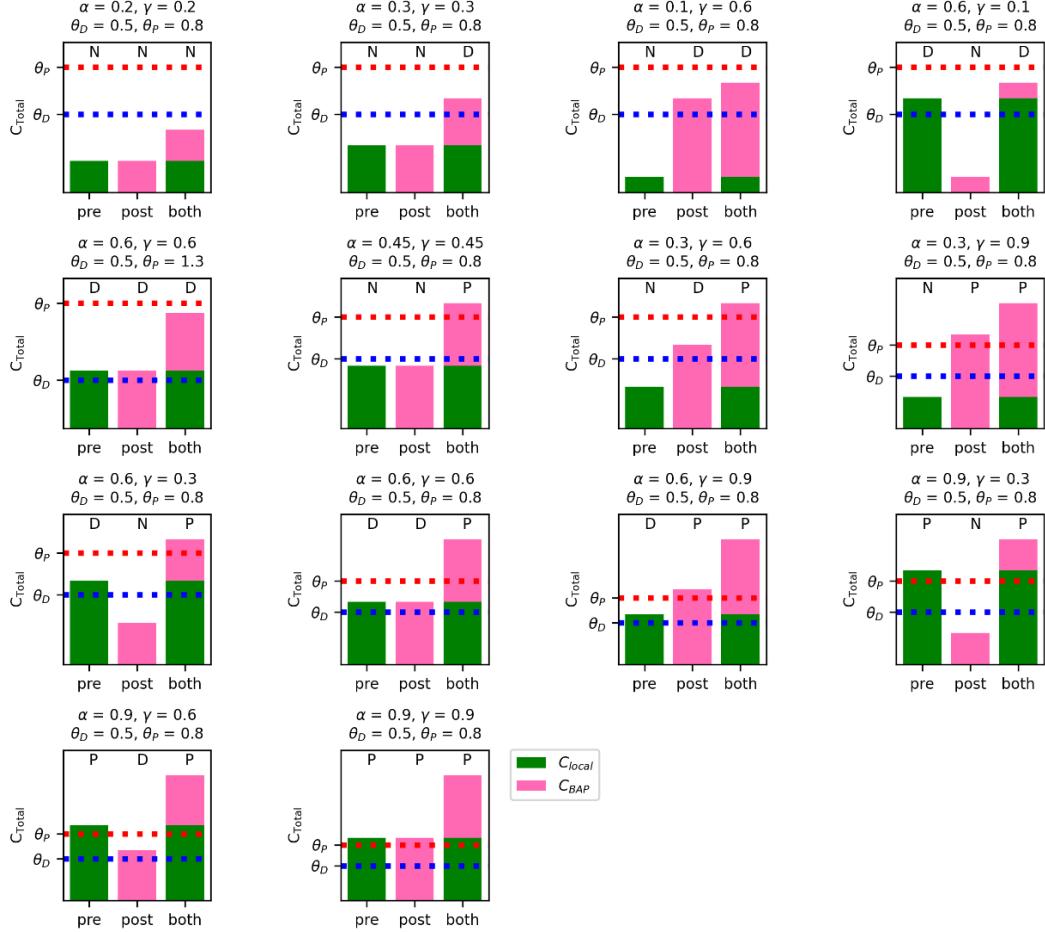


Figure 3: Fourteen possible plasticity rules for presynaptic input- and spike-dependent- calcium. Setting different coefficient values (α and γ in Eq. 6) for the local (C_{local}) and backpropagating spike-dependent (C_{BAP}) $[Ca^{2+}]$ can lead to fourteen possible learning rules in the case of binary input and output. Presynaptic input alone (standalone green bar in each subplot) can lead to potentiation (P), depression (D), or no change (N) at the active synapse, or a postsynaptic spike alone (standalone pink bar in each subplot) can lead to P, D, or N at non-activated synapses. When a postsynaptic spike occurs at the same time as a presynaptic input (stacked pink and green bars in each subplot), the resultant $[Ca^{2+}]$ at an activated synapse is the sum of the local and spike-dependent $[Ca^{2+}]$, constraining the number of qualitatively different rules to the 14 shown here (assuming $\theta_P > \theta_D$). Combinatorically, there are 27 possible rules (every combination of {P, D, N} with {pre, post, and both}), however only the 14 listed are consistent with the calcium control hypotheses.)

Frequency-dependent pre-post learning rules in a rate model

In the previous section, we assumed binary inputs and outputs, (i.e. x_i and \hat{y} are either 0 or 1). If we instead consider a rate model, where x_i and \hat{y} represent the firing rates of the presynaptic and postsynaptic neuron, respectively, it is possible to implement frequency-dependent plasticity whose outcome depends on both the pre-synaptic and postsynaptic firing rate (Fig. 4A). Experimentally, we note that presynaptic-only frequency-dependent plasticity has been observed, where low frequency stimulation causes depression and high frequency stimulation causes potentiation (Bliss & Lomo, 1973; Dudek & Bear, 1992; O'Connor et al., 2005a). To replicate classical (presynaptic only) frequency-dependent plasticity, we set all

coefficients other than α to 0. The direction of plasticity at any given synapse will depend on the input strength, i.e. the value of x_i . According to Eqs. 6 and 12, if x_i is between $\frac{\theta_D}{\alpha}$ and $\frac{\theta_P}{\alpha}$, w_i potentiates, if x_i is larger than $\frac{\theta_P}{\alpha}$, w_i depresses, otherwise there is no change (Fig. 4B1).

If we choose nonzero values for both α and γ , the direction of plasticity at each synapse will depend on sum of the strength of its presynaptic input x_i and well as the output strength \hat{y} . By choosing different values for α and γ , it is possible to emphasize the effect of the presynaptic input versus the postsynaptic output on the $[Ca^{2+}]$, and consequently the plasticity (Figure 4B2-4B4).

It is similarly possible to have postsynaptic-only frequency-dependent plasticity, which depends only on the output strength, \hat{y} , by setting all coefficients other than γ to 0. Now, the plasticity at all synapses will depend only on the output firing rate, \hat{y} . If \hat{y} is between $\frac{\theta_D}{\gamma}$ and $\frac{\theta_P}{\gamma}$, all synapses potentiate, if \hat{y} is larger than $\frac{\theta_P}{\gamma}$, all synapses depress, otherwise no change occurs at any synapse (Fig. 4B5).

Because we are using a step function for the learning rate $\eta(C)$ (Eq. (11)), the frequency of the input and output affect only the direction of the synaptic change, not its magnitude. However, if desired, it is possible to implement more biologically realistic frequency-dependent rules where the magnitude of plasticity is more precisely titrated by the $[Ca^{2+}]$ by defining $\eta(C)$ as a soft threshold function (e.g., a sum of sigmoid functions) instead of a step function (see (Moldwin et al., 2023)).

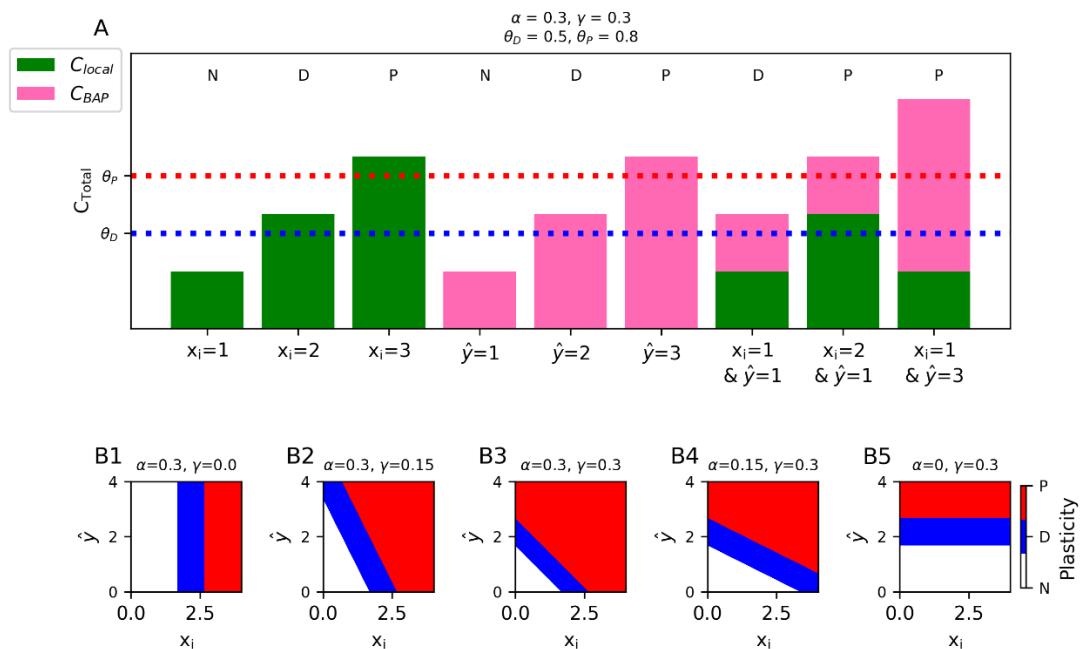


Figure 4: Frequency-dependent pre- and post-synaptic plasticity in rate-based models

(A) Calcium-dependent plasticity in a rate model. In the absence of postsynaptic spikes, sufficiently strong presynaptic inputs (x_i) alone can generate plasticity at active synapses (green bars). Postsynaptic firing alone (\hat{y}) can induce plasticity even at inactive synapses (pink bars), and the combination of presynaptic input and postsynaptic spiking can sum to induce plasticity at active synapses (stacked green and pink bars). **(B1-B5)** $[Ca^{2+}]$ (binned into regions of no change (white), depressive (blue) or potentiative (red)) as a function of presynaptic (x_i -axis) and postsynaptic (y -axis) firing rate. Each panel has a different value for α ($[Ca^{2+}]$ per presynaptic spike) and γ ($[Ca^{2+}]$ per postsynaptic spike).

Unsupervised learning of repetitive patterns with heterosynaptic plasticity

One task we might want a neuron to perform is to learn recognize a particular input pattern, i.e., by emitting a spike in response to it while not firing in response to other input patterns. This task is usually implemented as a supervised learning task (such as in the perceptron algorithm, see below), but it is also possible for a neuron to learn to recognize specific patterns in an unsupervised fashion using heterosynaptic plasticity. Instead of directly telling the neuron which inputs should elicit a spike, it is possible to teach a neuron using

heterosynaptic plasticity to spike only in response to frequently repeated “signal” patterns, while ignoring sporadic random “noise” patterns (Fig. 5A).

We assume here that both the signal and noise input patterns have the same sparsity, i.e. that there are always k out of N active synapses at every time step. We enforce that input patterns always heterosynaptically depress non-active synapses by setting β in Eq. 6 such that $\theta_D < \beta k w_{min} < \beta k w_{max} < \theta_P$ and homosynaptically potentiate active synapses by enforcing that $\beta k w_{min} + \alpha > \theta_P$ (Fig. 5B).

Every time an input pattern is presented to the neuron, the synapses that are active in that pattern will be potentiated and the synapses that are inactive will be depressed. If $\eta(C) < 1$, the potentiation and depression will occur gradually as the patterns are presented, and each synapse thus retains a “memory” of the recent input history. This creates a sort of competition between input patterns, because each pattern potentiates its active synapses while depressing synapses that were active in other patterns. Patterns that are repeated frequently (the signal patterns) will tend to dominate this competition, as the other (noise) non-repeated patterns will cancel out each other’s plastic influence on the synaptic weights. Over time, the calcitonin’s synapses will tend to strengthen the synapses associated with the frequently repeated signal pattern and depress other synapses, eventually inducing the calcitonin to spike only in response to the signal pattern (Fig. 5C1-C5).

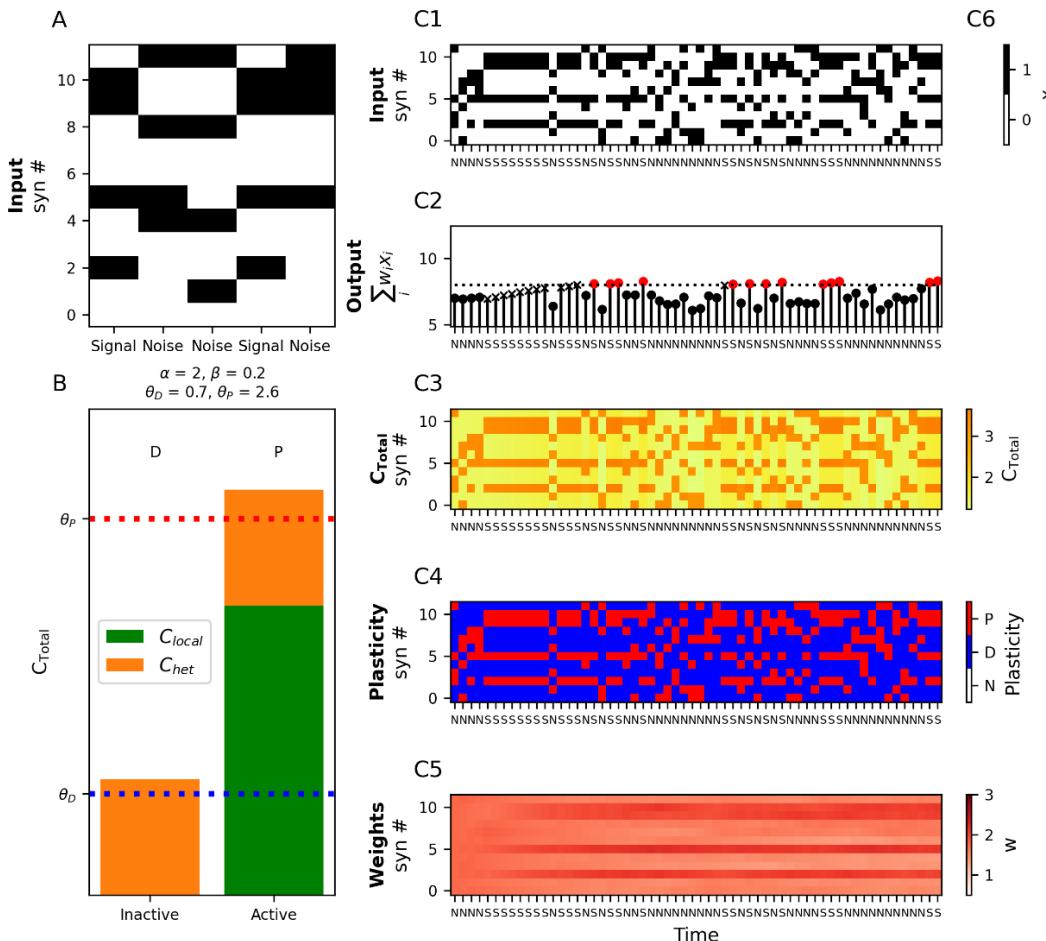


Figure 5: Learning to recognize repetitive patterns with heterosynaptic plasticity

(A) A “signal” pattern is presented repeatedly to the neuron interspersed with non-repeating random “noise” patterns of the same sparsity. **(B)** Within each input pattern (both signal or noise) inactive synapses depress (above θ_D at left) due to the heterosynaptic calcium, whereas active synapses will potentiate from the sum of heterosynaptic calcium, C_{het} , and local calcium C_{local} (above θ_P at right) **(1)** Signal and noise patterns are presented to the neuron. (S: signal, N: noise). **(C2)** Spiking output of the calcitonin. Black: no spike, Red: spike. An ‘x’ marker indicates incorrect output (e.g. no spike in response to a signal pattern), filled circles indicate correct outputs. Note the increase in correct spiking output for the signal inputs depicted by the red circles. **(C3-C5)** Calcium, plasticity, and weights over time respectively as the input patterns in C1 are presented. Note that the synaptic weights (C5) change so that they eventually resemble the signal patterns.

Very loosely speaking, the above plasticity rule can be understood as a process to set the neuron's weights to a running average over its recent inputs. This is why the neuron is able to detect the signal patterns from the noise patterns – because the noise patterns average each other out, leaving only the active synapses from the signal pattern. Because of this averaging effect, if there were multiple repeating input patterns, we would expect the neuron's weights to be strongest at synapses that participate in more patterns and weaker at synapses that only participate in a few patterns. We note that because this learning rule only depends on the weighted inputs and not the activation function g , the learning rule is similar in structure and function to Oja's rule for neural learning of principle components (Oja, 1982). For additional work on plasticity models for unsupervised learning of input patterns, see (Clopath et al., 2010; Yeung et al., 2004).

“One-shot flip-flop” (1SFF) model of behavioral time-scale plasticity (BTSP)

Recent experimental findings in the hippocampus have revealed a novel form of plasticity, known as behavioral time scale plasticity (BTSP) (Bittner et al., 2015, 2017b; Grienberger & Magee, 2022b; Milstein et al., 2021). A mouse running on a treadmill can spontaneously form a place field in a CA1 hippocampal neuron when the soma of the neuron is injected with a strong current, inducing a plateau potential (this also occurs spontaneously *in vivo* via a supervising signal from the entorhinal cortex, see (Bittner et al., 2017; Grienberger & Magee, 2022). After a single induction, this plateau potential results in the neuron exhibiting a place field selective to the mouse's location few seconds before or after the time of the plateau potential. Moreover, this place field can be modified; if a second plateau potential is induced while the mouse is at a different location near the first place field, the place field will shift to the new location, thus “overwriting” the first place field (Milstein et al., 2021). In previous work, we showed how the calcium control hypothesis might be able to explain various aspects of these experimental results (Moldwin et al., 2023).

BTSP can be reduced to a more abstract, idealized form of learning. If we ignore the precise temporal dynamics, BTSP can be thought of as a form of supervised one-shot learning, wherein a supervisory signal potentiates all synapses that were coactive with it, while depressing all other synapses. If we consider a bistable synaptic weight that can either be in an “potentiated” (1, i.e., w_{max}) or “depressed” (0, i.e., w_{min}) position, this supervisory plateau signal in BTSP is effectively a “write” command which tells the neuron to store the state of its inputs, i.e. active (1) or inactive (0) as its weights. We call this “one-shot flip-flop” learning, because the supervising signal overwrites all the weight to new binary values in a single timestep.

To implement this in the calcitonin, we again assume binary inputs, and we also enforce binary synaptic weights by setting $\eta(C) = 1$ in both the depressive and potentiative regions of calcium, so a synapse will immediately be set to w_{min} whenever the $[Ca^{2+}]$ in the depressive region and to w_{max} whenever there is a potentiative $[Ca^{2+}]$ value (Fig. 6A1-A3).

Assuming that the supervisory signal $Z = 1$ when active, one-shot flip-flop learning can implemented in the calcitonin by enforcing $\alpha < \theta_D$ (presynaptic input alone doesn't cause plasticity), $\theta_D < \delta < \theta_P$ (the supervisory signal depresses all inactive synapses) and $\delta + \alpha > \theta_P$ (the supervisory signal potentiates all active synapses) (Fig. 6B).

To illustrate the effect of one-shot flip-flop learning, we present the calcitonin with a repeated sequence of 4 binary input patterns, labeled with the numbers 0-3, to simulate a mouse running on a circular track (Figure 6C1). These 4 inputs patterns can be thought of as locations on a track traversed by the mouse. We then provide a supervisory provide a supervisory signal at random intervals (Figure 6C2). Every time the supervisory signal is presented, the binary state of the weights at the following time step are set to the binary state of the inputs that were active in tandem with the supervisory signal (Fig. 6C3-C5). The next time the neuron encounters the input pattern (location) at which it previously received a supervisory signal, the neuron fires. In other words, the supervisory signal turned the neuron to a place field for that location. When another supervisory signal comes at a different location, the previous place field is overwritten with a new place field at the new location (Fig. 6C6).

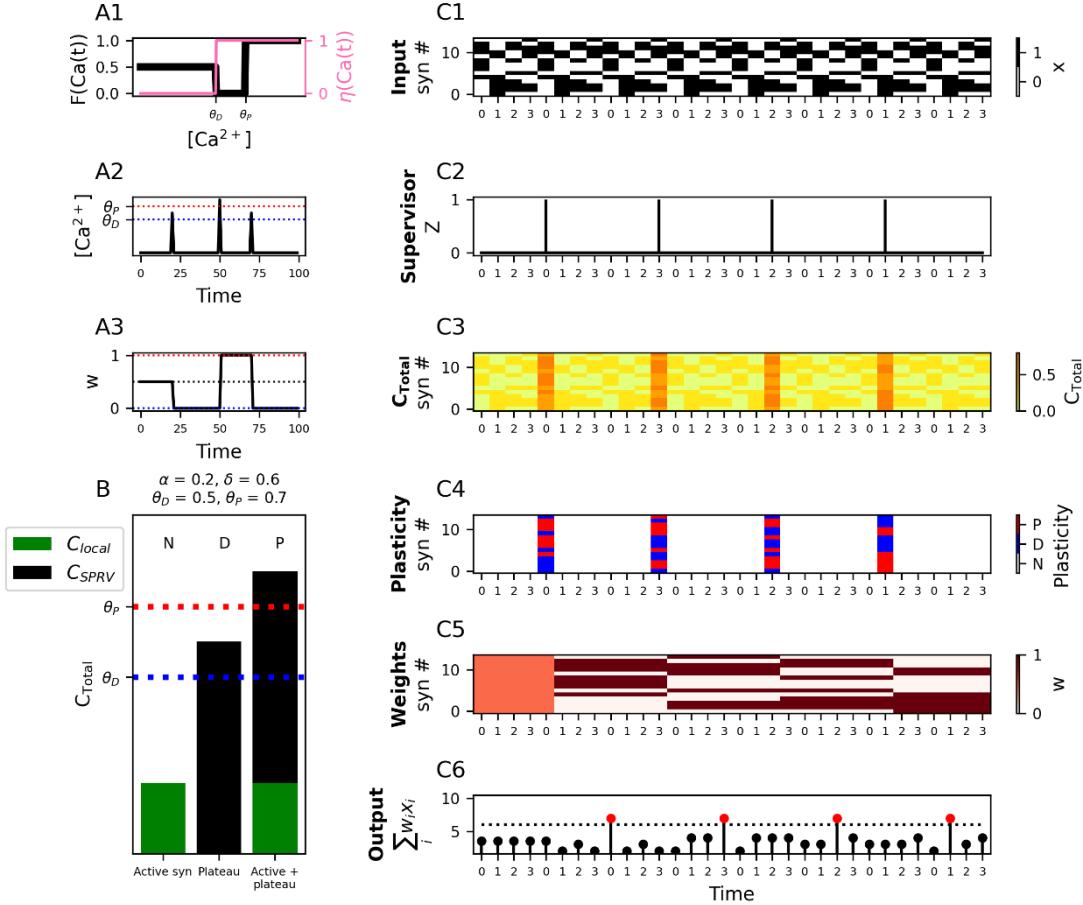


Figure 6: “One-shot flip-flop” (1SFF) plasticity

(A1) Fixed points ($F(Ca(t))$, black line, left y-axis) and learning rates ($\eta(Ca(t))$, pink line, right y-axis) for the different regions of $[Ca^{2+}]$. For 1SFF learning, the learning rate is set to 1 in the depressive and potentiative regions of $[Ca^{2+}]$ for immediate switch-like plasticity. **(A2)** Exemplar stimulus illustrating plasticity dynamics. An instantaneous $[Ca^{2+}]$ pulse is generated at three timesteps over the course of the experiment. **(A3)** Synaptic weights over time in response to stimulus presented in A2. **(B)** 1SFF plasticity rule. Local input alone does not reach the depression threshold, a plateau potential alone induces a depressive $[Ca^{2+}]$, but local input combined with a plateau potential induces potentiation. **(C1)** A repeated sequence of input patterns (0,1,2,3) corresponding to locations on a circular track that a mouse traverses at each timestep as it runs multiple laps. **(C2)** Externally generated supervisory signal (plateau potential), Z , presented at different locations over the course of the experiment. **(C3-C5)** $[Ca^{2+}]$, plasticity, and weights, respectively for each time step. Synaptic inputs at the time of the supervisory signal are “written” to the synaptic weights at the following respective time step. **(C6)** Neural output at each time step. Red circles indicate spikes. Note that the neuron spikes at the location at which the supervisory signal occurred in the previous lap.

Homeostatic Plasticity with both Internal and Circuit Mechanisms

Another important form of plasticity does not involve storing new information per se, but rather maintaining a regular average firing rate. This form of plasticity is known as homeostatic plasticity. Homeostatic plasticity can be important for the health of the neuron (i.e. too much firing can deplete neuronal resources, potentially leading to cell death) as well as maintaining stability and regularity within neural circuits (G. Turrigiano, 2012; G. G. Turrigiano, 2008; G. G. Turrigiano et al., 1998; G. G. Turrigiano & Nelson, 2004).

Before we demonstrate how homeostatic plasticity can be implemented in the calcitonin, we first note that our solution for the mechanism of homeostatic plasticity differs from what has been observed experimentally, which involves a different calcium signaling pathway and takes place over much longer time scales (many hours, instead of seconds or minutes) (G. Turrigiano, 2012). One form of experimentally-observed homeostatic plasticity seems to depend on *somatic* calcium concentrations – when a neuron is firing too slowly, the low calcium levels set off a signaling pathway to globally increase synaptic strengths, and when a neuron fires too much, the high calcium levels initiate a signaling pathway to decrease synaptic strengths. This form of homeostatic plasticity thus involves a different relationship between calcium and the direction of plasticity – low levels of somatic calcium induce

potentiation, medium levels induce no change, and high levels induce depression (G. Turrigiano, 2012). For the purposes of our work, however, we propose an alternative strategy that depends on the calcium at the spine with the standard plasticity thresholds: low levels of calcium induce no change, medium levels induce depression, and high levels induce potentiation. As such, the calcitonin version of homeostatic plasticity is a speculative exploration of how the brain could potentially implement homeostatic plasticity using the standard synaptic plasticity mechanisms and calcium thresholds.

In a rate model version of the calcitonin, we can think of homeostatic plasticity as the problem of trying to keep the output firing rate, \hat{y} , within a target range, between \hat{y}_{min}^* and \hat{y}_{max}^* . We will also assume that due to the postsynaptic refractory period, neurons have a maximum rate that it is physically possible to fire at, \hat{y}_{max} , and that neurons also have a minimum physically possible firing rate \hat{y}_{min} (trivially, neural firing rates can't be negative, so we will always have $\hat{y}_{min} \geq 0$). In general, then, we have that $\hat{y}_{min} \leq \hat{y}_{min}^* \leq \hat{y}_{max}^* \leq \hat{y}_{max}$.

For simplicity, if we assume that the synaptic inputs to the calcitonin are binary (even though the output is not), there are broadly two strategies we can take with a homeostatic plasticity rule. One strategy, which we term "global homeostasis", is that whenever the calcitonin's output \hat{y} is too large (or too small), we can depress (potentiate) all the calcitonin's synapses irrespective of their input. This will eventually result in the calcitonin's output being within the target range on average.

Although global homeostasis may be effective if the calcitonin's output is consistently too low or too high irrespective of the input patterns, globally depressing or potentiating all synapses is a drastic measure that can destroy previously stored information (unless scaling is multiplicative, see (G. G. Turrigiano, 2008; G. G. Turrigiano & Nelson, 2004)). Instead, we can use a more fine-grained approach, "targeted homeostasis", which only modifies the synapses that were active at the time when the output firing rate was out of range, thus only correcting "errant" synapses. (See also (Rabinowitch & Segev, 2006b, 2006a) regarding the consequences of the global vs. local strategies of homeostatic plasticity when considering a realistic dendritic tree.)

For the global homeostasis strategy we will always set $\alpha = 0$ so we can ignore presynaptic calcium, and for the targeted homeostasis strategy we will always set $0 < \alpha < \theta_D$ so the presynaptic calcium can break the symmetry between active and inactive synapses but isn't so large that it is sufficient on its own to induce plasticity.

We first consider how both homeostatic plasticity strategies might be implemented using internal mechanisms (i.e. without an external supervisory signal). If our neuron is firing too strongly, i.e. $\hat{y} > \hat{y}_{max}^*$, we would like to see the resultant calcium to be in the depressive $[Ca^{2+}]$ region. For the global homeostasis strategy, this can be implemented by setting $\gamma = \frac{\theta_D}{\hat{y}_{max}^*}$ and $\theta_P > \gamma \hat{y}_{max}$, and in the targeted homeostasis strategy by setting $\gamma = \frac{\theta_D}{\hat{y}_{max}^* + \alpha}$ and $\theta_P > \gamma \hat{y}_{max} + \alpha$. Because $\theta_P > \gamma \hat{y}_{max}$ in both strategies, there is no longer any way to potentiate synapses, because even the maximum physically possible firing rate will not produce enough $[Ca^{2+}]$ to induce potentiation. We must instead rely on an external supervisory mechanism to potentiate synapses when the calcitonin's output is too low.

To construct a potentiation supervisor, we consider a simple disinhibitory circuit. The calcitonin forms a synapse onto an inhibitory neuron, which inhibits a supervisory neuron that supervises the calcitonin. When the calcitonin's output rises above \hat{y}_{min}^* , the inhibitory neuron is active, thus preventing the supervisory neuron from sending the calcitonin an inhibitory signal. When the calcitonin's output falls below \hat{y}_{min}^* , however, the inhibitory neuron becomes inactive, thus permitting the supervisory neuron to send a potentiative supervisory signal $Z_P = 1$ to the calcitonin (Fig. 7A-7B). For the global homeostasis strategy, by setting $\delta = \theta_P + \epsilon$, the supervisor induces potentiation at all synapses whenever $\hat{y} < \hat{y}_{min}^*$. For the targeted homeostasis strategy, we set $\delta = \theta_P - \alpha + \epsilon$ to ensure that only active synapses are potentiated, and we apply an additional constraint that $\delta < \theta_D$, to ensure that non-active synapses aren't depressed by the supervisory signal (Fig. 7C). (This constraint imposes that $\alpha < \theta_P - \theta_D$, which in turn entails that the pre-depressive region of $[Ca^{2+}]$ is larger than the depressive region of $[Ca^{2+}]$.)

It is also possible to implement both homeostatic potentiation and depression using external supervisors, instead of using an internal supervisor for depression. To do this, we set $\gamma=0$, so the postsynaptic spike itself doesn't induce calcium influx, and we use the same disinhibitory circuit mechanism for depression as described above. To implement an external depression supervisor, we consider an addition circuit mechanism where the calcitonin also

synapses directly onto a new “depression supervisor” neuron, which synapses back onto the calcitonin with a supervising synapse. This depression supervisor will be active whenever the calcitonin’s firing rate \hat{y} exceeds \hat{y}_{max} (Fig. 7D).

Importantly, the depression supervisor and potentiation supervisor give supervisory signals of different strengths, Z_D and Z_P , respectively. Without loss of generality, we can set $\gamma = 1$, so it is only necessary to set magnitude of the supervising signals. For the global homeostasis strategy, we set $Z_D = \theta_D$ and $Z_P = \theta_P + \epsilon$, which provides the calcitonin with a global potentiative signal when $\hat{y} < \hat{y}_{min}^*$ and a global depressive signal when $\hat{y} > \hat{y}_{max}^*$ (Fig. 6E). For targeted homeostasis, we set $Z_D = \theta_D + \alpha$ and $Z_P = \theta_P + \alpha + \epsilon$, and we enforce that $Z_D < Z_P < \theta_D$ so that the supervising signals only modify active synapses (Fig. 6F).

To compare the two different supervisory circuits (external potentiation and internal depression vs. external potentiation and depression) and the targeted vs. global strategies, we created a calcitonin whose even-numbered synapses were initialized to small weights and whose odd-numbered synapses were initialized to large weights. We then randomly present input patterns that either only activate the even-numbered synapses (“even patterns”) or only activate the odd-numbered synapses (“odd patterns”). Initially, the even patterns produce an output which is too low (i.e., below $\hat{y}_{min}^* = 1$) and the odd patterns produce an output which is too high (i.e., above $\hat{y}_{max}^* = 3$). Over the course of presenting the patterns multiple times, the homeostatic mechanisms succeeded to increase the weights for the even synapses and decrease the weights for the odd synapses such that eventually the neuron fired within the target output range ($\hat{y}_{min}^* < \hat{y} < \hat{y}_{max}^*$) in response to both even and odd patterns. All four combinations of supervisory circuit structure and plasticity strategy (global vs. targeted) succeeded in this task, demonstrating that the same desired computational result can emerge from different underlying mechanisms (Fig. 6).

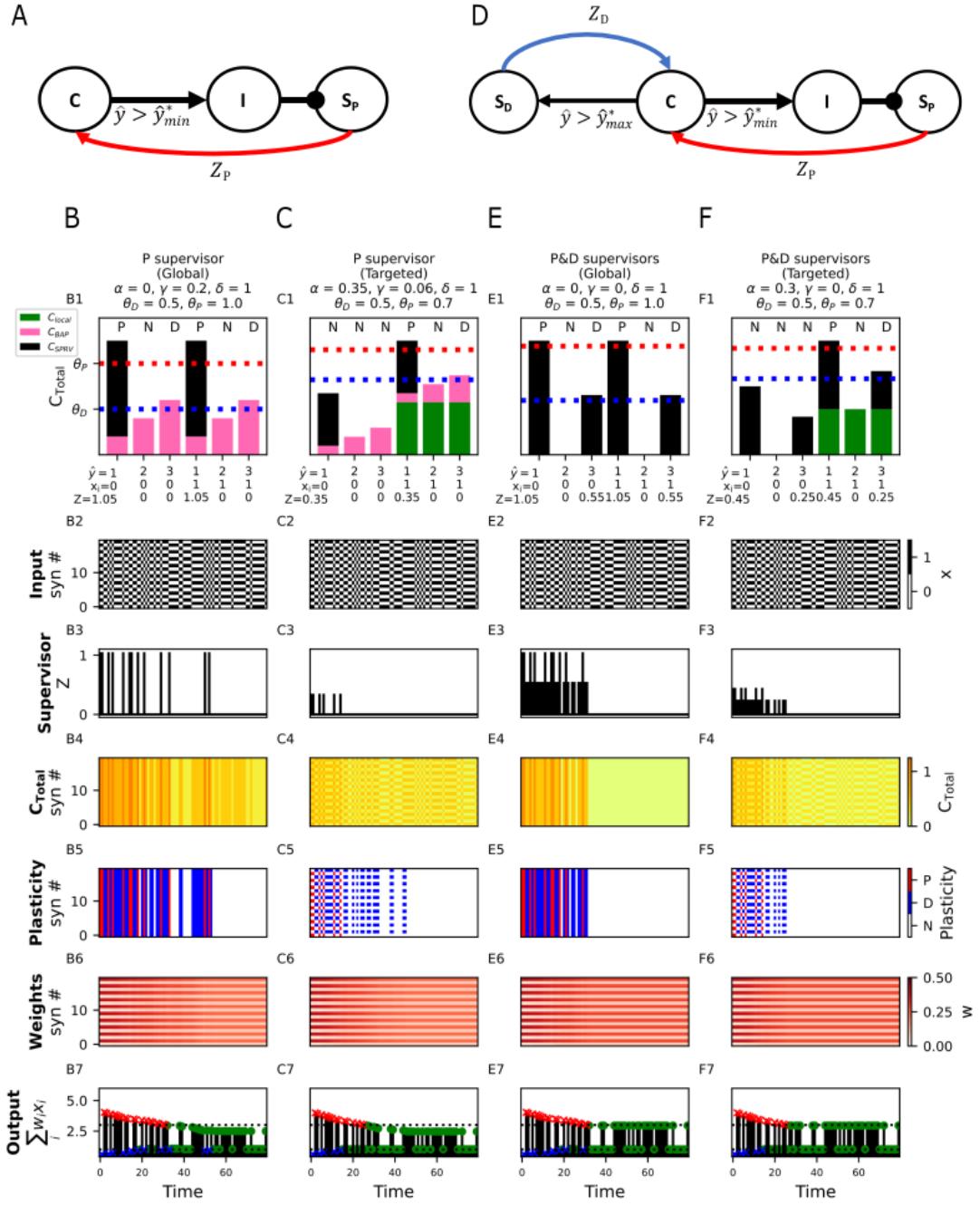


Figure 7: Different mechanisms for Homeostatic Plasticity

(A) Supervision circuit for homeostatic plasticity using only a potentiation supervisor. If the calcitonin (“C”) fires above the target minimum rate (i.e., $\hat{y} > \hat{y}_{min}^*$) it activates an inhibitory population (“I”), which prevents the potentiation supervisor (S_p) from producing a supervisory signal (in this case, $Z_p = 0$). If the calcitonin’s output falls below its target range (i.e., $\hat{y} < \hat{y}_{min}^*$) the supervisor is disinhibited, sending a potentiative calcium signal (Z_p) to the calcitonin. **(B)** Plasticity rule for global homeostatic plasticity using an internal mechanism for depression and a circuit mechanism for potentiation, as in (A). Here $\hat{y}_{min}^* = 1$ and $\hat{y}_{max}^* = 3$. **(B1)** Overly strong outputs ($\hat{y} > 3$) produce sufficient calcium to depress all synapses; Overly weak outputs ($\hat{y} < 1$) result in the activation of S_p , setting $Z_p = 1$, potentiating all synapses. **(B2)** Two input patterns (only even synapses or only odd synapses) are presented to the neuron in random order. **(B3)** Z_p occurs whenever $\hat{y} < 1$. **(B4-B6)** $[\text{Ca}^{2+}]$, plasticity, and weights over the course of the simulation. Even-numbered synapses are initialized to low weights; odd-numbered synapses are initialized to large weights. **(B7)** Neural rate output. Blue ‘x’ indicates output below \hat{y}_{min}^* (lower dashed line), red ‘x’ indicates output above \hat{y}_{max}^* (upper dashed line), green circles indicate output in the acceptable range. **(C)** Plasticity rule for targeted homeostatic plasticity using the circuit from (A) as well as local $[\text{Ca}^{2+}]$. **(D)** Supervision circuit for homeostatic plasticity using both a potentiation (S_p) and depression (S_D) supervisor. In addition to the disinhibitory circuit for the control of S_p as in (A), when the calcitonin’s output is above the target output range ($\hat{y} > \hat{y}_{max}^*$), a depression supervisor (S_D) is activated, sending a depressive signal (Z_D) to the calcitonin. **(E)** Plasticity rule for global homeostatic plasticity using both S_p and S_D . **(F)** Plasticity rule for targeted homeostatic plasticity using S_p and S_D in combination with local calcium at active synapses.

Perceptron learning algorithm with calcium-based plasticity

We now show that it is possible to implement the perceptron learning algorithm with calcium-based plasticity. The perceptron learning algorithm (Rosenblatt, 1958) is the procedure by which a single linear neuron (see Eq. (1)) can learn to solve classification tasks, such as distinguishing between images of cats and dogs, by modifying its synaptic weights.

The perceptron learning rule is a supervised learning rule. Namely, each input pattern comes with an associated target outcome – that the neuron should either spike or not spike. Formally, we have a set of P input patterns $\{x^1 \dots x^\mu \dots x^P\}$ and associated labels $\{y^1 \dots y^\mu \dots y^P\}$, where the bolded x^μ is an N -dimensional vector of activity of the μ th input pattern and $y^\mu \in \{0,1\}$ is the associated target label, or class. The goal of the perceptron learning algorithm is to ensure that the neuron’s output on each pattern (\hat{y}^μ) matches the target label, i.e., $\hat{y}^\mu = y^\mu$.

The perceptron rule states that if the neuron makes an error on an input pattern $\hat{y}^\mu \neq y^\mu$, we increase or decrease the perceptron’s synaptic weights in a manner proportional to the input vector. Formally, if $\hat{y}^\mu = 0$ and $y^\mu = 1$ (a false negative; the neuron should have spiked but didn’t) we update each weight w_i according to the rule $w_i \leftarrow w_i + \eta x_i$, where η is the learning rate. If $\hat{y}^\mu = 1$ and $y^\mu = 0$ (a false positive; the neuron spiked when it wasn’t supposed to) we update each weight w_i according to the rule $w_i \leftarrow w_i - \eta x_i$. If the neuron produced the correct output, $\hat{y}^\mu = y^\mu$, we don’t modify the weights. The change in each synaptic weight at each time step, $\Delta w_i(t) = w_i(t+1) - w_i(t)$, for the perceptron learning rule can be described by Table 1.

	$y = \mathbf{0}$	$y = \mathbf{1}$
$\hat{y} = \mathbf{0}$	0	ηx
$\hat{y} = \mathbf{1}$	$-\eta x$	0

Table 1: Weight update Δw in the standard perceptron learning rule.

We first note that because the fixed-point learning rate (FPLR) framework for calcium-based plasticity has asymptotic, weight-dependent dynamics, it is only possible to exactly replicate the perceptron learning rule with the linear calcium-based rule (Eq. (9)). However, because we would like to demonstrate perceptron-like dynamics with the more realistic FPLR rule (Eq. (11)), we propose an “asymptotic perceptron learning rule” which functions similarly to the original perceptron rule, except that weights increase or decrease asymptotically towards w_{min} or w_{max} instead of being able to increase or decrease indefinitely, as in the standard perceptron rule.

Formally, if $\hat{y}^\mu = 0$ and $y^\mu = 1$ (false negative), we update w_i according to the rule $w_i \leftarrow w_i + \eta(w_{max} - w_i)x_i$. If $\hat{y}^\mu = 1$ and $y^\mu = 0$ (false positive) we update each weight w_i according to the rule $w_i \leftarrow w_i + \eta(w_{min} - w_i)x_i$ (Table 2). Because it is always the case that $w_{min} < w_i < w_{max}$, the sign of the weight update in the asymptotic perceptron rule for each case is consistent with the sign of the update in the original perceptron rule; the rules only differ in whether the weights change linearly or asymptotically as a function of the current weight.

	$y = \mathbf{0}$	$y = \mathbf{1}$
$\hat{y} = \mathbf{0}$	0	$\eta_p(w_{max} - w)x$
$\hat{y} = \mathbf{1}$	$\eta_D(w_{min} - w)x$	0

Table 2: Weight update Δw in the asymptotic perceptron learning rule.

To implement the perceptron learning rule in the calciton, we again stipulate that the synaptic inputs are binary, so we only have to worry about the direction of synaptic change, not the magnitude. Because the perceptron is a supervised learning algorithm, we will also have a supervisory signal. In our first attempt at implementing the perceptron, the supervisory signal (the “label supervisor”) will simply indicate the value of label (Fig. 8A). In other words, for each input pattern x^μ , we have $Z^\mu = y^\mu$. The challenge here is to set the calcium thresholds and coefficients such that each quadrant of Table 2 (or Table 1 with the standard perceptron rule) is satisfied.

To satisfy the upper left quadrant ($\hat{y}^\mu = 0$ and $y^\mu = 0$), we stipulate that $\alpha < \theta_D$, so that in the absence of a postsynaptic spike and a supervising signal, the presynaptic $[Ca^{2+}]$

alone is too low to induce plasticity. For the lower left quadrant, ($\hat{y}^\mu = 1$ and $y^\mu = 0$, false positives) we require that a postsynaptic spike induces depression at an active synapse, but does nothing to inactive synapses. To accomplish this, we enforce $\gamma < \theta_D$ and $\theta_D < \gamma + \alpha < \theta_P$. For the upper right quadrant, ($\hat{y}^\mu = 0$ and $y^\mu = 1$, false negatives), we want the supervisor in the absence of a postsynaptic spike to induce potentiation at active synapses but do nothing at inactive synapses. For this we require that $\delta < \theta_D$ and $\delta + \alpha > \theta_P$.

A problem arises, however, when we get to the lower right quadrant ($\hat{y}^\mu = 1$ and $y^\mu = 1$, true positives). When there is both a postsynaptic spike and a supervising signal, we require that none of the synapses will be updated. But we already enforced that $\delta + \alpha > \theta_P$, so now when we have both a postsynaptic spike and a supervisory signal we have a $[Ca^{2+}]$ of $\delta + \alpha + \gamma$ at active synapses, which is certainly greater than θ_P ! We solve it by adding a third region of $[Ca^{2+}]$, the post-potentiative neutral zone, or PPNZ ($\theta_{PPNZ} > \theta_P$), wherein the $[Ca^{2+}]$ is so high that it ceases to potentiate synapses (there is some evidence for this, see (Tigaret et al., 2016)). We then enforce that $\delta + \gamma > \theta_{PPNZ}$, which gives us our fourth quadrant and allows us to reproduce the perceptron learning rule in its entirety (Fig. 8B).

If we don't wish to add a post-potentiative neutral zone, we can still implement the perceptron learning rule with the normal calcium thresholds if we use a "smart" supervisory signal. Instead of just telling the calciton what the correct label is, we can have a "critic supervisor" which compares the calciton's output \hat{y}^μ to the target output y^μ , and gives supervisory signals of different strengths depending on whether the input pattern resulted in a false positive or false negative. Importantly, the supervisor will no longer give any supervisory signal when the pattern is classified correctly, circumventing the problem we had with true positives using the label supervisor. Here, we will not explicitly describe the circuit necessary to construct such a supervisor, we will simply assume that some other brain circuit can perform the operation of comparing the calciton's output to the target output and produce the appropriate supervisory signals (Fig. 8C).

Once we have a critic supervisor, it is no longer necessary to use the calcium from the postsynaptic spike, so we set $\gamma = 0$. Now the strategy for implementing the perceptron is straightforward. For true positives ($\hat{y}^\mu = 1$ and $y^\mu = 1$) and true negatives ($\hat{y}^\mu = 0$ and $y^\mu = 0$), there is no supervisory calcium, so we just have to ensure that an active synapse by itself doesn't induce plasticity by setting $\alpha < \theta_D$. For false negatives ($\hat{y}^\mu = 0$ and $y^\mu = 1$) we need the presynaptic calcium combined with the active synapse to induce potentiation, while the supervisor alone doesn't induce any plasticity at inactive synapses. To do this we set $\delta = 1$ and we construct a potentiation supervisor for false negatives, Z_P , such that $Z_P < \theta_D$ and $Z_P + \alpha > \theta_P$. For false positives ($\hat{y}^\mu = 1$ and $y^\mu = 0$), we similarly construct a supervisor Z_D , such that $Z_D < \theta_D$ and $\theta_D < Z_D + \alpha < \theta_P$. These constraints satisfy the rules specified in Table 2 (and Table 1) in a much simpler manner.

To illustrate that the Calciton can indeed implement the perceptron learning rule just by setting the calcium coefficients and thresholds, we performed a simple classification experiment using a calciton with the constraints and critic supervisor described above. We generated six binary patterns with 24 synapses each. Half of the patterns were arbitrarily assigned to the positive class and half to the negative class ($y = 1$ or $y = 0$, respectively). We repetitively presented these patterns to both to the "critic" and "target" calciton supervisory circuits. Initially, the calciton makes mistakes on some of the patterns, of both the "false positive" and "false negative" variety. After a sufficient number of presentations of each input pattern, however, both supervisory circuits succeed to ensure that the calciton correctly classifies all patterns (Fig. 8).

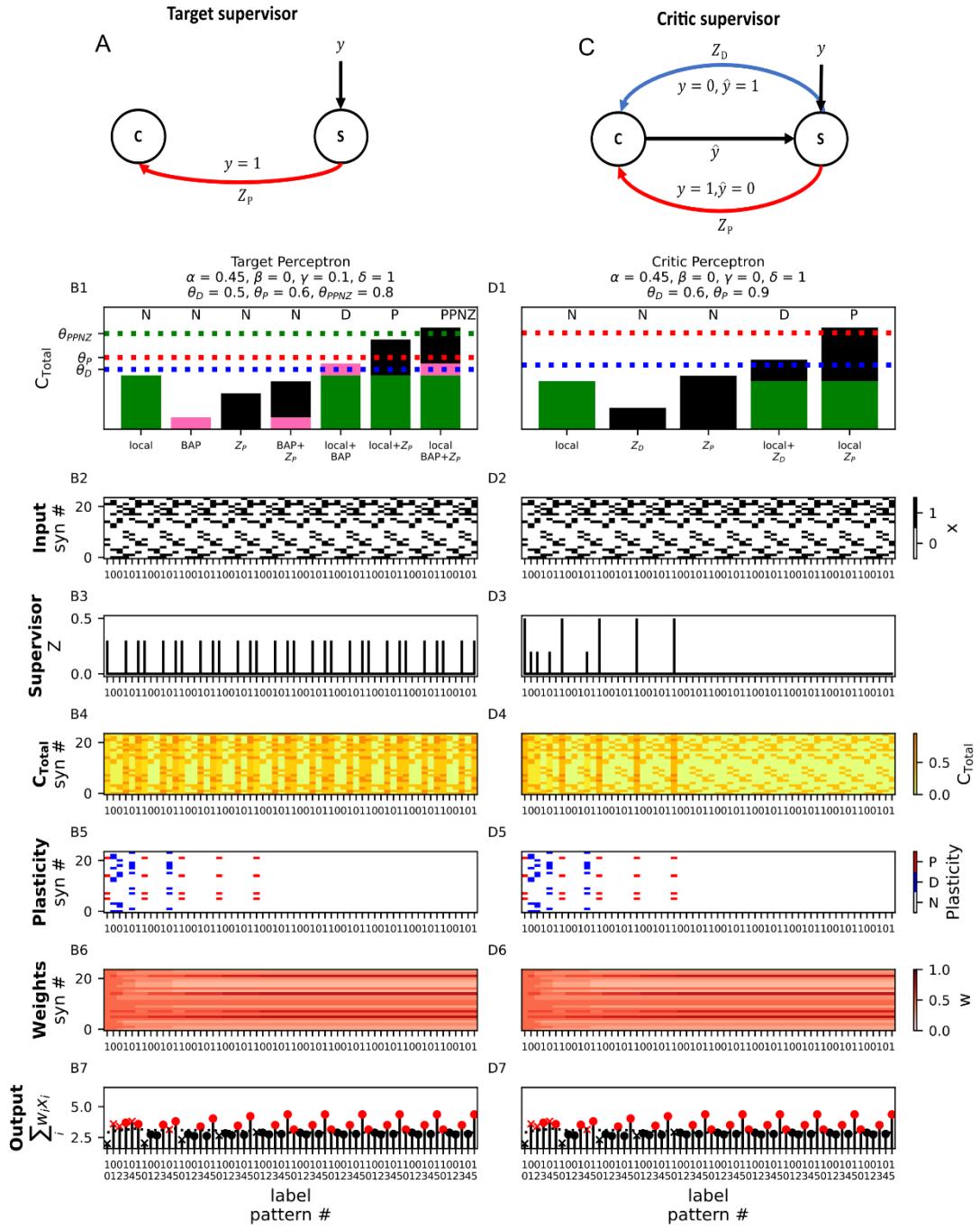


Figure 8: Perceptron learning with the calciton

(A) Supervision circuit for perceptron learning using a “target” supervisor. Whenever the target label is 1, the supervisor sends a potentiative supervisory signal Z_P to the calciton. (B1) Plasticity rule for the perceptron with a target supervisor. Note that an additional calcium threshold (dashed green line) for a post-potentiative neutral zone (PPNZ) where no plastic change occurs has been added to the plasticity rule. (B2) Six patterns, half of which are arbitrarily assigned to the positive class and half to the negative class ($y = 1$ or $y = 0$, respectively, see tick labels on x-axis) are repetitively presented to the calciton. (B3) Supervisory signal. Appears whenever the target label $y = 1$. (B4-B6) $[Ca^{2+}]$, plasticity, and weights over the course of the simulation. (B7) Calciton output. Red circle: true positive, red ‘x’: false positive, black circle: true negative, black ‘x’: false negative. (C) Supervision circuit for perceptron learning using a “critic” supervisor. The supervisor compares the target label y to the calciton output \hat{y} . If the trial was a false negative ($y = 1, \hat{y} = 0$), the supervisor sends a potentiative signal supervisory signal Z_P to the calciton. If this trial was a false positive ($y = 0, \hat{y} = 1$), the supervisor sends a depressive supervisory signal Z_D to the calciton. (D1-D7) Perceptron learning with the “critic” supervisory circuit. Note the different magnitudes of the supervisory signal in (D3) – the large signal corresponds to Z_P and the small signal corresponds to Z_D .

We have thus demonstrated that the perceptron learning algorithm can be implemented with the calciton, either with a “label supervisor” – which requires an additional plasticity

threshold, or with a “critic supervisor” which can use the standard plasticity thresholds.

Discussion

Summary

We have shown that the calcitron, a simple model neuron that uses four different sources of calcium to modify its synaptic weights, can implement a wide variety of learning and plasticity rules. We have demonstrated that merely by appropriately setting the amount of calcium influx synapses receive from each calcium source, it is possible to reproduce classic learning rules, like Hebb’s rule (Fig. 2) or the perceptron learning rule (Fig. 8). We have also reproduced, in a simplified manner, plastic phenomena observed in biology, such as frequency-dependent plasticity (Fig. 4, (O’Connor et al., 2005b)), homeostatic plasticity (Fig. 7, (G. Turrigiano, 2012; G. G. Turrigiano, 2008; G. G. Turrigiano & Nelson, 2004)), and behavioral time-scale plasticity (Bittner et al., 2015, 2017; Grienberger & Magee, 2022; Milstein et al., 2021). Moreover, we have shown how the calcium control hypothesis can result in novel plasticity and learning rules, such as the 14 pre-post rules (Fig. 3) and the unsupervised learning of large repetitive patterns with heterosynaptic plasticity (Fig. 5). We note that because each calcium source in the Calcitron is related to a current source (e.g. the presynaptic input or postsynaptic spike), the Calcitron bears some similarity to the BCM plasticity rule (Bienenstock et al., 1982).

The calcium control hypothesis was originally developed as an explanation for Hebbian and anti-Hebbian plasticity (J. Lisman, 1989). And early mathematical formulations of the calcium control hypothesis were designed to reproduce frequency-dependent and spike timing-dependent plasticity (Graupner & Brunel, 2012; Shouval et al., 2002). With the calcitron, however, we have expanded these earlier results into a generalized, simple model which can explain and predict a much wider set of plasticity and learning results from first principles. The calcitron can thus be helpful in providing an intuition for the range of possible calcium-based mechanisms underlying experimentally-observed plasticity, including forms of plasticity that are as-yet undiscovered.

The mathematical formalism of the calcitron also makes it easier to understand the limitations of calcium-based plasticity. The calcitron equations impose constraints on the potential learning rules that can emerge from the calcium control hypothesis, which can help us determine whether it is necessary to posit additional mechanisms, such as the supervisory circuit we proposed for homeostatic plasticity (Fig. 7). Importantly, we do not claim that every possible learning rule we propose here is implemented in biology as we have described it. Rather, the calcitron is intended to serve as a framework for exploring different predictions of the calcium control hypothesis.

Future Directions and Additional Biological Considerations

The calcitron model was formulated to be as simple as possible in order to provide straightforward mathematical intuitions about the calcium basis of synaptic plasticity. This simplicity comes with certain drawbacks. For example, the choice of using a perceptron-like neuron model that does not temporally integrate information means that the calcitron, as formulated here, cannot implement spike timing-dependent plasticity (STDP). Prior work has shown that a leaky integrator model with calcium dynamics can indeed generate STDP (Graupner & Brunel, 2012; Shouval et al., 2002). As such, future work can consider leaky integrator versions of the calcitron that can implement STDP and other temporally-sensitive plasticity rules. However, this temporal sensitivity introduces the need to finely tune calcium decay time constants in order to produce different plasticity rules, which introduces a level of complexity beyond the scope of the current work. It is also possible to explicitly model the kinetics of the relevant ion channels (i.e. VGCCs and NMDA receptors), adding even more biological detail at the expense of greater complexity.

The calcitron is also linear, both in terms of its input-output function (excluding the activation function) and in terms of how calcium from different sources combines to make calcium. Linear point neurons are commonly used to model neural phenomena, although experimental and theoretical work indicate that real neurons may integrate information in a nonlinear fashion (Beniaquev et al., 2021; Gordon et al., 2006; Mel, 1991; Moldwin et al., 2021; Moldwin & Segev, 2020; Poirazi et al., 2003; Poirazi & Mel, 2001; Polsky et al., 2004,

2009; Schiller et al., 2000; Tran-Van-Minh et al., 2015). Of particular relevance is the superlinear activation function of the NMDA receptor, whose conductance exhibits sigmoidal sensitivity to local voltage at the synapse location (Jahr et al., 1990; Jahr & Stevens, 1990). A more realistic model for local calcium influx could thus include this voltage-dependent nonlinearity, in line with experimental work showing that NMDA spikes induce plasticity (Kumar et al., 2021).

Another important feature of biological neurons not incorporated into the calcitron is the spatial distribution of synapses on a neuron's dendrites. This is especially relevant for heterosynaptic plasticity, which depends on the location of synapses relative to each other as well as their absolute location on the dendritic tree (Chater & Goda, 2021; Chistiakova et al., 2014; Moldwin et al., 2022; Tong et al., 2021). The calcitron can be augmented to include location-dependent heterosynaptic plasticity on a single dendrite following the schemes of the clusteron (Mel, 1991) or the G-clusteron (Moldwin et al., 2021) or to explicitly include a branching dendrite to account for the branch-dependent hierarchical heterosynaptic plasticity effect we posited in previous work (Moldwin et al., 2022). The spatial structure of the dendrite can also influence homeostatic plasticity (Rabinowitch & Segev, 2006a, 2006b) as well as how inhibitory inputs affect plasticity at excitatory synapses (Bar-Ilan et al., 2012).

A crucial assumption we made in formulating the calcitron is that local calcium is always a function of presynaptic activity. If we had allowed for synapse-specific supervisory signals, it would be possible to implement arbitrary learning rules without any concern for constraints, as we could simply engineer a supervisor to potentiate or depress each synapse independently. In biology, however, it is possible that synapse-specific plasticity supervisors exist. One candidate for such a mechanism are the internal calcium stores in the endoplasmic reticulum (ER) (Rose & Konnerth, 2001). The ER can invade individual spines (Spacek & Harris, 1997), making it a candidate for targeted calcium release at selected synapses. It is usually assumed that calcium released from the ER is mediated by calcium influx via a process called calcium-induced calcium release, or CICR. If the initial calcium needed to induce CICR at the ER in individual spines come from extracellular sources, CICR is still broadly consistent with our model; the internal calcium sources may simply be amplifying the calcium which enters via external sources. If, however, there are other endogenous mechanisms that can differentially induce calcium release from the ER at different synapses, the ER may be able to play the role of a synapse-specific supervisor. An additional candidate for synapse-specific supervisors are astrocytes, which target individual dendritic spines (Haber et al., 2006) and can influence plasticity via a variety of signaling pathways (Barker & Ullian, 2010). Either of these mechanisms can enable a less constrained, more diverse palette of neural learning rules when combined with the more classical calcium sources we suggested here.

With respect to the supervisory signal, we have focused on plasticity supervisors that operate via calcium mechanisms, such as observed in Purkinje neurons (Konnerth et al., 1992) and possibly in the CA1 region of the hippocampus during BTSP induction. However, there are other forms of plasticity that are more dependent on neuromodulation. For example, associative learning is often thought to require dopamine as a supervisory signal (Lisman et al., 2011; Puig et al., 2014). Future work can explore the interplay between calcium-based and neuromodulator-dependent mechanisms of plasticity.

There are a variety of ways to build on the calcitron. In this work, we have focused on building neurons and circuits that implement one learning rule at a time. It may be desirable, however, for a neuron to simultaneously implement, for example, Hebbian plasticity and homeostatic plasticity. The calcitron equations constrain which kind of learning rules can be simultaneously implemented in this fashion. It would be worthwhile to explore which learning rule combinations are mathematically valid in the calcitron framework.

Another natural extension of the calcitron would be to incorporate multiple calcitron nodes into a network to see how calcium-based plasticity affect network activity. It would be interesting to see how networks containing neurons with different calcium coefficients and thresholds (and thus different learning rules) would operate. Alternatively, it is possible to build large networks containing subnetworks with different calcium parameters, to explore how brain regions with different plasticity rules might interact with each other. The calcitron framework thus provides a theoretical tool to explore the network-level consequences of biologically realistic plasticity rules.

Methods

All simulations were performed on a Windows PC using Python and plotted using the Matplotlib library (Hunter, 2007). Parameters for each simulation are given in the table below:

Fig	Pane I	η_D	F_D	θ_P	F_P	η_P	α	β	γ	δ	b	g	N	Z	Z_P	Z_D	η_{PPNZ}	F_{PPNZ}	η_b
Fig 1																			
	E-G	-0.1			1		0.2												
	H-L	0.1	0	1	1	0.1													
Fig 2																			
	A1-F1	0.1	0	0.8	1	0.1	0.4	0	0.45	0	-1.8	step	10						
	A2-F2	0.1	0	0.8	1	0.1	0.55	0	0.7	0	-1.8	step	10						
	A3-F3	0.1	0	0.8	1	0.1	0.4	0	0.3	0	-1.8	step	10						
	A4-F4	0.1	0	0.5	1	0.1	0.55	0	0.6	0	-1.8	step	10						
Fig 3																			
	1,1	1	0	0.8	1	1	0.2	0	0.2	0									
	1,2	1	0	0.8	1	1	0.3	0	0.3	0									
	1,3	1	0	0.8	1	1	0.1	0	0.6	0									
	1,4	1	0	0.8	1	1	0.6	0	0.1	0									
	2,1	1	0	1.3	1	1	0.6	0	0.6	0									
	2,2	1	0	0.8	1	1	0.45	0	0.45	0									
	2,3	1	0	0.8	1	1	0.3	0	0.6	0									
	2,4	1	0	0.8	1	1	0.3	0	0.9	0									
	3,1	1	0	0.8	1	1	0.6	0	0.3	0									
	3,2	1	0	0.8	1	1	0.6	0	0.6	0									
	3,3	1	0	0.8	1	1	0.6	0	0.9	0									
	3,4	1	0	0.8	1	1	0.9	0	0.3	0									
	4,1	1	0	0.8	1	1	0.9	0	0.6	0									
	4,2	1	0	0.8	1	1	0.9	0	0.9	0									
Fig 4																			
	A	0.1	0	0.8	1	0.1	0.3	0	0.3	0									
	B1	0.1	0	0.8	1	0.1	0.3	0	0	0									
	B2	0.1	0	0.8	1	0.1	0.3	0	0.15	0									
	B3	0.1	0	0.8	1	0.1	0.3	0	0.3	0									
	B4	0.1	0	0.8	1	0.1	0.15	0	0.3	0									
	B5	0.1	0	0.8	1	0.1	0	0	0.3	0									
Fig 5																			
	All	0.025	1	2.6	3	0.025	2	0.2	0	0	-8	step	12						
Fig 6																			
	All	1	0	0.7	1	1	0.2	0	0	0.6	-6	step	14	1					
Fig 7																			
	B	0.02	0	1	0.5	0.02	0	0	0.2	1	0	linea r	20		1.05				
	C	0.02	0	0.7	0.5	0.02	0.35	0	0.06	1	0	linea r	20		0.35				
	E	0.02	0	1	0.5	0.02	0	0	0	1	0	linea r	20		1.05	0.55			
	F	0.02	0	0.7	0.5	0.02	0.3	0	0	1	0	linea r	20		0.45	0.25			
Fig 8																			

	B	0.2	0	0.6	1	0.2	0.4 5	0	0.1	1	- 2. 8	step	24		0.3		0	0.25	0.1
	D	0.2	0	0.9	1	0.2	0.4 5	0	0	1	- 2. 8	step	24		0.5	0.2			0.1

Code Availability

Code for this project can be found at <https://github.com/tmoldwin/Calcitron>.

References

- Amit, D. J. (1992). *Modeling brain function: The world of attractor neural networks* (M. brain function: T. world of attractor neural Networks, Ed.). Cambridge university press.
- Artola, A., Bröcher, S., & Singer, W. (1990). Different voltage-dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex. *Nature*, 347(6288), 69–72. <https://doi.org/10.1038/347069A0>
- Barker, A. J., & Ullian, E. M. (2010). *Astrocytes and Synaptic Plasticity Introduction and Background*. 16, 40–50. <https://doi.org/10.1177/1073858409339215>
- Beniaguev, D., Segev, I., & London, M. (2021). Single cortical neurons as deep artificial neural networks. *Neuron*, 109(17), 2727-2739.e3. <https://doi.org/10.1016/J.NEURON.2021.07.002>
- Bittner, K. C., Grienberger, C., Vaidya, S. P., Milstein, A. D., Macklin, J. J., Suh, J., Tonegawa, S., & Magee, J. C. (2015). Conjunctive input processing drives feature selectivity in hippocampal CA1 neurons. *Nature Neuroscience*, 18(8), 1133–1142. <https://doi.org/10.1038/nn.4062>
- Bittner, K. C., Milstein, A. D., Grienberger, C., Romani, S., & Magee, J. C. (2017). Behavioral time scale synaptic plasticity underlies CA1 place fields. *Science*, 357(6355), 1033–1036. <https://doi.org/10.1126/science.aan3846>
- Chater, T. E., & Goda, Y. (2021). My Neighbour Hetero — deconstructing the mechanisms underlying heterosynaptic plasticity. *Current Opinion in Neurobiology*, 67, 106–114. <https://doi.org/10.1016/J.CONB.2020.10.007>
- Chistiakova, M., Bannon, N. M., Bazhenov, M., & Volgushev, M. (2014). Heterosynaptic Plasticity. *The Neuroscientist*, 20(5), 483–498. <https://doi.org/10.1177/1073858414529829>
- Citri, A., & Malenka, R. C. (2008). Synaptic plasticity: Multiple forms, functions, and mechanisms. *Neuropsychopharmacology*, 33(1), 18–41. <https://doi.org/10.1038/sj.npp.1301559>
- Coesmans, M., Weber, J. T., De Zeeuw, C. I., & Hansel, C. (2004). Bidirectional Parallel Fiber Plasticity in the Cerebellum under Climbing Fiber Control. *Neuron*, 44(4), 691–700. <https://doi.org/10.1016/j.neuron.2004.10.031>
- Gordon, U., Polksy, A., & Schiller, J. (2006). Plasticity Compartments in Basal Dendrites of Neocortical Pyramidal Neurons. *Journal of Neuroscience*, 26(49), 12717–12726. <https://doi.org/10.1523/JNEUROSCI.3502-06.2006>
- Graupner, M., & Brunel, N. (2012). Calcium-based plasticity model explains sensitivity of synaptic changes to spike pattern, rate, and dendritic location. *Proceedings of the National Academy of Sciences of the United States of America*, 109(10), 3991–3996. <https://doi.org/10.1073/pnas.1109359109>
- Grienberger, C., & Magee, J. C. (2022). Entorhinal cortex directs learning-related changes in CA1 representations. *Nature* 2022 611:7936, 611(7936), 554–562. <https://doi.org/10.1038/s41586-022-05378-6>
- Hebb, D. (1949). *The organization of behavior. A neuropsychological theory*.
- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science and Engineering*, 9(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
- Jahr, C., & Stevens, C. (1990). A quantitative description of NMDA receptor-channel kinetic behavior. *Journal of Neuroscience*, 10(6), 1830–1837. <https://doi.org/10.1523/JNEUROSCI.10-06-01830.1990>
- Jahr, C., Stevens, C. C. F., Jahr', C. E., Stevens, C. C. F., Jahr, C., & Stevens, C. C. F. (1990). Voltage dependence of NMDA-activated macroscopic conductances predicted

- by single-channel kinetics. *Journal of Neuroscience*, 10(9), 3178–3182.
<https://doi.org/10.1523/JNEUROSCI.10-09-03178.1990>
- Kohonen, T. (1982). Self-Organized Formation of Topologically Correct Feature Maps. In *Biol. Cybern* (Vol. 43).
- Kriegeskorte, N. (2015). Deep Neural Networks: A New Framework for Modeling Biological Vision and Brain Information Processing. *Annual Review of Vision Science*, 1(1), 417–446. <https://doi.org/10.1146/annurev-vision-082114-035447>
- Kumar, A., Barkai, E., & Schiller, J. (2021). Plasticity of olfactory bulb inputs mediated by dendritic NMDA-spikes in rodent piriform cortex. *ELife*, 10. <https://doi.org/10.7554/ELIFE.70383>
- Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. In *Nature* (Vol. 521, Issue 7553, pp. 436–444). Nature Publishing Group. <https://doi.org/10.1038/nature14539>
- Lisman, J. (1989). A mechanism for the Hebb and the anti-Hebb processes underlying learning and memory. *Proceedings of the National Academy of Sciences*, 86(23), 9574–9578. <https://doi.org/10.1073/pnas.86.23.9574>
- Lisman, J. E. (2001). Three Ca²⁺ levels affect plasticity differently: the LTP zone, the LTD zone and no man's land. *The Journal of Physiology*, 532(2), 285–285. <https://doi.org/10.1111/J.1469-7793.2001.0285F.X>
- Lisman, J., & Goldring, M. (1988). Evaluation of a model of long-term memory based on the properties of the Ca²⁺/calmodulin-dependent protein kinase. *Journal de Physiologie*, 83(3), 187–197. <https://europepmc.org/article/med/2856110>
- McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133. <https://doi.org/10.1007/BF02478259>
- Mel, B. W. (1991). The clusteron: Toward a simple abstraction for a complex neuron. *Nips*, 35–42.
- Memmesheimer, R. M., Rubin, R., Ölveczky, B. P., & Sompolinsky, H. (2014). Learning Precisely Timed Spikes. *Neuron*, 82(4), 925–938. <https://doi.org/10.1016/j.neuron.2014.03.026>
- Milstein, A. D., Li, Y., Bittner, K. C., Grienberger, C., Soltesz, I., Magee, J. C., & Romani, S. (2021). Bidirectional synaptic plasticity rapidly modifies hippocampal representations. *ELife*, 10. <https://doi.org/10.7554/ELIFE.73046>
- Moldwin, T., Azran, L. S., & Segev, I. (2023). A Generalized Framework for the Calcium Control Hypothesis Describes Weight-Dependent Synaptic Changes in Behavioral Time Scale Plasticity. *BioRxiv*, 2023.07.13.548837. <https://doi.org/10.1101/2023.07.13.548837>
- Moldwin, T., Kalmenson, M., & Segev, I. (2021). The gradient clusteron: A model neuron that learns to solve classification tasks via dendritic nonlinearities, structural plasticity, and gradient descent. *PLOS Computational Biology*, 17(5), e1009015. <https://doi.org/10.1371/journal.pcbi.1009015>
- Moldwin, T., Kalmenson, M., & Segev, I. (2022). Asymmetric voltage attenuation in dendrites enables hierarchical heterosynaptic plasticity. *BioRxiv*, 2022.07.07.499166. <https://doi.org/10.1101/2022.07.07.499166>
- Moldwin, T., Kalmenson, M., & Segev, I. (2023). Asymmetric Voltage Attenuation in Dendrites Can Enable Hierarchical Heterosynaptic Plasticity. *ENeuro*, 10(7), ENEURO.0014-23.2023. <https://doi.org/10.1523/ENEURO.0014-23.2023>
- Moldwin, T., & Segev, I. (2020). Perceptron Learning and Classification in a Modeled Cortical Pyramidal Cell. *Frontiers in Computational Neuroscience*, 14(April), 1–13. <https://doi.org/10.3389/fncom.2020.00033>
- Oja, E. (1982). Simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15(3), 267–273. <https://doi.org/10.1007/BF00275687>
- Piochon, C., Titley, H. K., Simmons, D. H., Grasselli, G., Elgersma, Y., & Hansel, C. (2016). Calcium threshold shift enables frequency-independent control of plasticity by an instructive signal. 46, 13221–13226.
- Poirazi, P., Brannon, T., & Mel, B. W. (2003). Pyramidal Neuron as Two-Layer Neural Network. *Neuron*, 37(6), 989–999. [https://doi.org/10.1016/S0896-6273\(03\)00149-1](https://doi.org/10.1016/S0896-6273(03)00149-1)
- Poirazi, P., & Mel, B. W. (2001). Impact of active dendrites and structural plasticity on the memory capacity of neural tissue. *Neuron*, 29(3), 779–796. [https://doi.org/10.1016/S0896-6273\(01\)00252-5](https://doi.org/10.1016/S0896-6273(01)00252-5)

- Polsky, A., Mel, B., & Schiller, J. (2009). Encoding and decoding bursts by NMDA spikes in basal dendrites of layer 5 pyramidal neurons. *Journal of Neuroscience*, 29(38), 11891–11903. <https://doi.org/10.1523/JNEUROSCI.5250-08.2009>
- Polsky, A., Mel, B. W., & Schiller, J. (2004). Computational subunits in thin dendrites of pyramidal cells. *Nature Neuroscience* 2004 7:6, 7(6), 621–627. <https://doi.org/10.1038/nn1253>
- Rose, C. R., & Konnerth, A. (2001). Stores Not Just for Storage: Intracellular Calcium Release and Synaptic Plasticity. *Neuron*, 31(4), 519–522. [https://doi.org/10.1016/S0896-6273\(01\)00402-0](https://doi.org/10.1016/S0896-6273(01)00402-0)
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in *Psychological Review*, 65(6), 386–408. <https://doi.org/10.1037/h0042519>
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533–536. <https://doi.org/10.1038/323533a0>
- Rumelhart, D., McClelland, J., & Group, C. P. R. (1986). *Parallel distributed processing: explorations in the microstructure of cognition, vol. 2: psychological and biological models*.
- Sanes, J. R., & Lichtman, J. W. (1999). Can molecules explain long-term potentiation? *Nature Neuroscience* 1999 2:7, 2(7), 597–604. <https://doi.org/10.1038/10154>
- Saxe, A. M., McClelland, J. L., & Ganguli, S. (2019). A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences of the United States of America*, 166(23), 11537–11546. <https://doi.org/10.1073/pnas.1820226116>
- Schiller, J., Major, G., Koester, H. J., & Schiller, Y. (2000). NMDA spikes in basal dendrites of cortical pyramidal neurons. *Nature*, 404(6775), 285–289. <https://doi.org/10.1038/35005094>
- Shouval, H. Z., Bear, M. F., & Cooper, L. N. (2002). A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *Proceedings of the National Academy of Sciences*, 99(16), 10831–10836. <https://doi.org/10.1073/pnas.152343099>
- Shouval, H. Z., Wang, S. S. H.-H., & Wittenberg, G. M. (2010). Spike timing dependent plasticity: A consequence of more fundamental learning rules. *Frontiers in Computational Neuroscience*, 4(1), 19. <https://doi.org/10.3389/fncom.2010.00019>
- Spacek, J., & Harris, K. M. (1997). Three-Dimensional Organization of Smooth Endoplasmic Reticulum in Hippocampal CA1 Dendrites and Dendritic Spines of the Immature and Mature Rat. *The Journal of Neuroscience*, 17(1), 190. <https://doi.org/10.1523/JNEUROSCI.17-01-00190.1997>
- Tigaret, C. M., Olivo, V., Sadowski, J. H. L. P., Ashby, M. C., & Mellor, J. R. (2016). Coordinated activation of distinct Ca²⁺ sources and metabotropic glutamate receptors encodes Hebbian synaptic plasticity. *Nature Communications*, 7. <https://doi.org/10.1038/ncomms10289>
- Tong, R., Chater, T. E., Emptage, N. J., & Goda, Y. (2021). Heterosynaptic cross-talk of pre- and postsynaptic strengths along segments of dendrites. *Cell Reports*, 34(4), 108693. <https://doi.org/10.1016/J.CELREP.2021.108693>
- Tran-Van-Minh, A., Cazé, R. D., Abrahamsson, T., Cathala, L., Gutkin, B. S., & DiGregorio, D. a. (2015). Contribution of sublinear and supralinear dendritic integration to neuronal computations. *Frontiers in Cellular Neuroscience*, 9(March), 67. <https://doi.org/10.3389/fncel.2015.00067>
- Turrigiano, G. (2012). Homeostatic synaptic plasticity: local and global mechanisms for stabilizing neuronal function. *Cold Spring Harbor Perspectives in Biology*, 4(1). <https://doi.org/10.1101/CSHPERSPECT.A005736>

Discussion and Conclusions

Summary

In this dissertation, we have explored a variety of different lenses through which we can understand learning and plasticity in individual neurons. In the *Biophysical Perceptron* (**Chapter 1**), we demonstrated that a detailed biological model of a neuron can implement the (weight-constrained) perceptron learning algorithm, with the caveat that capacity is somewhat reduced when all synapses are placed on the apical tuft. This reinforces the idea that biological neurons, despite their complexity, can indeed learn in the manner that linear neurons do, suggesting that pyramidal neurons can indeed act as powerful classification devices. This article appeared in *Frontiers in Computational Neuroscience*.

In **Chapter 2**, we explored an alternative simplification of a neuron, the *Gradient Clusteron*, or *G-Clusteron*, which uses location-dependent nonlinearities (inspired by the NMDA receptor to solve supervised learning tasks). As opposed to the perceptron learning rule, the gradient clusteron learns via *structural plasticity*, i.e. by moving its synaptic locations. This structural plasticity algorithm involved the attraction and repulsion of synapses. We showed that the gradient clusteron can learn to solve a standard benchmark classification task (MNIST) by only optimizing its synaptic locations. We also showed that if we combine the gradient clusteron's structural plasticity algorithm with an addition plasticity rule for modifying synaptic weights, the gradient clusteron could also solve the exclusive or (XOR) problem, a task which cannot be solved by linear neurons. This article appeared in *Frontiers in Computational Neuroscience*.

There are many aspects of the gradient clusteron that are not biologically plausible (such as the multiplicative interactions between inhibitory synapses). Nevertheless, the basic concept of the gradient clusteron algorithm, namely that neurons can optimize the locations of its synapses via synaptic attraction and repulsion, is something that biological neurons may be able to implement via chemical signaling, as we describe in that chapter. While the G-clusteron presents a somewhat different vision of neural learning than the perceptron, the fact that it is possible to simultaneously use both the structural and functional (weight-based) plasticity in the G-clusteron indicates that structural and functional plasticity may not be at odds, but can rather complement each other in order to achieve greater computational functionality, as we showed with the XOR problem. This work appeared in *PLOS Computational Biology* (Moldwin et al., 2021).

The final three chapters of this dissertation dealt with the biological mechanisms of plasticity, namely the calcium control hypothesis. In **Chapter 3** we addressed the question of how cable theory interacts with the calcium control hypothesis to yield a surprising consequence for heterosynaptic plasticity: because voltage attenuates asymmetrically in dendrites, strong synaptic input at a branch proximal to the soma can elicit plastic changes at branches distal to the input location. We proposed that this phenomenon can enable dendritic branches to operate in a hierarchical manner, such that proximal inputs behave as “supervisors” for more distal locations. Although hierarchical heterosynaptic plasticity is a speculative claim that needs to be

experimentally verified, it may have important implications for neural computation and learning. This work appeared in Eneuro (Moldwin et al., 2023).

In **Chapter 4** we mathematically reframed prior models used to describe calcium based-plasticity, resulting in the *fixed point — learning rate (FPLR)* framework for calcium-based plasticity. The crucial idea behind the FPLR framework is that *calcium tells the synaptic weight where to go and how fast it gets there*. We showed that the FPLR framework is highly general and flexible, enabling it to reproduce experimental results while also being mathematically simple and easy to use. The FPLR framework captures both the early and late phases of long-term plasticity, both in the presence and absence of protein synthesis. In addition to replicating frequency- and spike timing dependent- plasticity, we also demonstrated with the FPLR rule that calcium may be responsible for hippocampal place field creation via plateau potential induction, a phenomenon known as behavioral time scale plasticity.

Finally, in **Chapter 5** we returned to the linear neuron, this time to see whether calcium-based plasticity could yield learning rules. In this project, we devised a neuron model called the *calciton*, which assumes that synapses can receive local or global signals from four different sources, each of which is associated with some aspect of the linear neuron. By manipulating the calcium obtained from each calcium source, we showed that it was possible to produce a very wide array of supervised and unsupervised learning rules, including Hebbian and anti-Hebbian plasticity, homeostatic plasticity, the perceptron rule, and many more. The *calciton* thus demonstrates that calcium-based plasticity is an incredibly versatile mechanism that can implement many of the algorithms that theoreticians have postulated for solving learning tasks.

Open questions

Because the nature of this dissertation is entirely theoretical and computational, all of the models of plasticity and learning we propose here require empirical validation. Unfortunately, the “dream experiments” which could truly expose the mechanisms neural learning would require technical capabilities beyond what currently exist. Although our analysis is limited to the single neuron, the behavior of the individual neuron can only be understood if we have knowledge about all of its presynaptic inputs. A comprehensive experimental study of plasticity would require the ability to perform simultaneous, *in vivo* recording of all voltage traces, calcium traces, and locations of each synapse in a neuron over long time scales during behavioral tasks with high spatiotemporal precision. Great progress has been made on these fronts in the past decades. We now have imaging and labeling systems that enable us to observe spine voltage activity *in vivo* (Cornejo et al., 2021), as well as to observe structural changes in synapses over time (Holtmaat & Svoboda, 2009; Villa et al., 2016). Work is also being done to be able to expand the spatial field of these observations so that the entire neuron can be observed simultaneously. (Kerlin et al., 2019).

However, these experimental techniques are still not quite mature enough to address our initial question. “How do plasticity and learning differ between biological neurons and perceptrons?” remains an open query from a biological standpoint. This dissertation proposed some models that can serve as guidelines for future experiments. The calciton model raises the possibility that the mechanism of calcium-based plasticity enables many different kinds of learning rules, and it is likely that these learning rules vary between different neuron types, or

even between synapses in the same neuron. (See (Chindemi et al., 2022) regarding implementation of the calcium control hypothesis in a simulated cortical network of detailed biophysical neurons.)

This dissertation has taken a “optimistic” view toward biological neurons; in other words, we start with the assumption that biology does things in an optimal way, and as such we can look to the biology of real neurons as an inspiration for designing better computational learning systems. This approach certainly paid off with the original McCulloch and Pitts neurons (McCulloch & Pitts, 1943) which eventually became the basis of the deep neural networks used for machine learning today.

We must be open to the possibility, however, that there are diminishing (and possibly even negative) consequences to adding additional biological realism to model neurons used in deep networks. It may be that the brain is an imperfect approximation of an artificial neural network (in the sense that ANNs are optimal for the tasks the brain needs to perform), as opposed to ANNs being imperfect approximations of the brain. While people were once skeptical of the ability of computers to replicate the cognitive powers of the human brain, recent advances in deep learning have ameliorated much of the skepticism. The machine learning community has built deep networks that can beat top players at famously challenging board games and produce stunning new art. Large language models (LLMs) have taken the world by storm in their ability to give useful, humanlike responses to arbitrary text queries, and can outperform humans on standard intelligence benchmark tests like the SAT and LSAT (OpenAI, 2023).

At the same time, there are arenas in which the brain remains superior. Dexterous motor control in non-stereotypical environments which require constant on-the-fly adjustments via a perception-action loop (such as construction work), is still beyond the capabilities (or at least at the limits of) today’s artificial intelligence systems. And while large language models (LLMs) have been enormously helpful as tools, in most fields they are from being able to fully replace human workers. It is still unclear whether and to what extent LLMs “understand” the queries that are posed to it, as evidenced by “hallucinations” where the AI gives bizarre and obviously false answers to certain questions (McKenna et al., 2023). This is not to say that AI won’t eventually get to the point where it can replicate or even surpass humans on every axis, but we may need to have a deeper understanding of the brain to get there.

The burgeoning field of neuroAI, which aims to use our neuroscience knowledge to help advance artificial intelligence and vice versa, can help advance this endeavor (Richards et al., 2019). We can use connectomics data, which gives us the connectivity structure between neurons in the brain (Abbott et al., 2020; Bidel et al., 2023; Eichler et al., 2017; Hildebrand et al., 2017; Scheffer et al., 2020; White et al., 1986; Witvliet et al., 2020; Zheng et al., 2018), to constrain the architecture of neural networks. We can use experimental data from behaving animals to carefully dissect biological circuits and understand how the parts fit together to perform computations(Schaffer et al., 2023). We can use detailed biophysical models to simulate brain function (Markram et al., 2015). We can use the constraints of biology to inspire and constrain the space of neural learning algorithms. (Hinton, 2022; Lillicrap et al., 2020; Pulvermüller et al., 2021). And, as we and others have shown, we can use our knowledge of the details of the single neuron to inspire new forms of neural computation (Pagkalos et al., 2023). We are entering an

exciting new time for artificial intelligence, neuroscience, and the intersection between them. There is a great deal more to be understood and built, but that's what makes the road ahead so exhilarating.

References

- Abbott, L. F., Bock, D. D., Callaway, E. M., Denk, W., Dulac, C., Fairhall, A. L., Fiete, I., Harris, K. M., Helmstaedter, M., Jain, V., Kasthuri, N., LeCun, Y., Lichtman, J. W., Littlewood, P. B., Luo, L., Maunsell, J. H. R., Reid, R. C., Rosen, B. R., Rubin, G. M., ... Van Essen, D. C. (2020). The Mind of a Mouse. *Cell*, 182(6), 1372–1376. <https://doi.org/10.1016/j.cell.2020.08.010>
- Amit, D. J., Wong, K. Y. M., & Campbell, C. (1999). Perceptron learning with sign-constrained weights. *Journal of Physics A: Mathematical and General*, 22(12), 2039–2045. <https://doi.org/10.1088/0305-4470/22/12/009>
- Azevedo, F. A. C., Carvalho, L. R. B., Grinberg, L. T., Farfel, J. M., Ferretti, R. E. L., Leite, R. E. P., Filho, W. J., Lent, R., & Herculano-Houzel, S. (2009). Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *The Journal of Comparative Neurology*, 513(5), 532–541. <https://doi.org/10.1002/CNE.21974>
- Beniaguev, D., Segev, I., & London, M. (2021). Single cortical neurons as deep artificial neural networks. *Neuron*, 109(17), 2727-2739.e3. <https://doi.org/10.1016/J.NEURON.2021.07.002>
- Beniaguev, D., Shapira, S., Segev, I., & London, M. (2022). Multiple Synaptic Contacts combined with Dendritic Filtering enhance Spatio-Temporal Pattern Recognition capabilities of Single Neurons. *BioRxiv*, 2022.01.28.478132. <https://doi.org/10.1101/2022.01.28.478132>
- Bi, G., & Poo, M. (1998). Synaptic Modifications in Cultured Hippocampal Neurons: Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type. *Journal of Neuroscience*, 18(24), 10464–10472. <https://doi.org/10.1523/JNEUROSCI.18-24-10464.1998>
- Bidel, F., Meirovitch, Y., Schalek, R. L., Lu, X., Pavarino, E. C., Yang, F., Peleg, A., Wu, Y., Shomrat, T., Berger, D. R., Shaked, A., Lichtman, J. W., & Hochner, B. (2023). Connectomics of the Octopus vulgaris vertical lobe provides insight into conserved and novel principles of a memory acquisition network. *ELife*, 12. <https://doi.org/10.7554/ELIFE.84257>
- Byrne, J. H. (n.d.). *Introduction to Neurons and Neuronal Networks*. Retrieved September 6, 2023, from <https://nba.uth.tmc.edu/neuroscience/m/s1/introduction.html>
- Chapeton, J., Fares, T., LaSota, D., & Stepanyants, A. (2012). Efficient associative memory storage in cortical circuits of inhibitory and excitatory neurons. *Proceedings of the National Academy of Sciences of the United States of America*, 109(51), E3614-22. <https://doi.org/10.1073/pnas.1211467109>
- Chindemi, G., Abdellah, M., Amsalem, O., Benavides-Piccione, R., Delattre, V., Doron, M., Ecker, A., Jaquier, A. T., King, J., Kumbhar, P., Monney, C., Perin, R., Rössert, C., Tuncel, A. M., Van Geit, W., DeFelipe, J., Graupner, M., Segev, I., Markram, H., & Muller, E. B. (2022). A calcium-based plasticity model for predicting long-term potentiation and depression in the neocortex. *Nature Communications*, 13(1), 3038. <https://doi.org/10.1038/s41467-022-30214-w>

- Cornejo, V. H., Ofer, N., & Yuste, R. (2021). Voltage compartmentalization in dendritic spines in vivo. *Science*. <https://doi.org/10.1126/SCIENCE.ABG0501>
- Eichler, K., Li, F., Litwin-Kumar, A., Park, Y., Andrade, I., Schneider-Mizell, C. M., Saumweber, T., Huser, A., Eschbach, C., Gerber, B., & al., et. (2017). The complete connectome of a learning and memory centre in an insect brain. *Nature*, 548, 175–182.
- Eyal, G., Verhoog, M. B., Testa-Silva, G., Deitcher, Y., Benavides-Piccione, R., DeFelipe, J., de Kock, C. P. J., Mansvelder, H. D., & Segev, I. (2018). Human Cortical Pyramidal Neurons: From Spines to Spikes via Models. *Frontiers in Cellular Neuroscience*, 12(June), 1–24. <https://doi.org/10.3389/fncel.2018.00181>
- Gütig, R., Gutig, R., Gütig, R., & Gutig, R. (2016). Spiking neurons can discover predictive features by aggregate-label learning. *Science*, 351(6277), aab4113–aab4113. <https://doi.org/10.1126/science.aab4113>
- Gütig, R., & Sompolinsky, H. (2006). The tempotron: a neuron that learns spike timing-based decisions. *Nature Neuroscience*, 9(3), 420–428. <https://doi.org/10.1038/nn1643>
- Gütig, R., & Sompolinsky, H. (2009). Time-Warp–Invariant Neuronal Processing. *PLoS Biology*, 7(7), e1000141. <https://doi.org/10.1371/journal.pbio.1000141>
- Harris, K. (n.d.). Axons. In *Kristen Harris*. Retrieved September 5, 2023, from <https://synapseweb.clm.utexas.edu/axons>
- Hildebrand, D. G. C., Cicconet, M., Torres, R. M., Choi, W., Quan, T. M., Moon, J., Wetzel, A. W., Champion, A. S., Graham, B. J., Randlett, O., & al., et. (2017). Whole-brain serial-section electron microscopy in larval zebrafish. *Nature*, 545, 345–349.
- Hinton, G. (2022). *The Forward-Forward Algorithm: Some Preliminary Investigations*. <https://arxiv.org/abs/2212.13345v1>
- Holtmaat, A., & Svoboda, K. (2009). Experience-dependent structural synaptic plasticity in the mammalian brain. *Nature Reviews Neuroscience*, 10(9), 647–658. <https://doi.org/10.1038/nrn2721>
- Jahr, C., & Stevens, C. (1990). A quantitative description of NMDA receptor-channel kinetic behavior. *Journal of Neuroscience*, 10(6), 1830–1837. <https://doi.org/10.1523/JNEUROSCI.10-06-01830.1990>
- Jahr, C., Stevens, C. C. F., Jahr', C. E., Stevens, C. C. F., Jahr, C., & Stevens, C. C. F. (1990). Voltage dependence of NMDA-activated macroscopic conductances predicted by single-channel kinetics. *Journal of Neuroscience*, 10(9), 3178–3182. <https://doi.org/10.1523/JNEUROSCI.10-09-03178.1990>
- Kerlin, A., Boaz, M., Flickinger, D., Maclennan, B. J., Dean, M. B., Davis, C., Spruston, N., & Svoboda, K. (2019). Functional clustering of dendritic activity during decision-making. *eLife*, 8. <https://doi.org/10.7554/eLife.46966>

- Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., & Hinton, G. (2020). Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6), 335–346. <https://doi.org/10.1038/S41583-020-0277-3>
- Markram, H., Muller, E., Ramaswamy, S., Reimann, M. W., Abdellah, M., Sanchez, C. A., Ailamaki, A., Alonso-Nanclares, L., Antille, N., Arsever, S., Kahou, G. A. A., Berger, T. K., Bilgili, A., Buncic, N., Chalimourda, A., Chindemi, G., Courcol, J. D., Delalondre, F., Delattre, V., ... Schürmann, F. (2015). Reconstruction and Simulation of Neocortical Microcircuitry. *Cell*, 163(2), 456–492. <https://doi.org/10.1016/J.CELL.2015.09.029/ATTACHMENT/874AC270-5448-4DCE-B047-9123D6947152/MMC7.MP4>
- McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133. <https://doi.org/10.1007/BF02478259>
- McKenna, N., Li, T., Cheng, L., Hosseini, M. J., Johnson, M., & Steedman, M. (2023). *Sources of Hallucination by Large Language Models on Inference Tasks*. <https://arxiv.org/abs/2305.14552v1>
- Mel, B. W. (1991). The clusteron: Toward a simple abstraction for a complex neuron. *Nips*, 35–42.
- Memmesheimer, R. M., Rubin, R., Ölveczky, B. P., & Sompolinsky, H. (2014). Learning Precisely Timed Spikes. *Neuron*, 82(4), 925–938. <https://doi.org/10.1016/j.neuron.2014.03.026>
- Moldwin, T., Kalmenson, M., & Segev, I. (2021). The gradient clusteron: A model neuron that learns to solve classification tasks via dendritic nonlinearities, structural plasticity, and gradient descent. *PLOS Computational Biology*, 17(5), e1009015. <https://doi.org/10.1371/journal.pcbi.1009015>
- Moldwin, T., Kalmenson, M., & Segev, I. (2023). Asymmetric Voltage Attenuation in Dendrites Can Enable Hierarchical Heterosynaptic Plasticity. *ENeuro*, 10(7), ENEURO.0014-23.2023. <https://doi.org/10.1523/ENEURO.0014-23.2023>
- Moldwin, T., & Segev, I. (2020). Perceptron Learning and Classification in a Modeled Cortical Pyramidal Cell. *Frontiers in Computational Neuroscience*, 14(April), 1–13. <https://doi.org/10.3389/fncom.2020.00033>
- O'Connor, D. H., Wittenberg, G. M., & Wang, S. S. H. (2005). Dissection of bidirectional synaptic plasticity into saturable unidirectional processes. *Journal of Neurophysiology*, 94(2), 1565–1573. <https://doi.org/10.1152/JN.00047.2005/ASSET/IMAGES/MEDIUM/Z9K0080547751001.JPG>
- OpenAI. (2023). *GPT-4 Technical Report*. <https://arxiv.org/abs/2303.08774v3>
- Pagkalos, M., Makarov, R., & Poirazi, P. (2023). *Leveraging dendritic properties to advance machine learning and neuro-inspired computing*. <https://arxiv.org/abs/2306.08007v1>
- Polsky, A., Mel, B. W., & Schiller, J. (2004). Computational subunits in thin dendrites of pyramidal cells. *Nature Neuroscience* 7:6, 7(6), 621–627. <https://doi.org/10.1038/nn1253>

- Pulvermüller, F., Tomasello, R., Henningsen-Schomers, M. R., & Wennekers, T. (2021). Biological constraints on neural network models of cognitive function. *Nature Reviews Neuroscience*, 22(8), 488–502. <https://doi.org/10.1038/S41583-021-00473-5>
- Rall, W. (1959). Branching dendritic trees and motoneuron membrane resistivity. *Experimental Neurology*, 1(5), 491–527. [https://doi.org/10.1016/0014-4886\(59\)90046-9](https://doi.org/10.1016/0014-4886(59)90046-9)
- Rall, W. (1964). Theoretical Significance of Dendritic Trees for Neuronal Input-Output Relations (1964). In *Neural Theory and Modeling*. Stanford University Press (pp. 73–79). The MIT Press. <https://doi.org/10.7551/MITPRESS/6743.003.0015>
- Rall, W. (1967). Distinguishing theoretical synaptic potentials computed for different soma-dendritic distributions of synaptic input. *Journal of Neurophysiology*, 30(5), 1138–1168. <https://doi.org/10.1152/jn.1967.30.5.1138>
- Rall, W., & Rinzel, J. (1973). Branch input resistance and steady attenuation for input to one branch of a dendritic neuron model. *Biophysical Journal*, 13(7), 648–688. [https://doi.org/10.1016/S0006-3495\(73\)86014-X](https://doi.org/10.1016/S0006-3495(73)86014-X)
- Réus, G. Z., Abelaira, H. M., Tuon, T., Titus, S. E., Ignácio, Z. M., Rodrigues, A. L. S., & Quevedo, J. (2016). Glutamatergic NMDA Receptor as Therapeutic Target for Depression. *Advances in Protein Chemistry and Structural Biology*, 103, 169–202. <https://doi.org/10.1016/BS.APCSB.2015.10.003>
- Richards, B. A., Lillicrap, T. P., Beaudoin, P., Bengio, Y., Bogacz, R., Christensen, A., Clopath, C., Costa, R. P., de Berker, A., Ganguli, S., Gillon, C. J., Hafner, D., Kepecs, A., Kriegeskorte, N., Latham, P., Lindsay, G. W., Miller, K. D., Naud, R., Pack, C. C., ... Kording, K. P. (2019). A deep learning framework for neuroscience. *Nature Neuroscience* 2019 22:11, 22(11), 1761–1770. <https://doi.org/10.1038/s41593-019-0520-2>
- Schaffer, E. S., Mishra, N., Whiteway, M. R., Li, W., Vancura, M. B., Freedman, J., Patel, K. B., Voleti, V., Paninski, L., Hillman, E. M. C., Abbott, L. F., & Axel, R. (2023). The spatial and temporal structure of neural activity across the fly brain. *Nature Communications* 2023 14:1, 14(1), 1–15. <https://doi.org/10.1038/s41467-023-41261-2>
- Scheffer, L. K., Xu, C. S., Januszewski, M., Lu, Z., Takemura, S.-Y., Hayworth, K. J., Huang, G. B., Shinomiya, K., Maitin-Shepard, J., Berg, S., & al., et. (2020). A Connectome and Analysis of the Adult Drosophila Central Brain. *BioRxiv*. <https://doi.org/10.1101/2020.04.07.030213>
- Shouval, H. Z., Wang, S. S. H.-H., & Wittenberg, G. M. (2010). Spike timing dependent plasticity: A consequence of more fundamental learning rules. *Frontiers in Computational Neuroscience*, 4(1), 19. <https://doi.org/10.3389/fncom.2010.00019>
- Stufflebeam, R. (2006). *Neurons, Synapses, Action Potentials, and Neurotransmission*. https://mind.ilstu.edu/curriculum/neurons_intro/neurons_intro.html
- Synapses*. (n.d.). Labster Theory. Retrieved September 6, 2023, from <https://theory.labster.com/synapses/>

- Tran-Van-Minh, A., Cazé, R. D., Abrahamsson, T., Cathala, L., Gutkin, B. S., & DiGregorio, D. a. (2015). Contribution of sublinear and supralinear dendritic integration to neuronal computations. *Frontiers in Cellular Neuroscience*, 9(March), 67. <https://doi.org/10.3389/fncel.2015.00067>
- Villa, K. L., Berry, K. P., Subramanian, J., Cha, J. W., Oh, W. C., Kwon, H. B., Kubota, Y., So, P. T. C., & Nedivi, E. (2016). Inhibitory Synapses Are Repeatedly Assembled and Removed at Persistent Sites In Vivo. *Neuron*, 89(4), 756–769. <https://doi.org/10.1016/j.neuron.2016.01.010>
- Von Neumann, J. (1958). *The computer & the brain*. 83.
- White, J. G., Southgate, E., Thomson, J. N., & Brenner, S. (1986). The structure of the nervous system of the nematode *Caenorhabditis elegans*. *Philos. Trans. R. Soc. Lond. B Biol. Sci.*, 314, 1–340.
- Wittner, L., Henze, D. A., Záboršky, L., & Buzsáki, G. (2007). Three-dimensional reconstruction of the axon arbor of a CA3 pyramidal cell recorded and filled in vivo. *Brain Structure & Function*, 212(1), 75. <https://doi.org/10.1007/S00429-007-0148-Y>
- Witvliet, D., Mulcahy, B., Mitchell, J. K., Meirovitch, Y., Berger, D. R., Wu, Y., Liu, Y., Koh, W. X., Parvathala, R., Holmyard, D., & al., et. (2020). Connectomes across development reveal principles of brain maturation in *C. elegans*. *BioRxiv*. <https://doi.org/10.1101/2020.04.30.066209>
- Zheng, Z., Lauritzen, J. S., Perlman, E., Robinson, C. G., Nichols, M., Milkie, D., Torrens, O., Price, J., Fisher, C. B., Sharifi, N., & al., et. (2018). A Complete Electron Microscopy Volume of the Brain of Adult *Drosophila melanogaster*. *Cell*, 174, 730-743.e22.

תקציר

המודל הקונקסנייטי של תפקוד המוח טווען שניtan להבין בצורה הטובה ביותר את המוח כרשת חישובית של נירונים, שמתפרקדים בנפרד צמתים של אחד מהם מבצע פעולה מתמטית פשוטה. המוח יכול ללמידה לאחסן מידע חדש על ידי שינוי עצמות החיבור, או המשקولات, בין אוטם נירונים. הצלחה המדינה של רשתות עצמים מלאכותיות ולמידה عمוקה - המבוססת על עקרונות קונקסניות - בפתרון משימות חישוביות קשות וונתנת אמון ברעיון שהמוח עשוי לעסוק על פי עקרונות דומים. אבל בשאלת השאלה, עד כמה דומים נירונים ביולוגיים אמיתיים ליחידות הליניאריות הפחות המשמשות ברשתות עצמים מלאכותיות, הן מבחן תפקוד הקלט-פלט שלהם והן מבחן האופן שבו הם לומדים לפתור משימות חדשות? התזה זו עוסקת בשאלת זו מגוון נקודות מבט.

אנו מתחלים בשאלת האם נירונים ביולוגיים יכולים לישם את אלגוריתם למידת הפרטורים, אחד מלאgorיתמי הלמידה הפחותים ביותר לנירונים מלאכותיים. באמצעות מודל כבל בי-פיizi מפורט של נירון פירמידלי בקיליפת המוח שכבה 5, מתרבר שההתשובה היא "כן".

לאחר שחקרנו האם נירונים ביולוגיים יכולים להתנהג כמו נירונים מלאכותיים, אנו שואלים את השאלה הפוכה: האם בכלל להמציא מודל נירון מלאכותי פשוט אשר לוודأن מההתפעות הלא-ליניארית המעניניות שנצפו בביולוגיה, ולאחר מכן להשתמש במודל זהה כדי לפתור משימת למידה? לשם כך, פיתחנו את הגרדיינט קלאסטרון, מודל נירון משתמש באין-לייניאריות דדרטיבית תלויה מיקום כדי ליצור פלט. הפקנו באופן אינטיטי כל למידה בדרגה גבוהה עבור הדרון Gradient Clustering והראינו שהוא אכן יכול ללמידה משימות סיווג, כגון MNIST או בעית XOR, על ידי גירמת סינפסות לתקרב או להתרחק זו מזו על גבי הדנדרייט.

במהרש, כדי להרחב את ההבנה כיצד המבנה המרחבי של דנדרייטים יכול להשפיע על פלסטיות ולמידה, בחנו גם פלסטיות הטרנסינפטיות במודלים של כבילים של דנדרייטים, והצענו שהא-סימטריה החשמלית הטבעה בעיצים דנדריטים יכולה לאפשר סכימת פיקוח היררכית בדנדרייט באמצעות פעילות חשמלית תלויה NMDA.

לאחר מכן אנו פונים לבחון את המנגנונים הביאולוגיים העומדים בסיס הפלסטיות הסינפטיות באמצעות השערת בקרת הסידן (CCH), אשר גורסים שכיוון גודל הפלסטיות תלויות ברכיב הסידן בסינפסה. בהתבסס על עבודה קודמת, אנו מציעים מודל מתמטי פשוט להבנת השערת בקרת הסידן, המכונה כלל FPLR – Learning Rate – Fixed Point – נקודת קבועה – קצב למידה. לאחר מכן השתמשנו בכלל זה כדי למדוד כמה תוצאות חדשות ומרגשות הנוגעות לפלסטיות הרלוונטי בפרק זמן התנהגותיים (BTSP). אנו מציעים גם מודל נירון פשוט, המכונה קלציטרון (Calcitrone), המשמש בהשערה בקרת הסידן כדי לישם מגוון רחב של תוצאות למידה ופלסטיות. עבודות גמר זו מבוססת על 3 מאמרם שפורסמו ו-2 מאמרים נוספים לקרה פרטום (אחד מהם bioRxiv).

עובדת זו נעשתה בהדריכתו של פרופסור עידן שגב

פלסיטיות ואלגוריתמי למידה במודלים של הנירון הבודד

חיבור לשם קבלת תואר דוקטור לפילוסופיה

מאת

טוביה מולדוין

הוגש לסנט האוניברסיטה העברית בירושלים

אוקטובר 2023